DATA HANDBOOK

Microcontrollers NMOS, CMOS

BOOK IC14 1989

Philips Components

PHILIPS

IC14 MICROCONTROLLERS

page

Selection guide
- Functional index 1
- Numerical index 7
- Maintenance type list 17

Type designation 20

Rating systems 24

Handling MOS devices 28

Single-chip 8-bit microcontrollers (NMOS, CMOS) 29

Package information 715
- Package outlines 717
- Soldering 733

User Manual 1988 *
"Single-chip 8-bit Microcontrollers"
Available on request

* Includes I^2C-bus specification.

SELECTION GUIDE

Functional index **1**
Numerical index **7**
Maintenance type list **17**

FUNCTIONAL INDEX

type number	RAM	ROM	description	page
SINGLE-CHIP 8-BIT MICROCONTROLLERS				
NMOS				
MAB8035HLP; HLWP	64	–	ROM-less version of MAB8048H	99
MAF8035HLP; HLWP	64	–	ROM-less version of MAB8048H; extended temperature range	99
MAF80A35HLP; HLWP	64	–	ROM-less version of MAB8048H; automotive temperature; reduced frequency	99
MAB8048HP; HWP	64	1K	mask-programmable ROM	99
MAF8048HP; HWP	64	1K	like MAB8048H; extended temperature range	99
MAF80A48HP; HWP	64	1K	like MAB8048H; automotive temperature; reduced frequency	99
MAB8411P; T	64	1K	plus 8-bit LED driver	125
MAF8411P	64	1K	plus 8-bit LED driver; extended temperature range	125
MAF84A11P	64	1K	plus 8-bit LED driver; automotive temperature; reduced frequency	125
MAB8421P; T	64	2K	plus 8-bit LED driver	125
MAF8421P	64	2K	plus 8-bit LED driver; extended temperature range	125
MAF84A21P	64	2K	plus 8-bit LED driver; automotive temperature; reduced frequency	125
MAB8422P	64	2K	plus 8-bit LED driver	157
MAF8422P	64	2K	plus 8-bit LED driver; extended temperature range	157
MAF84A22P	64	2K	plus 8-bit LED driver; automotive temperature; reduced frequency	157
MAB8031AHP; AHWP	128	–	ROM-less version of MAB8051AH	31
MAF8031AHP; AHWP	128	–	ROM-less version of MAB8051AH; extended temperature range	31
MAF80A31AHP; AHWP	128	–	ROM-less version of MAB8051AH; automotive temperature; reduced frequency	31
MAB8039HLP; HLWP	128	–	ROM-less version of MAB8049H	99
MAF8039HLP; HLWP	128	–	ROM-less version of MAB8049H; extended temperature range	99
MAF80A39HLP; HLWP	128	–	ROM-less version of MAB80A49H; automotive temperature; reduced frequency	99
MAB8401B; WP	128	–	piggy-back; bond-out version for MAB84X1 family plus 8-bit LED-driver	125

Operating temperature range : 0 to 70 °C.
Extended temperature range : –40 to + 85 °C.
Automotive temperature range: –40 to + 110 °C;
except MAF80A31/51AH : –40 to + 100 °C;
MAF80A31/51AH-2: –40 to + 110 °C.

type number	description			page
NMOS (continued)	RAM	ROM		
MAB8049HP; HWP	128	2K		99
MAF8049HP; HWP	128	2K	like MAB8049H; extended temperature range	99
MAF80A49HP; HWP	128	2K	like MAB8049H; automotive temperature; reduced frequency	99
MAB8051AHP; AHWP	128	4K	mask-programmable ROM	31
MAF8051AHP; AHWP	128	4K	like MAB8051AH; extended temperature range	31
MAF80A51AHP; AHWP	128	4K	like MAB8051AH; automotive temperature; reduced frequency	31
MAB8441P; T	128	4K	plus 8-bit LED driver	125
MAF8441P	128	4K	plus 8-bit LED driver; extended temperature range	125
MAF84A41P	128	4K	plus 8-bit LED driver; automotive temperature; reduced frequency	125
MAB8442P	128	4K	plus 8-bit LED driver	157
MAF8442P	128	4K	plus 8-bit LED driver; extended temperature range	157
MAF84A42P	128	4K	plus 8-bit LED driver; automotive temperature; reduced frequency	157
MAB8461P; T	128	6K	plus 8-bit LED driver	125
MAF8461P	128	6K	plus 8-bit LED driver; extended temperature range	125
MAF84A61P	128	6K	plus 8-bit LED driver; automotive temperature; reduced frequency	125
MAB8032AHWP	256	–	ROM-less version of MAB8052AH	65
MAF8032AHWP	256	–	ROM-less version of MAF8052; extended temperature range	65
MAF80A32AHWP	256	–	ROM-less version of MAF80A52; automotive temperature; reduced frequency	65
MAB8040HLP; HLWP	256	–	ROM-less version of MAB8050H	99
MAF8040HLP; HLWP	256	–	ROM-less version of MAB8050H; extended temperature range	99
MAF80A40HLP	256	–	ROM-less version of MAB8050H; automotive temperature; reduced frequency	99
MAB8050HP; HWP	256	4K	mask-programmable ROM	99
MAF8050HP; HWP	256	4K	like MAB8050H; extended temperature range	99
MAF80A50HP; HWP	256	4K	like MAB8050H; automotive temperature; reduced frequency	99
MAB8052AHWP	256	8K	mask-programmable ROM	65
MAF8052AHWP	256	8K	mask-programmable ROM; extended temperature range	65
MAF80A52AHWP	256	8K	mask-programmable ROM; automotive temperature; reduced frequency	65

Operating temperature range : 0 to 70 °C.
Extended temperature range : −40 to + 85 °C.
Automotive temperature range: −40 to + 110 °C;
except MAF80A31/51AH : −40 to + 100 °C;
MAF80A31/51AH-2: −40 to + 110 °C.

type number	description			page
CMOS				
	RAM	ROM		
PCF84C12P; T	64	1K	mask-programmable ROM	577
PCF84C21P; T	64	2K	plus 8-bit LED driver; extended temperature range	593
PCF84C22P; T	64	2K	mask-programmable ROM	577
PCF84C42P; T	64	4K	mask-programmable ROM	577
PCB80C31BH-3P; 3WP	128	–	ROM-less version of PCB80C51	177
PCF80C31BH-3P; 3WP	128	–	ROM-less version of PCF80C51	177
PCA80C31BH-3P; 3WP	128	–	ROM-less version of PCA80C51	177
PCB80C39P; WP	128	–	ROM-less version of PCB80C49	209
PCF80C39P; WP	128	–	ROM-less version of PCF80C49; extended temperature range	209
PCA80C39P; WP	128	–	ROM-less version of PCA80C49; automotive temperature	209
PCB80C49P; WP	128	2K	mask-programmable ROM	209
PCF80C49P; WP	128	2K	like PCB80C49; extended temperature range	209
PCA80C49P; WP	128	2K	like PCB80C49; automotive temperature	209
PCB80C51BH-3P; 3WP	128	4K	mask-programmable ROM	177
PCF80C51BH-3P; 3WP	128	4K	mask-programmable ROM; extended temperature range; reduced frequency	177
PCA80C51BH-3P; 3WP	128	4K	mask-programmable ROM; automotive temperature; reduced frequency	177

Operating temperature range (PCB80C51/C31BH-3) : 0 to 70 °C.
(PCF80C51/C31BH-3; PCF84CXX): –40 to + 85 °C.
(PCA80C51/C31BH-3) : –40 to + 125 °C.
(PCA80C39/49) : –40 to + 110 °C.

type number	description			page
CMOS (continued)	RAM	ROM		
PCF84C41P; T	128	4K	plus 8-bit LED driver; extended temperature range	593
PCF84C430H	128	4K	with LCD driver	659
SC83C451	128	4K	I/O expanded microcontroller	241
PCB83C851P; WP	128	4K	with 256 bytes of EEPROM	439
SC80C451	128	–	ROM-less version of SC83C451	241
PCB80C851P; WP	128	–	ROM-less version of PCB83C851	439
PCD3315CP; T	160	1K	telephony microcontroller	471
PCD3343P; T	224	3K	telephony microcontroller	475
PCD3349P; T	224	4K	telephony microcontroller	515
PCB80C552WP	256	–	ROM-less version of PCB83C552	257
PCA80C552WP	256	–	like PCB80C552 with extended temperature range	257
PCF80C552WP	256	–	like PCB80C552 with extended temperature range	257
PCB83C562WP	256	8K	PCB80C51 plus additional functions	307
PCF83C562WP	256	8K	like PCB83C562 with extended temperature range	307
PCA83C562WP	256	8K	like PCB83C562 with extended temperature range	307
PCB80C652P; WP	256	–	ROM-less version of PCB83C652	353
PCF80C652P; WP	256	–	like PCB80C652 with extended temperature range	353
PCA80C652P; WP	256	–	like PCB80C652 with extended temperature range	353
PCB80C662P; WP; H	256	–	ROM-less version of PCB83C652	391
PCF80C662P; WP; H	256	–	like PCB80C662 with extended temperature range	391
PCA80C662P; WP; H	256	–	like PCB80C662 with extended temperature range	391

Operating temperature range (PCB versions): 0 to 70 °C.
(PCF versions): –40 to + 85 °C.
(PCA versions): –40 to † 125 °C.
(PCD versions): –25 to + 70 °C.

type number	description			page
CMOS (continued)				
	RAM	ROM		
PCB80C654H; P; WP	256	–	ROM-less version of PCB83C654	397
PCF80C654H; P; WP	256	–	like PCB80C654 with extended temperature range	397
PCA80C654H; P; WP	256	–	like PCB80C654 with extended temperature range	397
PCF84C00B; T	256	–	bond-out version PCF84CXX family	593
PCB83C552WP	256	8K	PCB80C51 plus additional functions	257
PCF83C552WP	256	8K	like PCB83C552 with extended temperature range	257
PCA83C552WP	256	8K	like PCB83C552 with extended temperature range	257
PCB80C562WP	256	–	ROM-less version of PCB83C562	307
PCF80C562WP	256	–	like PCB80C562 with extended temperature range	307
PCA80C562WP	256	–	like PCB80C562 with extended temperature range	307
PCB83C652P; WP	256	8K	PCB80C51 plus additional functions	353
PCF83C652P; WP	256	8K	like PCB83C652 with extended temperature range	353
PCA83C652P; WP	256	8K	like PCB83C652 with extended temperature range	353
PCB83C662P; WP; H	256	8K	PCB80C51 plus additional functions	391
PCF83C662P; WP; H	256	8K	like PCB83C662 with extended temperature range	391
PCA83C662P; WP; H	256	8K	like PCB83C662 with extended temperature range	391
PCF84C81P; T	256	8K	plus 8-bit LED driver; extended temperature range	593
PCF84C85P; T	256	8K	32 I/O; extended temperature range	619

Operating temperature range (PCB versions): 0 to 70 °C.
(PCF versions): −40 to + 85 °C.
(PCA versions): −40 to + 125 °C.

NUMERICAL INDEX

type number	description			package code	page
SINGLE-CHIP 8-BIT MICROCONTROLLERS					
NMOS					
	RAM	ROM			
MAB8031AHP	128	–	ROM-less version of MAB8051AH	DIL-40; SOT129	31
MAB8031AHWP	128	–	ROM-less version of MAB8051AH	44-PLCC; SOT187	31
MAB8032AHP	256	–	ROM-less version of MAB8052AH	DIL-40; SOT129	65
MAB8032AHWP	256	–	ROM-less version of MAB8052AH	44-PLCC; SOT187	65
MAB8035HLP	64	–	ROM-less version of MAB8048H	DIL-40; SOT129	99
MAB8035HLWP	64	–	ROM-less version of MAB8048H	44-PLCC; SOT187	99
MAB8039HLP	128	–	ROM-less version of MAB8049H	DIL-40; SOT129	99
MAB8039HLWP	128	–	ROM-less version of MAB8049H	44-PLCC; SOT187	99
MAB8040HLP	256	–	ROM-less version of MAB8050H	DIL-40; SOT129	99
MAB8040HLWP	256	–	ROM-less version of MAB8050H	44-PLCC; SOT187	99
MAB8048HP	64	1K		DIL-40; SOT129	99
MAB8048HWP	64	1K		44-PLCC; SOT187	99
MAB8049HP	128	2K		DIL-40; SOT129	99
MAB8049HWP	128	2K		44-PLCC; SOT187	99
MAB8050HP	256	4K		DIL-40; SOT129	99
MAB8050HWP	256	4K		44-PLCC; SOT187	99
MAB8051AHP	128	4K	mask-programmable ROM	DIL-40; SOT129	31
MAB8051AHWP	128	4K	mask-programmable ROM	44-PLCC; SOT187	31
MAB8052AHP	256	8K	mask-programmable ROM	DIL-40; SOT129	65
MAB8052AHWP	256	8K	mask-programmable ROM	44-PLCC; SOT187	65
MAB8401B	128	–	bond-out version for MAB84X1 family plus 8-bit LED-driver	28/28 Piggy-back	125
MAB8401WP	128	–	bond-out version for MAB84X1 family plus 8-bit LED-driver	68-PLCC; SOT188	125
MAB8411P	64	1K	plus 8-bit LED driver	DIL-28; SOT117	125
MAB8411T	64	1K	plus 8-bit LED driver	SO28; SOT136A	125
MAB8421P	64	2K	plus 8-bit LED driver	DIL-28; SOT117	125
MAB8421T	64	2K	plus 8-bit LED driver	SO28; SOT136A	125

Operating temperature range : 0 to 70 °C.
Extended temperature range : –40 to + 85 °C.
Automotive temperature range: –40 to + 110 °C.

type number	description			package code	page
NMOS (continued)	RAM	ROM			
MAB8422P	64	2K	plus 8-bit LED driver	DIL-20; SOT146	157
MAB8441P	128	4K	plus 8-bit LED driver	DIL-28; SOT117	125
MAB8441T	128	4K	plus 8-bit LED driver	SO28; SOT136A	125
MAB8442P	128	4K	plus 8-bit LED driver	DIL-20; SOT146	157
MAB8461P	128	6K	plus 8-bit LED driver	DIL-28; SOT117	125
MAB8461T	128	6K	plus 8-bit LED driver	SO28; SOT136A	125
MAF8031AHP	128	–	ROM-less version of MAB8051AH; extended temperature range	DIL-40; SOT129	31
MAF8031AHWP	128	–	ROM-less version of MAB8051AH; extended temperature range	44-PLCC; SOT187	31
MAF80A31AHP	128	–	ROM-less version of MAB8051AH; automotive temperature; reduced frequency	DIL-40; SOT129	31
MAF80A31AHWP	128	–	ROM-less version of MAB8051AH; automotive temperature; reduced frequency	44-PLCC; SOT187	31
MAF8032AH	256	–	ROM-less version of MAF8052AH; extended temperature range	DIL-40; SOT129	65
MAF80A32AHP	256	–	ROM-less version of MAF80A52AH; automotive temperature; reduced frequency	DIL-40; SOT129	65
MAF8032AHWP	256	–	ROM-less version of MAF80A52AH; extended temperature range	44-PLCC; SOT187	65
MAF80A32AHWP	256	–	ROM-less version of MAF80A52AH; automotive temperature; reduced frequency	44-PLCC; SOT187	65
MAF8035HLP	64	–	ROM-less version of MAB8048H; extended temperature range	DIL-40; SOT129	99
MAF80A35HLP	64	–	ROM-less version of MAB8048H; automotive temperature; reduced frequency	DIL-40; SOT129	99
MAF8039HLP	128	–	ROM-less version of MAB8049H; extended temperature range	DIL-40; SOT129	99
MAF80A39HLP	128	–	ROM-less version of MAB8049H; automotive temperature; reduced frequency	DIL-40; SOT129	99
MAF8040HLP	256	–	ROM-less version of MAB8050H; extended temperature range	DIL-40; SOT129	99
MAF80A40HLP	256	–	ROM-less version of MAB8050H; automotive temperature; reduced frequency	DIL-40; SOT129	99
MAF8048HP	64	1K	like MAB8048H; extended temperature range	DIL-40; SOT129	99

Operating temperature range: 0 to 70 °C.
Extended temperature range: −40 to + 85 °C.
Automotive temperature range : −40 to + 110 °C;
except MAF80A31/51AH only: −40 to + 100 °C.

type number	description			package code	page
NMOS					
	RAM	ROM			
MAF80A48HP	64	1K	like MAB8048H; automotive temperature; reduced frequency	DIL-40; SOT129	99
MAF8049HP	128	2K	like MAB8049H; extended temperature range	DIL-40; SOT129	99
MAF80A49HP	128	2K	like MAB8049H; automotive temperature; reduced frequency	DIL-40; SOT129	99
MAF8050HP	256	4K	like MAB8050H; extended temperature range	DIL-40; SOT129	99
MAF80A50HP	256	4K	like MAB8050H; automotive temperature; reduced frequency	DIL-40; SOT129	99
MAF8051AHP	128	4K	like MAB8051AH; extended temperature range	DIL-40; SOT129	31
MAF8051AHWP	128	4K	like MAB8051AH; extended temperature range	44-PLCC; SOT187	31
MAF80A51AHP	128	4K	like MAB8051AH; automotive temperature; reduced frequency	DIL-40; SOT129	31
MAF80A51AHWP	128	4K	like MAB8051AH; automotive temperature; reduced frequency	44-PLCC; SOT187	31
MAF8052AHP	256	8K	mask programmable ROM; extended temperature range	DIL-40; SOT129	65
MAF80A52AHP	256	8K	mask programmable ROM; automotive temperature; reduced frequency	DIL-40; SOT129	65
MAF8052AHWP	256	8K	mask programmable ROM; extended temperature range	44-PLCC; SOT187	65
MAF80A52AHWP	256	8K	mask programmable ROM; automotive temperature; reduced frequency	44-PLCC; SOT187	65
MAF8411P	64	1K	plus 8-bit LED driver; extended temperature range	DIL-28; SOT117	125
MAF84A11P	64	1K	plus 8-bit LED driver; automotive temperature; reduced frequency	DIL-28; SOT117	125
MAF8421P	64	2K	plus 8-bit LED driver; extended temperature range	DIL-28; SOT117	125
MAF84A21P	64	2K	plus 8-bit LED driver; automotive temperature; reduced frequency	DIL-28; SOT117	125

Operating temperature range : 0 to 70 °C.
Extended temperature range : −40 to + 85 °C.
Automotive temperature range : −40 to + 110 °C.
except MAF80A31/51AH only: −40 to + 100 °C.

type number	description			package code	page
NMOS (continued)	RAM	ROM			
MAF8422P	64	2K	plus 8-bit LED driver; extended temperature range	DIL-20; SOT146	157
MAF84A22P	64	2K	plus 8-bit LED driver; automotive temperature; reduced frequency	DIL-20; SOT146	157
MAF8441P	128	4K	plus 8-bit LED driver; extended temperature range	DIL-28; SOT117	125
MAF84A41P	128	4K	plus 8-bit LED driver; automotive temperature; reduced frequency	DIL-28; SOT117	125
MAF8442P	128	4K	plus 8-bit LED driver; extended temperature range	DIL-20; SOT146	157
MAF84A42P	128	4K	plus 8-bit LED driver; automotive temperature; reduced frequency	DIL-20; SOT146	157
MAF8461P	128	6K	plus 8-bit LED driver; extended temperature range	DIL-28; SOT117	125
MAF84A61P	128	6K	plus 8-bit LED driver; automotive temperature; reduced frequency	DIL-28; SOT117	125

Operating temperature range PCA80C31/51BH : −40 to + 125 °C.
MAF84XX series : −40 to + 80 °C.
MAF84AXX series: −40 to + 110 °C.

type number	description			package code	page
CMOS					
	RAM	ROM			
PCA80C31BH-3P	128	–	ROM-less version of PCA80C51; extended temperature range	DIL-40; SOT129	177
PCA80C31BH-3WP	128	–	ROM-less version of PCA80C51; extended temperature range	44-PLCC; SOT187	177
PCA80C51BH-3P	128	4K	like PCB80C51; extended temperature range	DIL-40; SOT129	177
PCA80C51BH-3WP	128	4K	like PCB80C51; extended temperature range	44-PLCC; SOT187	177
PCA80C39P	128	–	ROM-less version of PCA80C39; extended temperature range	DIL-40; SOT129	209
PCA80C39WP	128	–	ROM-less version of PCA80C39; extended temperature range	44-PLCC; SOT187	209
PCA80C49P	128	2K	like PCB80C49; extended temperature range	DIL-40; SOT129	209
PCA80C49WP	128	2K	like PCB80C49; extended temperature range	44-PLCC; SOT187	209
PCA80C552WP	256	–	like PCB80C552; extended temperature range	68-PLCC; SOT188	257
PCA80C562WP	256	–	like PCB80C562; extended temperature range	68-PLCC; SOT188	307
PCA80C652P	256	–	ROM-less version of PCA83C652; extended temperature range	DIL-40; SOT129	353
PCA80C652WP	256	–	ROM-less version of PCA83C652; extended temperature range	44-PLCC; SOT187	353
PCA80C662P	256	–	like PCB80C662; extended temperature range	DIL-40; SOT129	391

Operating temperature range: –40 to + 125 °C.
except PCA80C39/C49 : –40 to + 110 °C.

type number	description			package code	page
CMOS (continued)	RAM	ROM			
PCA80C662WP	256	–	like PCB80C662; extended temperature range	44-PLCC; SOT187	391
PCA80C662H	256	–	like PCB80C662; extended temperature range	44-QFP; SOTXXX	391
PCA83C552WP	256	8K	like PCB83C552; extended temperature range	68-PLCC; SOT188	257
PCA83C562WP	256	8K	like PCB83C562; extended temperature range	68-PLCC; SOT188	307
PCA83C652P	256	8K	like PCB83C652; extended temperature range	DIL-40; SOT129	353
PCA83C652WP	256	8K	like PCB83C652; extended temperature range	44-PLCC; SOT187	353
PCA83C654P	256	16K	like PCB83C654; extended temperature range	DIL-40; SOT129	397
PCA83C654WP	256	16K	like PCB83C654; extended temperature range	44-PLCC; SOT187	397
PCA83C654H	256	16K	like PCB83C654; extended temperature range	44-QFP; SOT205A	397
PCA83C662P	256	8K	like PCB83C662; extended temperature range	DIL-40; SOT129	391
PCA83C662WP	256	8K	like PCB83C662; extended temperature range	44-PLCC; SOT187	391
PCA83C662H	256	8K	like PCB83C662; extended temperature range	44-QFP; SOTXXX	391

Operating temperature range: −40 to + 125 °C.

type number	description			package code	page
CMOS	RAM	ROM			
PCB80C31BH-3P	128	–	ROM-less version of PCB80C51	DIL-40; SOT129	177
PCB80C31BH-3WP	128	–	ROM-less version of PCB80C51	44-PLCC; SOT187	177
PCB80C39P	128	–	ROM-less version of PCB80C49	DIL-40; SOT129	209
PCB80C39WP	128	–	ROM-less version of PCB80C49	44-PLCC; SOT187	209
PCB80C49P	128	2K	mask-programmable ROM	DIL-40; SOT129	209
PCB80C49WP	128	2K	mask-programmable ROM	44-PLCC; SOT187	209
PCB80C51BH-3P	128	4K	mask-programmable ROM	DIL-40; SOT129	177
PCB80C51BH-3WP	128	4K	mask-programmable ROM	44-PLCC; SOT187	177
PCB80C552WP	256	–	ROM-less version of PCB83C552	68-PLCC; SOT188	257
PCB80C562WP	256	–	ROM-less version of PCB83C562	68-PLCC; SOT188	307
PCB80C652P	256	–	ROM-less version of PCB83C652	DIL-40; SOT129	353
PCB80C652WP	256	–	ROM-less version of PCB83C652	44-PLCC; SOT187	353
PCB80C662P	256	–	ROM-less version of PCB83C662	DIL-40; SOT129	391
PCB80C662WP	256	–	ROM-less version of PCB83C662	44-PLCC; SOT187	391
PCB80C662H	256	–	ROM-less version of PCB83C662	44-QFP; SOTXXX	391
PCB80C851P	256	–	ROM-less version of PCB83C851	DIL-40; SOT129	439
PCB80C851WP	256	–	ROM-less version of PCB83C851	44-PLCC; SOT187	439
PCB83C552WP	256	8K	PCB80C51 plus additional functions	68-PLCC; SOT188	257
PCB83C562WP	256	8K	PCB80C51 plus additional functions	68-PLCC; SOT188	307
PCB83C652P	256	8K	PCB80C51 plus additional functions	DIL-40; SOT129	353
PCB83C652WP	256	8K	PCB80C51 plus additional functions	44-PLCC; SOT187	353

Operating temperature range: 0 to + 70 °C.

type number	description			package code	page
CMOS (continued)	RAM	ROM			
PCB83C662P	256	8K	PCB80C51 plus additional functions	DIL-40; SOT129	391
PCB83C662WP	256	8K	PCB80C51 plus additional functions	44-PLCC; SOT187	391
PCB83C662H	256	8K	PCB80C51 plus additional functions	44-QFP; SOTXXX	391
PCB83C654H	256	16K	PCB80C51 plus additional functions	44-QFP; SOT205A	397
PCB83C654P	256	16K	PCB80C51 plus additional functions	DIL-40; SOT129	397
PCB83C654WP	256	16K	PCB80C51 plus additional functions	44-PLCC; SOT187	397
PCB83C851P	256	4K	PCB80C51 plus additional functions	DIL-40; SOT129	439
PCB83C851WP	256	4K	PCB80C51 plus additional functions	44-PLCC; SOT187	439
PCD3315CP	160	1,2K	telephony microcontroller	DIL-28; SOT117	471
PCD3315CT	160	1,2K	telephony microcontroller	SO28; SOT136A	471
PCD3343P	224	3K	telephony microcontroller	DIL-28; SOT117	475
PCD3343T	224	3K	telephony microcontroller	SO28; SOT136A	475
PCD3349P	224	4K	telephony microcontroller	DIL-28; SOT117	515
PCD3349T	224	4K	telephony microcontroller	SO28; SOT136A	515
PCF80C31BH-3P	128	–	ROM-less version of PCB80C51	DIL-40; SOT129	177
PCF80C31BH-3WP	128	–	ROM-less version of PCB80C51	44-PLCC; SOT187	177
PCF80C39P	128	–	ROM-less version of PCB80C49; extended temperature range	DIL-40; SOT129	209
PCF80C39WP	128	–	ROM-less version of PCB80C49; extended temperature range	44-PLCC; SOT187	209
PCF80C49P	128	2K	like PCB80C49; extended temperature range	DIL-40; SOT129	209
PCF80C49WP	128	2K	like PCB80C49; extended temperature range	44-PLCC; SOT187A	209
PCF80C51BH-3P	128	4K	mask-programmable ROM	DIL-40; SOT129	177
PCF80C51BH-3WP	128	4K	mask-programmable ROM	44-PLCC; SOT187	177
PCF80C552WP	256	–	like PCB80C552; extended temperature range	68-PLCC; SOT188	257
PCF80C562WP	256	–	like PCB80C562; extended temperature range	68-PLCC; SOT188	307

Operating temperature range (PCB versions): −40 to + 70 °C.
(PCF versions): −40 to + 85 °C.
(PCD versions): −25 to + 70 °C.
Extended temperature range : −40 to + 85 °C.

type number	description			package code	page
CMOS					
	RAM	ROM			
PCF80C652P	256	–	like PCB80C652; extended temperature range	DIL-40; SOT129	353
PCF80C652WP	256	–	like PCB80C652; extended temperature range	44-PLCC; SOT187	353
PCF80C652H	256	–	like PCB80C652; extended temperature range	44-QFP; SOTXXX	353
PCF80C662P	256	–	like PCB80C662; extended temperature range	DIL-40; SOT129	391
PCF80C662WP	256	–	like PCB80C662; extended temperature range	44-PLCC; SOT187	391
PCF80C662H	256	–	like PCB80C662; extended temperature range	44-QFP; SOTXXX	391
PCF83C552WP	256	8K	like PCB83C552; extended temperature range	68-PLCC; SOT188	257
PCF83C562WP	256	8K	like PCB83C562; extended temperature range	68-PLCC; SOT188	307
PCF83C662P	256	8K	like PCB83C662; extended temperature range	DIL-40; SOT129	391
PCF83C662WP	256	8K	like PCB83C662; extended temperature range	44-PLCC; SOT187	391
PCF83C662H	256	8K	like PCB83C662; extended temperature range	44-QFP; SOTXXX	391
PCF83C654P	256	16K	like PCB83C654; extended temperature range	DIL-40; SOT129	397
PCF83C654WP	256	16K	like PCB83C654; extended temperature range	44-PLCC; SOT187	397
PCF83C654H	256	16K	like PCB83C654; extended temperature range	44-QFP; SOT205A	397
PCF84C00B	256	–	bond-out version PCF84CXX family	28/28 Piggy-back	593
PCF84C00T	256	–	bond-out version PCF84CXX family	VSO-56; SOT190	593
PCF84C12P	64	1K	mask-programmable ROM	DIL-20; SOT146	577
PCF84C12T	64	1K	mask-programmable ROM	SO20; SOT163A	577
PCF84C21P	64	2K	plus 8-bit LED driver; extended temperature range	DIL-28; SOT117	593
PCF84C21T	64	2K	plus 8-bit LED driver; extended temperature range	SO28; SOT136A	593
PCF84C22P	64	2K	mask-programmable ROM	DIL-20; SOT146	577
PCF84C22T	64	2K	mask-programmable ROM	SO20; SOT163A	577

Operating temperature range (PCF84CXX series): –40 to + 85 °C.
Extended temperature range : –40 to + 85 °C.

type number	description			package code	page
CMOS (continued)	RAM	ROM			
PCF84C41P	128	4K	plus 8-bit LED driver; extended temperature range	DIL-28; SOT117	593
PCF84C41T	128	4K	plus 8-bit LED driver; extended temperature range	SO28; SOT136A	593
PCF84C42P	64	4K	mask programmable ROM	DIL-20; SOT146	577
PCF84C42T	64	4K	mask programmable ROM	SO20; SOT163A	577
PCF84C81P	256	8K	plus 8-bit LED driver; extended temperature range	DIL-28; SOT117	593
PCF84C81T	256	8K	plus 8-bit LED driver; extended temperature range	SO28; SOT136A	593
PCF84C85P	256	8K	32 I/O; extended temperature range	DIL-40; SOT129	619
PCF84C85T	256	8K	32 I/O; extended temperature range	VSO-40; SOT158A	619
PCF84C430H	128	4K	with LCD driver; extended temperature range	64-QFP; SOT208	659
SC80C451	128	4K	ROM-less version SC83C451	N, A; 68-PLCC/64-DIP	241
SC83C451	128	4K	I/O expanded CMOS	N, A; 68-PLCC/64-DIP	241

Operating temperature range: −40 to + 85 °C.

MAINTENANCE TYPE LIST

The types listed below are not included in this handbook. Detailed information will be supplied on request.

PCB85C51	software development version of PCB80C51BH
PCB80C31/51BH	replaced by PCB80C31/51BH-3

TYPE DESIGNATION

PRO ELECTRON TYPE DESIGNATION CODE FOR INTEGRATED CIRCUITS

This type nomenclature applies to semiconductor monolithic, semiconductor multi-chip, thin-film, thick-film and hybrid integrated circuits.

A basic type number consists of:

THREE LETTERS FOLLOWED BY A SERIAL NUMBER

FIRST AND SECOND LETTER

1. DIGITAL FAMILY CIRCUITS

   The FIRST TWO LETTERS identify the FAMILY (see note 1).

2. SOLITARY CIRCUITS

   The FIRST LETTER divides the solitary circuits into:

   S : Solitary digital circuits
   T : Analogue circuits
   U : Mixed analogue/digital circuits

   The SECOND LETTER is a serial letter without any further significance except 'H' which stands for hybrid circuits (see note 3).

3. MICROPROCESSORS

   The FIRST TWO LETTERS identify microprocessors and correlated circuits as follows:

   MA : { Microcomputer / Central processing unit
   MB : Slice processor (see note 2)
   MD : Correlated memories
   ME : Other correlated circuits (interface, clock, peripheral controller, etc.)

4. CHARGE-TRANSFER DEVICES AND SWITCHED CAPACITORS

   The FIRST TWO LETTERS identify the following:

   NH : Hybrid circuits
   NL : Logic circuits
   NM : Memories
   NS : Analogue signal processing, using switched capacitors
   NT : Analogue signal processing, using CTDs
   NX : Imaging devices
   NY : Other correlated circuits

Notes

1. A logic family is an assembly of digital circuits designed to be interconnected and defined by its basic electrical characteristics (such as: supply voltage, power consumption, propagation delay, noise immunity).
2. By 'slice processor' is meant: a functional slice of microprocessor.
3. The first letter 'S' should be used for all solitary memories, to which, in the event of hybrids, the second letter 'H' should be added (e.g. SH for Bubble-memories).

THIRD LETTER

It indicates the operating ambient temperature range.
The letters A to G give information about the temperature:

A : temperature range not specified
B : 0 to + 70 °C
C : −55 to + 125 °C
D : −25 to + 70 °C
E : −25 to + 85 °C
F : −40 to + 85 °C
G : −55 to + 85 °C

If a circuit is published for another temperature range, the letter indicating a narrower temperature range may be used or the letter 'A'.

Example: the range 0 to + 75 °C can be indicated by 'B' or 'A'.

SERIAL NUMBER

This may be either a 4-digit number assigned by Pro Electron, or the serial number (which may be a combination of figures and letters) of an existing company type designation of the manufacturer.

To the basic type number may be added:

A VERSION LETTER

Indicates a minor variant of the basic type or the package. Except for 'Z', which means customized wiring, the letter has no fixed meaning. The following letters are recommended for package variants:

C : for cylindrical
D : for ceramic DIL
F : for flat pack
L : for chip on tape
P : for plastic DIL
Q : for QIL
T : for miniature plastic (mini-pack)
U : for uncased chip

Alternatively a TWO LETTER SUFFIX may be used instead of a single package version letter, if the manufacturer (sponsor) wishes to give more information.

FIRST LETTER: General shape

C : Cylindrical
D : Dual-in-line (DIL)
E : Power DIL (with external heatsink)
F : Flat (leads on 2 sides)
G : Flat (leads on 4 sides)
K : Diamond (TO-3 family)
M : Multiple-in-line (except Dual-, Triple-, Quadruple-in-line)
Q : Quadruple-in-line (QIL)
R : Power QIL (with external heatsink)
S : Single-in-line
T : Triple-in-line
W : Lead chip-carrier (LCC)
X : Leadless chip-carrier (LLCC)
Y : Pin grid array (PGA)

SECOND LETTER: Material

C : Metal-ceramic
G : Glass-ceramic (cerdip)
M : Metal
P : Plastic

A hyphen precedes the suffix to avoid confusion with a version letter.

RATING SYSTEMS

RATING SYSTEMS

The rating systems described are those recommended by the International Electrotechnical Commission (IEC) in its Publication 134.

DEFINITIONS OF TERMS USED

Electronic device. An electronic tube or valve, transistor or other semiconductor device.

Note
This definition excludes inductors, capacitors, resistors and similar components.

Characteristic. A characteristic is an inherent and measurable property of a device. Such a property may be electrical, mechanical, thermal, hydraulic, electro-magnetic, or nuclear, and can be expressed as a value for stated or recognized conditions. A characteristic may also be a set of related values, usually shown in graphical form.

Bogey electronic device. An electronic device whose characteristics have the published nominal values for the type. A bogey electronic device for any particular application can be obtained by considering only those characteristics which are directly related to the application.

Rating. A value which establishes either a limiting capability or a limiting condition for an electronic device. It is determined for specified values of environment and operation, and may be stated in any suitable terms.

Note
Limiting conditions may be either maxima or minima.

Rating system. The set of principles upon which ratings are established and which determine their interpretation.

Note
The rating system indicates the division of responsibility between the device manufacturer and the circuit designer, with the object of ensuring that the working conditions do not exceed the ratings.

ABSOLUTE MAXIMUM RATING SYSTEM

Absolute maximum ratings are limiting values of operating and environmental conditions applicable to any electronic device of a specified type as defined by its published data, which should not be exceeded under the worst probable conditions.

These values are chosen by the device manufacturer to provide acceptable serviceability of the device, taking no responsibility for equipment variations, environmental variations, and the effects of changes in operating conditions due to variations in the characteristics of the device under consideration and of all other electronic devices in the equipment.

The equipment manufacturer should design so that, initially and throughout life, no absolute maximum value for the intended service is exceeded with any device under the worst probable operating conditions with respect to supply voltage variation, equipment component variation, equipment control adjustment, load variations, signal variation, environmental conditions, and variations in characteristics of the device under consideration and of all other electronic devices in the equipment.

DESIGN MAXIMUM RATING SYSTEM

Design maximum ratings are limiting values of operating and environmental conditions applicable to a bogey electronic device of a specified type as defined by its published data, and should not be exceeded under the worst probable conditions.

These values are chosen by the device manufacturer to provide acceptable serviceability of the device, taking responsibility for the effects of changes in operating conditions due to variations in the characteristics of the electronic device under consideration.

The equipment manufacturer should design so that, initially and throughout life, no design maximum value for the intended service is exceeded with a bogey device under the worst probable operating conditions with respect to supply voltage variation, equipment component variation, variation in characteristics of all other devices in the equipment, equipment control adjustment, load variation, signal variation and environmental conditions.

DESIGN CENTRE RATING SYSTEM

Design centre ratings are limiting values of operating and environmental conditions applicable to a bogey electronic device of a specified type as defined by its published data, and should not be exceeded under normal conditions.

These values are chosen by the device manufacturer to provide acceptable serviceability of the device in average applications, taking responsibility for normal changes in operating conditions due to rated supply voltage variation, equipment component variation, equipment control adjustment, load variation, signal variation, environmental conditions, and variations in the characteristics of all electronic devices.

The equipment manufacturer should design so that, initially, no design centre value for the intended service is exceeded with a bogey electronic device in equipment operating at the stated normal supply voltage.

HANDLING MOS DEVICES

HANDLING MOS DEVICES

Though all our MOS integrated circuits incorporate protection against electrostatic discharges, they can nevertheless be damaged by accidental over-voltages. In storing and handling them, the following precautions are recommended.

Caution

Testing or handling and mounting call for special attention to personal safety. Personnel handling MOS devices should normally be connected to ground via a resistor.

Storage and transport

Store and transport the circuits in their original packing. Alternatively, use may be made of a conductive material or special IC carrier that either short-circuits all leads or insulates them from external contact.

Testing or handling

Work on a conductive surface (e.g. metal table top) when testing the circuits or transferring them from one carrier to another. Electrically connect the person doing the testing or handling to the conductive surface, for example by a metal bracelet and a conductive cord or chain. Connect all testing and handling equipment to the same surface.
Signals should not be applied to the inputs while the device power supply is off. All unused input leads should be connected to either the supply voltage or ground.

Mounting

Mount MOS integrated circuits on printed circuit boards *after* all other components have been mounted. Take care that the circuits themselves, metal parts of the board, mounting tools, and the person doing the mounting are kept at the same electric (ground) potential. If it is impossible to ground the printed-circuit board the person mounting the circuits should touch the board before bringing MOS circuits into contact with it.

Soldering

Soldering iron tips, including those of low-voltage irons, or soldering baths should also be kept at the same potential as the MOS circuits and the board.

Static charges

Dress personnel in clothing of non-electrostatic material (no wool, silk or synthetic fibres). After the MOS circuits have been mounted on the board proper handling precautions should still be observed. Until the sub-assemblies are inserted into a complete system in which the proper voltages are supplied, the board is no more than an extension of the leads of the devices mounted on the board. To prevent static charges from being transmitted through the board wiring to the device it is recommended that conductive clips or conductive tape be put on the circuit board terminals.

Transient voltages

To prevent permanent damage due to transient voltages, do not insert or remove MOS devices, or printed-circuit boards with MOS devices, from test sockets or systems with power on.

Voltage surges

Beware of voltage surges due to switching electrical equipment on or off, relays and d.c. lines.

SINGLE-CHIP 8-BIT MICROCONTROLLERS (NMOS, CMOS)

MAB8031AH; MAB8051AH 31
MAB8032AH; MAB8052AH 65
MAB8048H/35HL; MAB8049H/39HL; MAB8050H/40HL 99
MAB84X1; MAF84X1; MAF84AX1 family 125
MAB8422/42; MAF8422/42; MAF84A22/42 157
PCB80C31BH-3; PCB80C51BH-3 177
PCB80C39; PCB80C49 209
SC83C451; SC80C451 241
PCB83C552; PCB80C552 257
PCB83C562; PCB80C562 307
PCB83C652; PCB80C652 353
PCB83C662; PCB80C662 391
PCB83C654; PCB80C654 397
PCB83C851; PCB80C851 439
PCD3315C 471
PCD3343 475
PCD3349 515
PCF84CXXX family spec 551
PCF84C12/22/42 577
PCF84C21/41/81 593
PCF84C85 619
PCF84C430 659

SUPERSEDES DATA OF JANUARY 1987

SINGLE-CHIP 8-BIT MICROCONTROLLER

DESCRIPTION

The MAB8051AH-2 family of single-chip 8-bit microcontrollers is manufactured in an advanced 2 μ NMOS process. The family consists of the following members:

- MAB8031AH-2: ROM-less version of the MAB8051AH-2
- MAB8051AH-2: 4 K bytes mask-programmable ROM, 128 bytes RAM

Both types are available in 8, 10 and 12 MHz versions and 15 MHz for the MAB8031AH-2. In the following, the generic term "MAB8051AH-2" is used to refer to both family members.

The device provides hardware features, architectural enhancements and new instructions to function as a controller for applications requiring up to 64 K bytes of program memory and/or up to 64 K bytes of data storage.

The MAB8051AH-2 contains a non-volatile 4 K x 8 read-only program memory (not ROM-less version); a volatile 128 x 8 read/write data memory; 32 I/O lines; two 16-bit timer/event counters; a five-source, two-priority-level, nested interrupt structure; a serial I/O power for either multi-processor communications, I/O expansion, or full duplex UART; and on-chip oscillator and timing circuits. For systems that require extra capability, the MAB8051AH-2 can be expanded using standard TTL compatible memories and logic.

The device also functions as an arithmetic processor having facilities for both binary and BCD arithmetic plus bit-handling capabilities. The instruction set consists of 255 instructions; 44% one-byte, 41% two-byte and 15% three-byte. With a 12 MHz crystal, 58% of the instructions are executed in 1 μs and 40% in 2 μs. Multiply and divide instructions require 4 μs. Multiply, divide, subtract and compare are among the many instructions added to the standard MAB8048H instruction set.
For further detailed information see users manual 'Single-chip 8-bit microcontrollers'.

Features

- 4 K x 8 ROM (MAB8051AH-2 only), 128 x 8 RAM
- Four 8-bit ports, 32 I/O lines
- Two 16-bit timer/event counters
- Full duplex serial port
- External memory expandable to 128 K
- Boolean processing
- 218 bit-addressable locations
- On-chip oscillator
- Five-source interrupt structure with two priority levels
- 58% of instructions executed in 1 μs; multiply and divide in 4 μs (at 12 MHz clock)
- Enhanced architecture with:
  - non-page-oreinted instructions
  - direct addressing
  - four 8-bit register banks
  - stack depth up to 128-bytes
  - multiply, divide, subtract and compare
- Available with extended temperature range: –40 to + 85 °C (MAF8031/51AH-2)
- Available with automotive temperature range: –40 to + 100 °C (MAF80A31/51AH-2)

PACKAGE OUTLINES

MAB8031/51AHP; MAF8031/51AHP; MAF80A31/51AHP: 40-lead DIL; plastic (SOT-129).
MAB8031/51AHWP; MAF8031/51AHWP; MAF80A31/51AHWP: 44-lead, plastic leaded-chip-carrier. 'pocket' version (PLCC); SOT187AA.

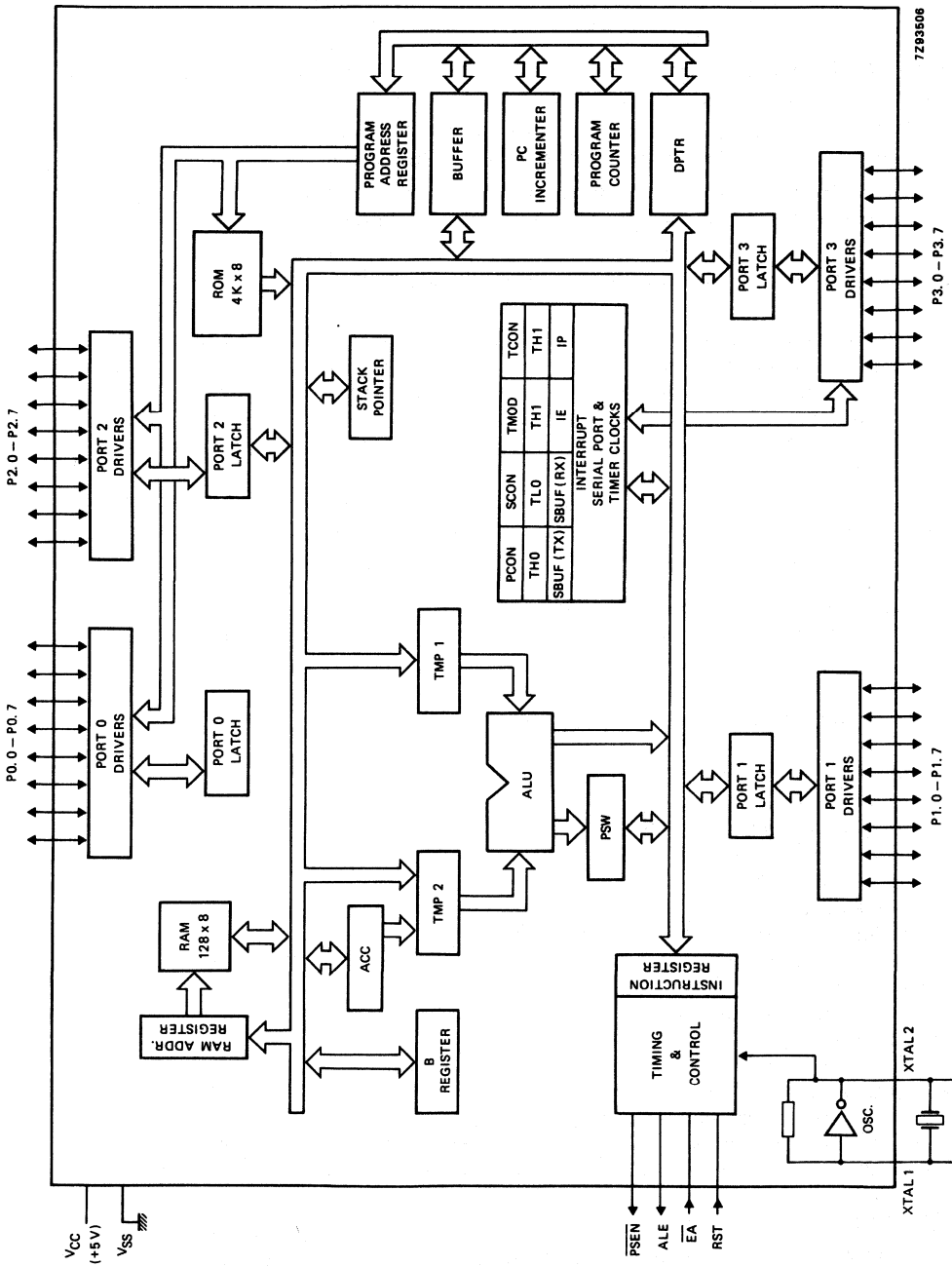


Fig. 1 Block diagram.

DEVELOPMENT DATA

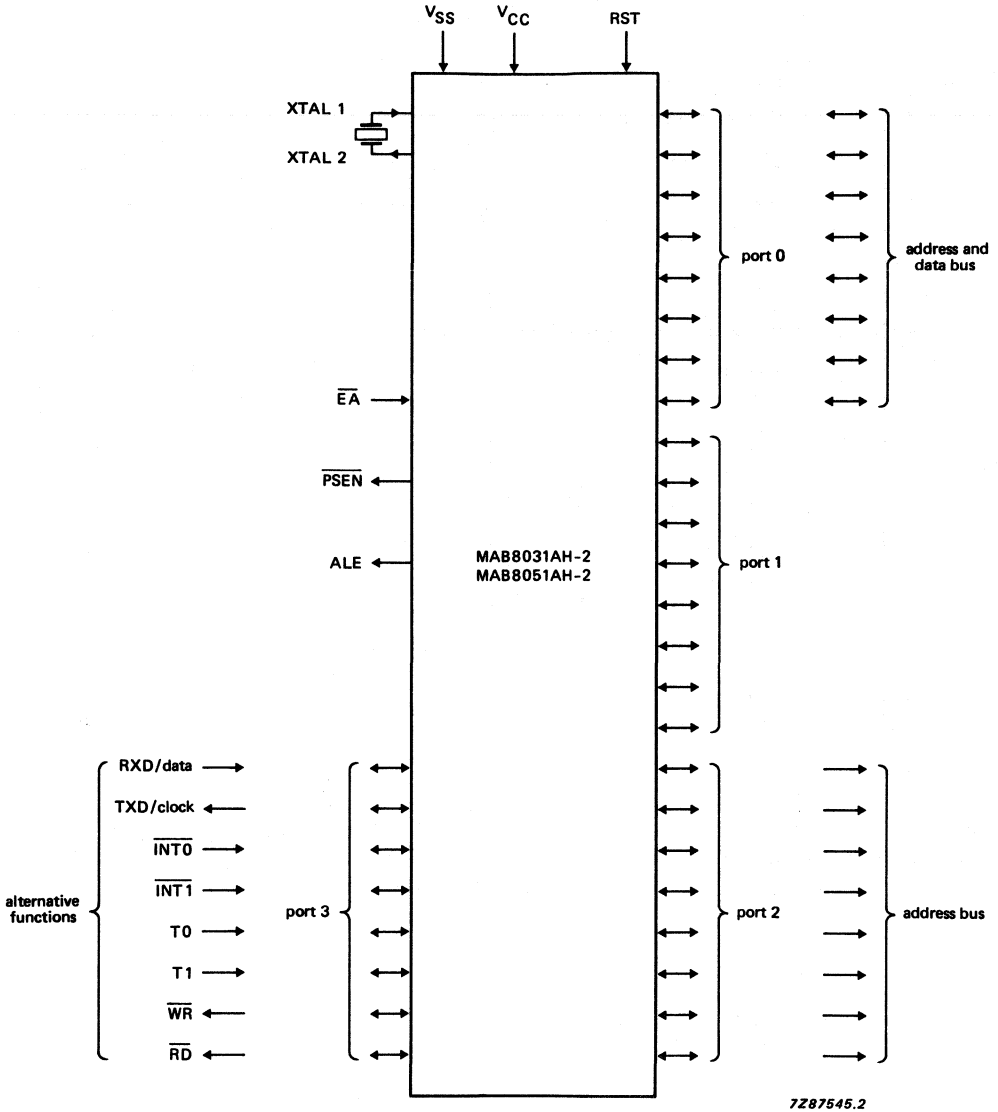


Fig. 2 Functional diagram.

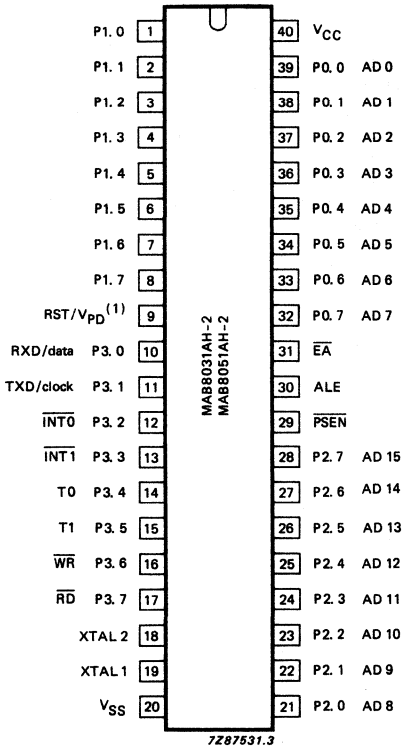


Fig. 3a Pinning diagram for MAB8031/51AHP-2; MAF8031/51AHP-2; MAF80A31/51AHP-2.

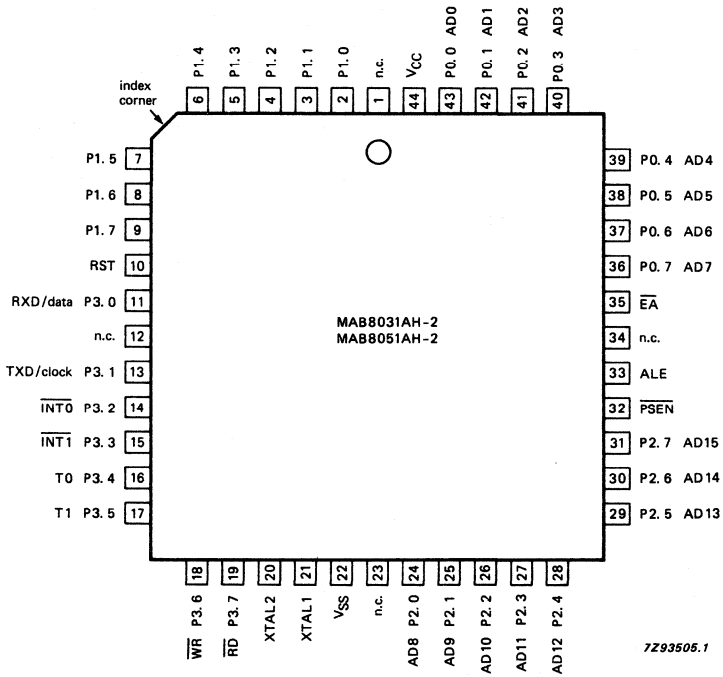


Fig. 3b Pinning diagram for MAB8031/51AHWP-2; MAF8031/51AHWP-2; MAF80A31/51AHWP-2.

(1) V_{PD} option for MAB8051AH only available on request.

PINNING (DIL package)

1-8	P1.0-P1.7	**Port 1:** 8-bit quasi-bidirectional I/O port. It receives the low-order address byte during program verification. Port 1 can sink/source one TTL (= 4 LS TTL) input. It can drive MOS inputs without external pull-ups.
9		**RST/V_{PD}**: a high level on this pin for two machine cycles while the oscillator is running resets the device. An internal pulldown permits Power-On reset using only a capacitor connected to V_{CC}. As an available option, this pin also supplies standby power to the RAM: V_{PD} should be held within its specified limit while V_{CC} drops below its specified limit. When V_{PD} is LOW the RAM current is drawn from V_{CC} (PD mode for MAB8051AH-2 only).
10-17	P3.0-P3.7	**Port 3:** 8-bit quasi-bidirectional I/O port with internal pull-ups. It also serves the following alternative functions:

Port pin	*Alternative function*
P3.0	**RXD/data:** serial port receiver data input (asynchronous) or data input/output (synchronous)
P3.1	**TXD/clock:** serial port transmitter data output (asynchronous) or clock output (synchronous)
P3.2	**$\overline{\text{INT0}}$:** external interrupt 0 or gate control input for timer/event counter 0
P3.3	**$\overline{\text{INT1}}$** : external interrupt 1 or gate control input for timer/event counter 1
P3.4	**T0** : external input for timer/event counter 0
P3.5	**T1** : external input for timer/event counter 1
P3.6	**$\overline{\text{WR}}$:** external data memory write strobe
P3.7	**$\overline{\text{RD}}$** : external data memory read strobe.

Operation of an alternative function is determined by the relevant output latch programmed to logic 1. Port 3 can sink/source one TTL input. It can drive MOS inputs without external pull-ups.

18	XTAL 2	**Crystal input 2:** output of the inverting amplifier that forms the oscillator, and input to the internal clock generator. Receives the external oscillator signal when an external oscillator is used (see figures 6 and 7).
19	XTAL 1	**Crystal input 1:** input to the inverting amplifier that forms the oscillator. Connected to V_{SS} when an external oscillator is used (see figures 6 and 7).
20	V_{SS}	**Ground:** circuit ground potential.
21-28	P2.0-P2.7	**Port 2:** 8-bit quasi-bidirectional I/O port with internal pull-ups. It emits the high-order address byte when accessing external memory. It also receives the high-order address bits and control signals during program verification. Port 2 can sink/source one TTL input. It can drive MOS inputs without external pull-ups.
29	$\overline{\text{PSEN}}$	**Program Store Enable output:** read strobe to the external Program Memory. It is activated twice each machine cycle during fetches from external Program Memory. When executing out of external Program Memory two activations of $\overline{\text{PSEN}}$ are skipped during each access to external Data Memory. $\overline{\text{PSEN}}$ is not activated (remains HIGH) during fetches from internal Program Memory.
30	ALE	**Address Latch Enable output:** latches the low byte of the address during accesses to external memory in normal operation. It is activated every six oscillator periods except during an external data memory access.

DEVELOPMENT DATA

PINNING (continued)

31	$\overline{EA}$	When $\overline{EA}$ is held at a TTL high level the CPU executes out of the internal Program Memory (ROM), provided the Program Counter is less than 4096. When $\overline{EA}$ is held at a TTL low level the CPU executes out of external Program Memory. $\overline{EA}$ does not float.
32-39	P0.7-P0.0	**Port 0**: 8-bit open drain bidirectional I/O port. It is also the multiplexed low-order address and data bus during accesses to external memory (during these accesses it activates internal pull-ups). It also outputs instruction bytes during program verification. (External pull-ups are required during program verification). Port 0 can sink/source two TTL inputs.
40	V_{CC}	**Power Supply**: + 5 V power supply pin during normal operation.

FUNCTIONAL DESCRIPTION

General

The MAB8051AH-2 is a stand-alone high-performance microcontroller designed for use in real-time applications such as instrumentation, industrial control and intelligent computer peripherals.

The device provides hardware features, architectural enhancements and new instructions to function as a controller for applications requiring up to 64 K bytes of program memory and/or up to 64 K bytes of data storage.

The MAB8031AH-2 is a control-oriented CPU without on-chip program memory. It can address 64 K bytes of external program memory in addition to 64 K bytes of external data memory. The MAB8051AH-2 is a MAB8031AH-2 with the lower 4 K bytes of program memory filled with on-chip mask programmable ROM. For systems requiring extra capability, the MAB8051AH-2 can be expanded using standard TTL memories and peripherals.

The two pin-compatible versions of this component reduce development problems to a minimum and provide maximum flexibility. The MAB8051AH-2 is for low-cost, high volume production; and the MAB8031AH-2 for applications requiring the flexibility of external program memory which can be easily modified and updated in the field.

The MAB8051AH-2 contains a non-volatile 4 K x 8 read-only program memory; a volatile 128 x 8 read/write data memory; 32 I/O lines; two 16-bit timer/event counters; a five-source, two-priority-level, nested interrupt structure; a serial I/O port for either multi-processor communications, I/O expansion, or full duplex UART; and on-chip oscillator and timing circuits.

Central processing unit

The central processing unit (CPU) manipulates operands in four memory spaces. These are the 64 K-byte external data memory, 256-byte internal data memory and 16-bit program counter spaces. The internal data memory address space is sub-divided into the 128-byte internal data RAM and 128-byte special function register (SFR) address spaces, as shown in Fig. 4a. See also Figures 4b to 4f.

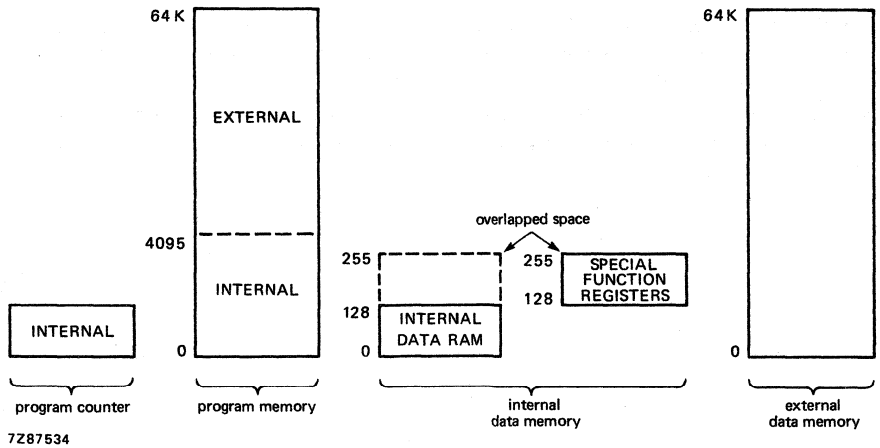


Fig. 4a Memory map.

Where:

The internal data memory locations are addressable direct/indirect as follows:

location	addressed
RAM 0 to 127	direct and indirect
SFR 128 to 255	direct only

FUNCTIONAL DESCRIPTION (continued)

The internal data RAM contains four register banks (each with eight registers), 128 addressable bits, and the stack. The stack depth is limited by the available internal data RAM and its location is determined by the 8-bit stack pointer. All registers except the program counter and the four 8-register banks reside in the special function register (SFR) address space. These memory mapped registers include arithmetic registers, pointers, I/O ports, interrupt system registers, timers and serial port. There are 128 addressable bit locations in the SFR address space.

The MAB8051AH-2 contains 128 bytes of internal data RAM and 20 special function registers. It provides a non-paged program memory address space to accommodate relocatable code. Conditional branches are performed relative to the program counter. The register-indirect jump permits branching relative to a 16-bit base register with an offset provided by an 8-bit index register. 16-bit jumps and calls permit branching to any location in the contiguous 64 K program memory address space.

The MAB8051AH-2 has five methods for addressing source operands:

- Register
- Direct
- Register-Indirect
- Immediate
- Base-Register-plus Index-Register-Indirect

The first three methods can be used for addressing destination operands. Most instructions have a "destination source" field that specifies the data type, addressing methods and operands involved. For operations other than moves, the destination operand is also a source operand.

Access addressing is as follows:

- Registers in the four 8-register banks through Register, Direct, or Register-Indirect.
- 128 bytes of internal data RAM through Direct or Register-Indirect.
- Special function registers through Direct.
- External data memory through Register-Indirect.
- Program memory look-up tables through Base-Register-plus Index-Register-Indirect.

The MAB8051AH-2 is classified as an 8-bit device since the internal ROM, RAM, Special Function Registers (SFR), Arithmetic Logic Unit (ALU), and external data bus are each 8-bits wide. It performs operations on bit, nibble, byte and double-byte data types.

Facilities are available for byte transfer, logic, and integer arithmetic operations. Data transfer, logic, and conditional branch operations can be performed directly on Boolean variables to provide excellent bit handling.

DEVELOPMENT DATA

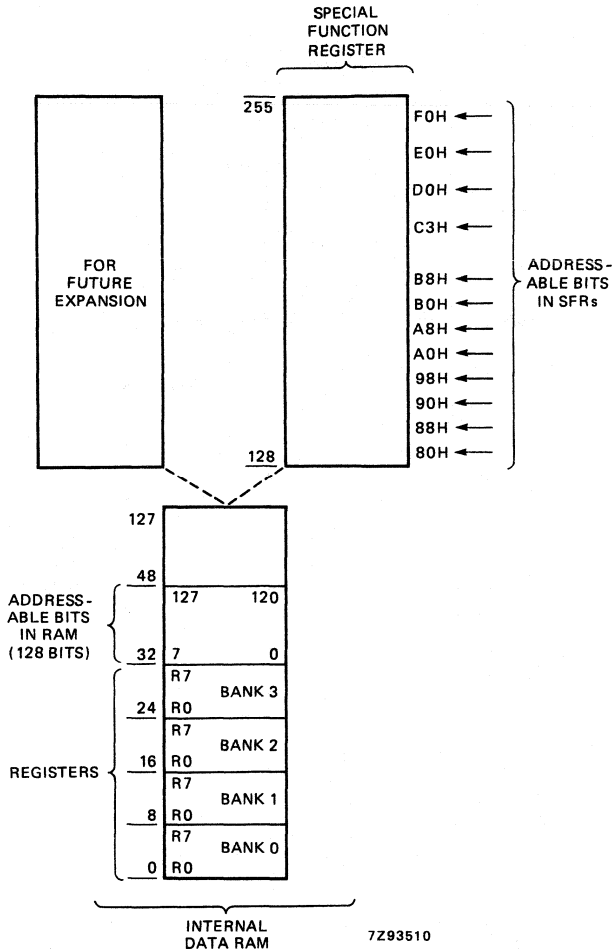


Fig. 4b Internal data memory address space.

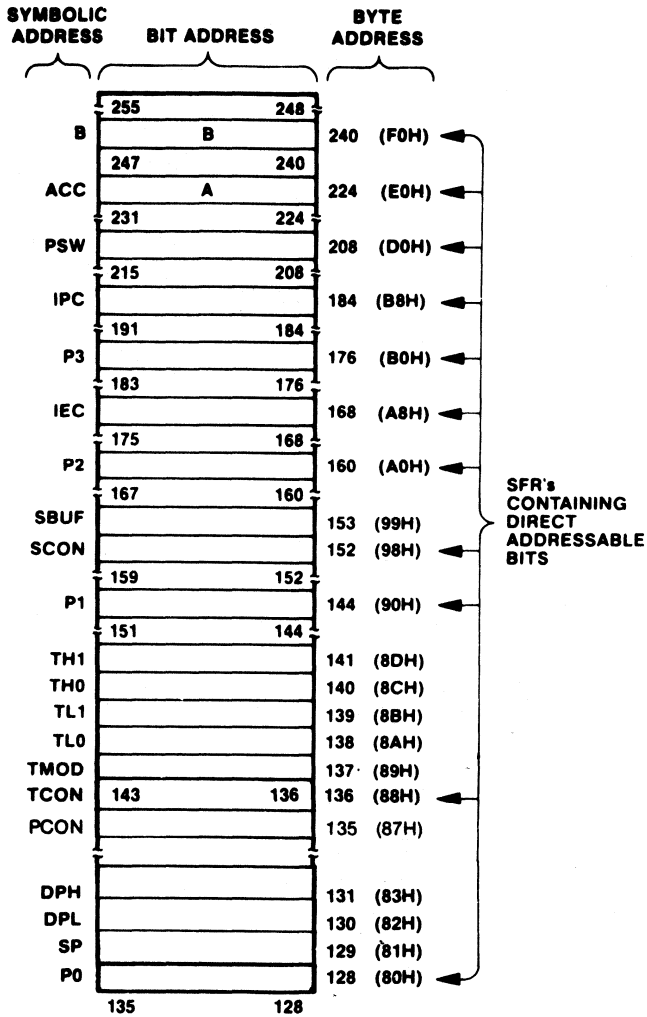


Fig. 4c Mapping of special function registers.

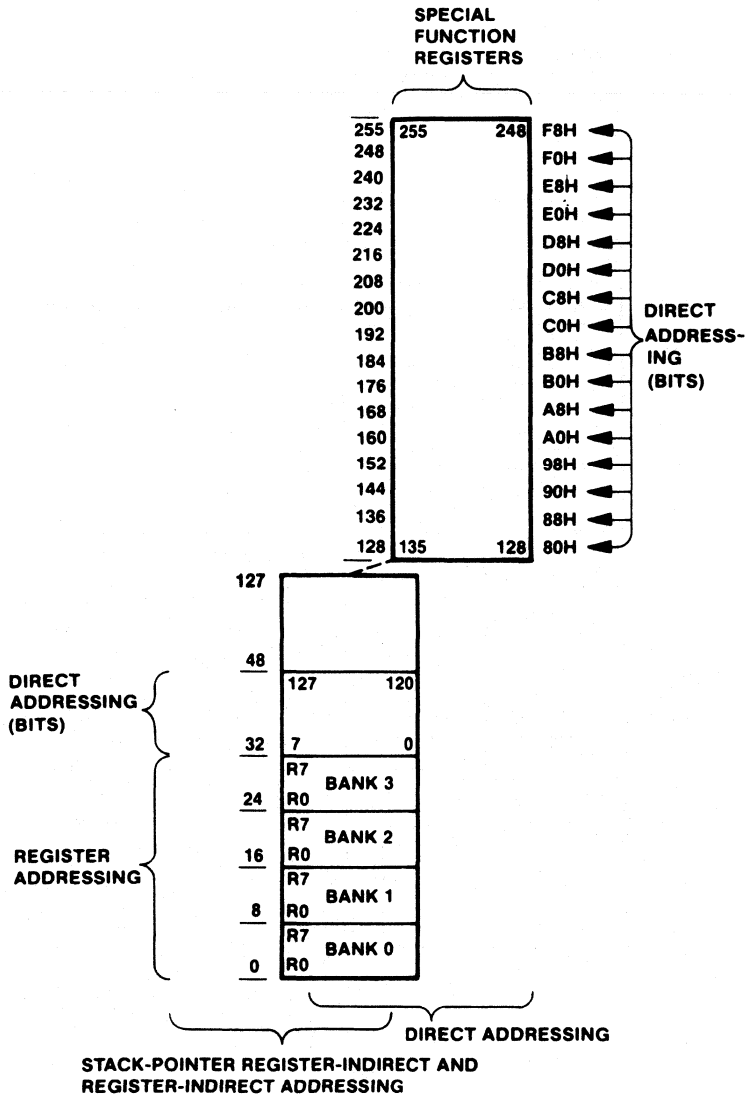


Fig. 4d Special function register bit addresses.

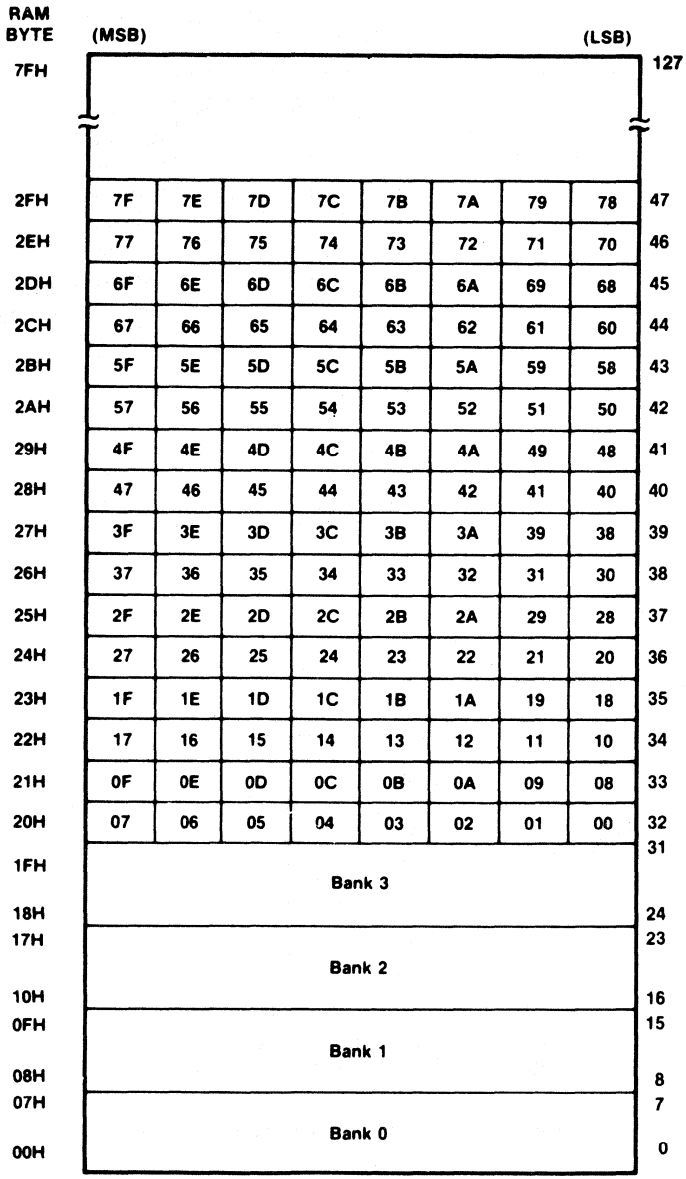


Fig. 4e RAM bit addresses.

DEVELOPMENT DATA

Direct Byte Address	(MSB)			Bit Addresses				(LSB)	Hardware Register Symbol
240	F7	F6	F5	F4	F3	F2	F1	F0	B
224	E7	E6	E5	E4	E3	E2	E1	E0	ACC
	CY	AC	FO	RS1	RS0	OV		P	
208	D7	D6	D5	D4	D3	D2	D1	D0	PSW
				PS	PT1	PX1	PT0	PX0	
184	—	—	—	BC	BB	BA	B9	B8	IP
176	B7	B6	B5	B4	B3	B2	B1	B0	P3
	EA			ES	ET1	EX1	ET0	EX0	
168	AF	—	—	AC	AB	AA	A9	A8	IE
160	A7	A6	A5	A4	A3	A2	A1	A0	P2
	SMO	SM1	SM2	REN	TB8	RB8	TI	RI	
152	9F	9E	9D	9C	9B	9A	99	98	SCON
144	97	96	95	94	93	92	91	90	P1
	TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0	
136	8F	8E	8D	8C	8B	8A	89	88	TCON
128	87	86	85	84	83	82	81	80	P0

Fig. 4f Addressing operands in internal data memory.

FUNCTIONAL DESCRIPTION (continued)

I/O facilities

The MAB8051AH-2 has 32 I/O lines treated as 32 individual addressable bits and as four parallel 8-bit addressable ports. Ports 0, 1, 2 and 3 perform the following alternate functions:

- Port 0; provides the multiplexed low-order address and data bus used for expanding the MAB8051AH-2 with standard memories and peripherals
- Port 2; provides the high-order address bus when expanding the MAB8051AH-2 with external program memory or more than 256 bytes of external data memory
- Port 3; pins can be configured individually to provide:-

  external interrupt requests inputs
  counter inputs
  serial port receiver input and transmitter output
  control signals to READ and WRITE to external data memory

The generation or use of a Port 3 pin as an alternate function is carried out automatically by the MAB8051AH-2 provided the pin is configured as an output.

Timer/event counters

The MAB8051AH-2 contains two 16-bit registers, Timer 0 and Timer 1, that can be used as timers or event counters to carry out the following functions:

- Measure time intervals and pulse durations
- Count events
- Generate interrupt requests

Each timer/event counter can be programmed independently to operate in three modes:

- Mode 0; 8-bit timer or 8-bit counter each with divide by 32 prescaler
- Mode 1; 16-bit time-interval or event counter
- Mode 2; 8-bit time-interval or event counter with automatic reload upon overflow

Counter 0 can be programmed to operate in an additional mode as follows:

- Mode 3; one 8-bit time-interval or event counter and one 8-bit time-interval counter

When counter 0 is in Mode 3, counter 1 can be programmed to operate in Modes 0, 1 or 2 but cannot set an interrupt request flag or generate an interrupt. However the overflow from counter 1 can be used to pulse the serial Port transmission-rate generator.

The frequency handling range of these counters with a 3,5 to 12 MHz crystal is as follows:

- 0,3 to 1 MHz when programmed for an input that is a division by 12 of the oscillator frequency
- 0 Hz to an upper limit of 150 kHz to 0,5 MHz when programmed for external inputs

Both internal and external inputs can be gated to the counter by a second external source for directly measuring pulse durations.

The counters are started and stopped under software control. Each one sets its interrupt request flag when it overflows from all 1's to all 0's (or automatic reload value).

On-chip peripheral functions

In addition to the CPU and memories, an interrupt system, extensive I/O facilities, and several peripheral functions are integrated on-chip to relieve the CPU of repetitious, complicated or time-critical tasks and to permit stringent real-time control of external system interfaces. The I/O facilities include the I/O pins, parallel ports, bidirectional address/data bus and the serial port for I/O expansion. The CPU peripheral functions integrated on-chip are the two 16-bit timer/event counters and the serial port.

Interrupt system (see Fig. 5)

External events and the real-time-driven on-chip peripherals require service by the CPU asynchronous to the execution of any particular section of code. To tie the asynchronous activities of these functions to normal program execution a multiple-source, two-priority-level, nested interrupt system is provided. Interrupt response latency is from 3 μs to 7 μs when using a 12 MHz crystal.

The MAB8051AH-2 acknowledge interrupt requests from five sources as follows:

- $\overline{\text{INT0}}$ and $\overline{\text{INT1}}$; externally via pins 12 and 13 respectively
- Timer 0 and Timer 1; from the two internal counters
- Serial Port; from the internal serial I/O port

Each interrupt vectors to a separate location in program memory for its service program.
Each source can be individually enabled or disabled and can be programmed to a high or low priority level. Also all enabled sources can be globally disabled or enabled. Both external interrupts can be programmed to be level-activated or transition-activated and is active LOW to allow "wire-ORing" of several interrupt sources to the input pin.

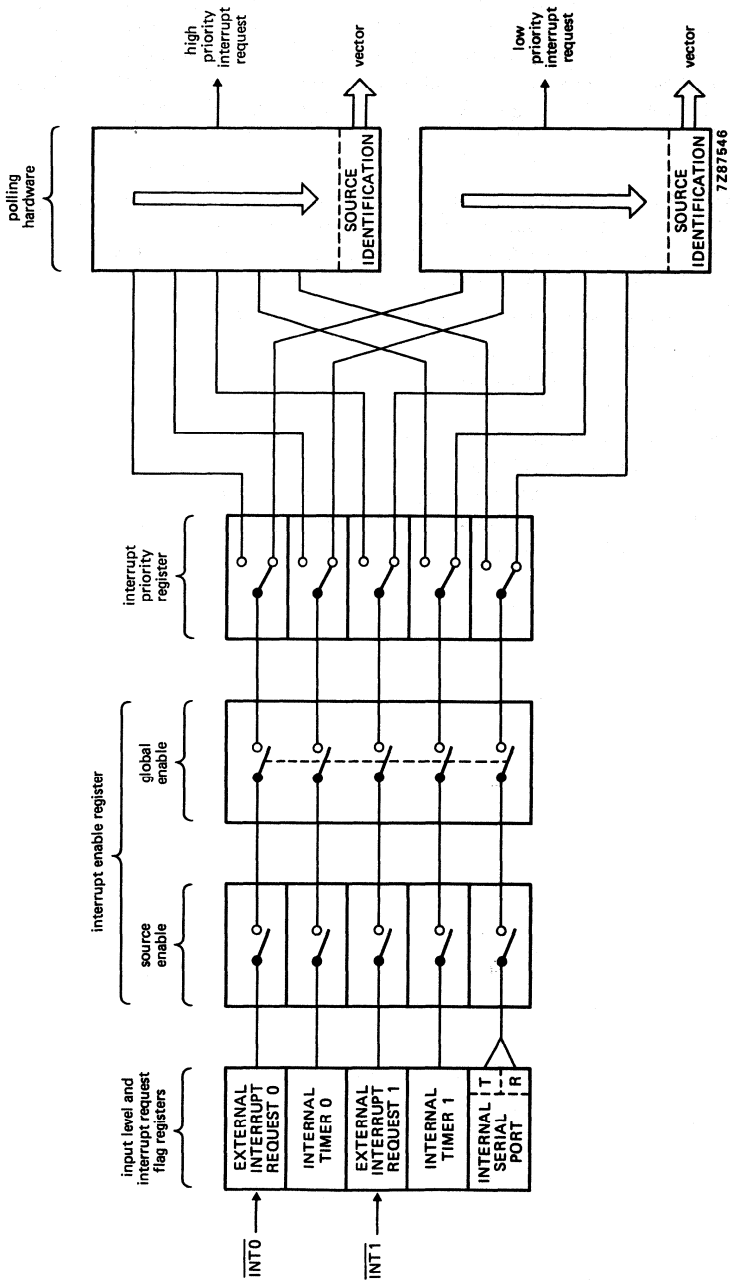


Fig. 5 Interrupt system.

OSCILLATOR CIRCUITRY

The oscillator circuitry of the MAB8051AH-2 is a single-stage inverting amplifier in a Pierce oscillator configuration. The circuitry has a combination of depletion and enhancement mode MOS FETs to produce the inverting characteristics, and not passive components. Either a crystal or ceramic resonator can be used as the feedback element to complete the oscillator circuitry. XTAL1, pin 19, is the high gain amplifier input, and XTAL2, pin 18, is the output (see Fig. 6).

To drive the MAB8051AH-2 externally, XTAL1 should be connected to ground and XTAL2 driven from an external source (see Fig. 7).

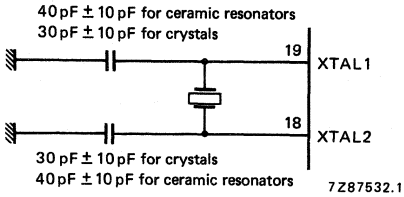


Fig. 6 Oscillator circuit.

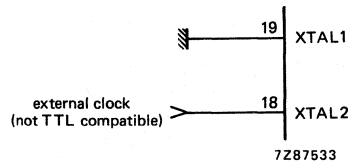


Fig. 7 Driving from an external source.

DEVELOPMENT DATA

RESET CIRCUITRY

The reset circuitry for the MAB8051AH-2 is connected to the reset pin, RST, as shown in Fig. 8. A Schmitt trigger is used th the input for noise rejection. The output of the Schmitt trigger is sampled by the reset circuitry every machine cycle.

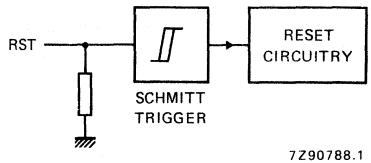


Fig. 8 Reset configuration at RST.

A reset is accomplished by holding the RST pin HIGH for at least two machine cycles (24 oscillator periods), while the oscillator is running. The CPU responds by executing an internal reset. It also configures the ALE and $\overline{PSEN}$ pins as inputs. (They are quasi-bidirectional.) The internal reset is executed during the second cycle in which RST is high and is repeated every cycle until RST goes LOW. It leaves the internal registers as follows:

REGISTER	CONTENT
PC	0000H
ACC	00H
B	00H
PSW	00H
SP	07H
DPTR	0000H
P0 -- P3	FFH
IP	XXX00000B
IE	XXX00000B
TMOD	00H
TCON	00H
TH0	00H
TL0	00H
TH1	00H
TL1	00H
SCON	00H
SBUF	Indeterminate
PCON	0XXXXXXXB

Where

H = Hexadecimal
B = Binary

The internal RAM is not affected by reset. When VCC is turned on, the RAM content is indeterminate.

Power-on reset

An automatic reset when V_{CC} is turned on can be obtained by connecting the RST pin to V_{CC} through a 10 μF capacitor, as long as the V_{CC} rise-time does not exceed 10 milliseconds. This power-on reset circuit is shown in Fig. 9. When the power is switched on, the current drawn by RST is the difference between V_{CC} and the capacitor voltage, and decreases from V_{CC} as the capacitor charges. The larger the capacitor, the more slowly V_{RST} decreases. V_{RST} must remain above the lower threshold of the Schmitt trigger long enough to effect a complete reset. The time required is the oscillator start-up time, plus 2 machine cycles.

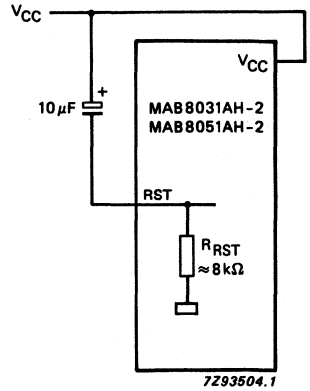


Fig. 9 Power-on reset.

INSTRUCTION SET

The MAB8051AH-2 uses a powerful instruction set to allow expansion of on-chip CPU peripherals and to optimize byte efficiency and execution speed. Reassigned opcodes add new high-power operations and permit new addressing modes to make old operations more orthogonal. The instruction set consists of 49 single-byte, 45 two-byte and 17 three-byte instructions. When using a 12 MHz oscillator, 64 instructions execute in 1 μs and 45 instructions execute in 2 μs. Multiply and divide instructions execute in 4 μs.

Table 1 Instruction set description

mnemonic		description	bytes/ cycles	opcode (hex.)
Arithmetic operation				
ADD	A,Rr	Add register to A	1 1	2*
ADD	A,direct	Add direct byte to A	2 1	25
ADD	A,@Ri	Add indirect RAM to A	1 1	26, 27
ADD	A,#data	Add immediate data to A	2 1	24
ADDC	A,Rr	Add register to A with carry flag	1 1	3*
ADDC	A,direct	Add direct byte to A with carry flag	2 1	35
ADDC	A,@Ri	Add indirect RAM to A with carry flag	1 1	36, 37
ADDC	A,#data	Add immediate data to A with carry flag	2 1	34
SUBB	A,Rr	Subtract register from A with borrow	1 1	9*
SUBB	A,direct	Subtract direct byte from A with borrow	2 1	95
SUBB	A,@Ri	Subtract indirect RAM from A with borrow	1 1	96, 97
SUBB	A,#data	Subtract immediate data from A with borrow	2 1	94
INC	A	Increment A	1 1	04
INC	Rr	Increment register	1 1	0*
INC	direct	Increment direct byte	2 1	05
INC	@Ri	Increment indirect RAM	1 1	06, 07
DEC	A	Decrement A	1 1	14
DEC	Rr	Decrement register	1 1	1*
DEC	direct	Decrement direct byte	2 1	15
DEC	@Ri	Decrement indirect RAM	1 1	16, 17
INC	DPTR	Increment data pointer	1 2	A3
MUL	AB	Multiply A & B	1 4	A4
DIV	AB	Divide A by B	1 4	84
DA	A	Decimal adjust A	1 1	D4

mnemonic		description	bytes/ cycles	opcode (hex.)
Logic operations				
ANL	A,Rr	AND register to A	1 1	5*
ANL	A,direct	AND direct byte to A	2 1	55
ANL	A,@Ri	AND indirect RAM to A	1 1	56, 57
ANL	A,#data	AND immediate data to A	2 1	54
ANL	direct,A	AND A to direct byte	2 1	52
ANL	direct,#data	AND immediate data to direct byte	3 2	53
ORL	A,Rr	OR register to A	1 1	4*
ORL	A,direct	OR direct byte to A	2 1	45
ORL	A,@Ri	OR indirect RAM to A	1 1	46, 47
ORL	A,#data	OR immediate data to A	2 1	44
ORL	direct,A	OR A to direct byte	2 1	42
ORL	direct,#data	OR immediate data to direct byte	3 2	43
XRL	A,Rr	Exclusive-OR register to A	1 1	6*
XRL	A,direct	Exclusive-OR direct byte to A	2 1	65
XRL	A,@Ri	Exclusive-OR indirect RAM to A	1 1	66, 67
XRL	A,#data	Exclusive-OR immediate data to A	2 1	64
XRL	direct, A	Exclusive-OR to direct byte	2 1	62
XRL	direct,#data	Exclusive-OR immediate data to direct byte	3 2	63
CLR	A	Clear A	1 1	E4
CPL	A	Complement A	1 1	F4
RL	A	Rotate A left	1 1	23
RLC	A	Rotate A left through the carry flag	1 1	33
RR	A	Rotate A right	1 1	03
RRC	A	Rotate A right through the carry flag	1 1	13
SWAP	A	Swap nibbles within A	1 1	C4

INSTRUCTION SET (continued)

mnemonic		description	bytes/ cycles	opcode (hex.)
Data transfer				
MOV*	A,Rr	Move register to A	1 1	E*
MOV	A,direct	Move direct byte to A	2 1	E5
MOV	A,@Ri	Move indirect RAM to A	1 1	E6, E7
MOV	A,#data	Move immediate data to A	2 1	74
MOV	Rr,A	Move A to register	1 1	F*
MOV	Rr,direct	Move direct byte to register	2 2	A*
MOV	Rr,#data	Move immediate data to register	2 1	7*
MOV	direct,A	Move A to direct byte	2 1	F5
MOV	direct,Rr	Move register to direct byte	2 2	8*
MOV	direct,direct	Move direct byte to direct	3 2	85
MOV	direct,@Ri	Move indirect RAM to direct byte	2 2	86, 87
MOV	direct,#data	Move immediate data to direct byte	3 2	75
MOV	@Ri,A	Move A to indirect RAM	1 1	F6, F7
MOV	@Ri,direct	Move direct byte to indirect RAM	2 2	A6, A7
MOV	@Ri,#data	Move immediate data to indirect RAM	2 1	76, 77
MOV	DPTR,#data16	Load data pointer with a 16-bit constant	3 2	90
MOVC	A,@A+DPTR	Move code byte relative to DPTR to A	1 2	93
MOVC	A,@A+PC	Move code byte relative to PC to A	1 2	83
MOVX	A,@Ri	Move external RAM (8-bit address) to A	1 2	E2, E3
MOVX	A,@DPTR	Move external RAM (16-bit address) to A	1 2	E0
MOVX	@Ri,A	Move A to external RAM (8-bit address)	1 2	F2, F3
MOVX	@DPTR,A	Move A to external RAM (16-bit address)	1 2	F0
PUSH	direct	Push direct byte onto stack	2 2	C0
POP	direct	Pop direct byte from stack	2 2	D0
XCH	A,Rr	Exchange register with A	1 1	C*
XCH	A,direct	Exchange direct byte with A	2 1	C5
XCH	A,@Ri	Exchange indirect RAM with A	1 1	C6, C7
XCHD	A,@Ri	Exchange LOW-order digit indirect RAM with A	1 1	D6, D7

* MOV A,ACC not valid instruction.

mnemonic		description	bytes/ cycles	opcode (hex.)
Boolean variable manipulation				
CLR	C	Clear carry flag	1 1	C3
CLR	bit	Clear direct bit	2 1	C2
SETB	C	Set carry flag	1 1	D3
SETB	bit	Set direct bit	2 1	D2
CPL	C	Complement carry flag	1 1	B3
CPL	bit	Complement direct bit	2 1	B2
ANL	C,bit	AND direct bit to carry flag	2 2	82
ANL	C,/bit	AND complement of direct bit to carry flag	2 2	B0
ORL	C,bit	OR direct bit to carry flag	2 2	72
ORL	C,/bit	OR complement of direct bit to carry flag	2 2	A0
MOV	C,bit	Move direct bit to carry flag	2 1	A2
MOV	bit,C	Move carry flag to direct bit	2 2	92
Program and machine control				
ACALL	addr11	Absolute subroutine call	2 2	•1addr
LCALL	addr16	Long subroutine call	3 2	12
RET		Return from subroutine	1 2	22
RETI		Return from interrupt	1 2	32
AJMP	addr11	Absolute jump	2 2	▲1addr
LJMP	addr16	Long jump	3 2	02
SJMP	rel	Short jump (relative address)	2 2	80
JMP	@A+DPTR	Jump indirect relative to the DPTR	1 2	73
JZ	rel	Jump if A is zero	2 2	60
JNZ	rel	Jump if A is not zero	2 2	70
JC	rel	Jump if carry flag is set	2 2	40
JNC	rel	Jump if no carry flag	2 2	50
JB	bit,rel	Jump if direct bit is set	3 2	20
JNB	bit,rel	Jump if direct bit is not set	3 2	30
JBC	bit,rel	Jump if direct bit is set and clear bit	3 2	10
CJNE	A,direct,rel	Compare direct to A and jump if not equal	3 2	B5
CJNE	A,#data,rel	Compare immediate to A and jump if not equal	3 2	B4
CJNE	Rr,#data,rel	Compare immed. to reg. and jump if not equal	3 2	B*
CJNE	@Ri,#data,rel	Compare immed. to ind. and jump if not equal	3 2	B6, B7
DJNZ	Rr,rel	Decrement register and jump if not zero	2 2	D*
DJNZ	direct,rel	Decrement direct and jump if not zero	3 2	D5
NOP		No operation	1 1	00

DEVELOPMENT DATA

Notes to Table 1

Data addressing modes

Rr	Working register R0-R7.
direct	128 internal RAM locations and any special function register (SFR).
@Ri	Indirect internal RAM location addressed by register R0 or R1.
#data	8-bit constant included in instruction.
#data16	16-bit constant included as bytes 2 and 3 of instruction.
bit	Direct addressed bit in internal RAM or SFR.
addr16	16-bit destination address. Used by LCALL and LJMP. The branch will be anywhere within the 64 K-byte program memory address space.
addr11	11-bit destination address. Used by ACALL and AJMP. The branch will be within the same 2 K-byte page of program memory as the first byte of the following instruction.
rel	Signed (two's complement) 8-bit offset byte. Used by SJMP and all conditional jumps. Range is −128 to + 127 bytes relative to first byte of the following instruction.

Hexadecimal opcode cross-reference to Table 2

*: 8, 9, A, B, C, D, E, F.

●: 11, 31, 51, 71, 91, B1, D1, F1.

▲: 01, 21, 41, 61, 81, A1, C1, E1.

Table 2 Instruction map

first hexadecimal chacter of opcode ↓ — second hexadecimal character of opcode →

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	NOP	AJMP page 0	LJMP addr16	RR A	INC A	INC dir	INC @Ri 0	1	INC Rr 0	1	2	3	4	5	6	7
1	JBC bit, addr8	ACALL page 0	LCALL addr16	RRC A	DEC A	DEC dir	DEC @Ri 0	1	DEC Rr 0	1	2	3	4	5	6	7
2	JB bit, addr8	AJMP page 1	RET	RL A	ADD A,#data	ADD A,dir	ADD A,@Ri 0	1	ADD A,Rr 0	1	2	3	4	5	6	7
3	JNB bit, addr8	ACALL page 1	RETI	RLC A	ADDC A,#data	ADDC A,dir	ADDC A,@Ri 0	1	ADDC A,Rr 0	1	2	3	4	5	6	7
4	JC addr8	AJMP page 2	ORL dir,A	ORL dir,#data	ORL A,#data	ORL A,dir	ORL A,@Ri 0	1	ORL A,Rr 0	1	2	3	4	5	6	7
5	JNC addr8	ACALL page 2	ANL dir,A	ANL dir,#data	ANL A,#data	ANL A,dir	ANL A,@Ri 0	1	ANL A,Rr 0	1	2	3	4	5	6	7
6	JZ addr8	AJMP page 3	XRL dir,A	XRL dir,#data	XRL A,#data	XRL A,dir	XRL A,@Ri 0	1	XRL A,Rr 0	1	2	3	4	5	6	7
7	JNZ addr8	ACALL page 3	ORL C,bit	JMP @A+DPTR	MOV A,#data	MOV dir,#data	MOV @Ri,#data 0	1	MOV Rr,#data 0	1	2	3	4	5	6	7
8	SJMP addr8	AJMP page 4	ANL C,bit	MOVC A,@A+PC	DIV AB	MOV dir,dir	MOV dir,@Ri 0	1	MOV dir,Rr 0	1	2	3	4	5	6	7
9	MOV DPTR, #data 16	ACALL page 4	MOV bit,C	MOVC A,@A+DPTR	SUBB A,#data	SUBB A,dir	SUBB A,@Ri 0	1	SUBB A,Rr 0	1	2	3	4	5	6	7
A	ORL C,/bit	AJMP page 5	MOV C,bit	INC DPTR	MUL AB		MOV @Ri,dir 0	1	MOV Rr,dir 0	1	2	3	4	5	6	7
B	ANL C,/bit	ACALL page 5	CPL bit	CPL C	CJNE A, #data, addr8	CJNE A,dir, addr8	CJNE @Ri,#data, addr8 0	1	CJNE Rr,#data, addr8 0	1	2	3	4	5	6	7
C	PUSH dir	AJMP page 6	CLR bit	CLR C	SWAP A	XCH A,dir	XCH A,@Ri 0	1	XCH A,Rr 0	1	2	3	4	5	6	7
D	POP dir	ACALL page 6	SETB bit	SETB C	DA A	DJNZ dir, addr8	XCHD A,@Ri 0	1	DJNZ Rr, addr8 0	1	2	3	4	5	6	7
E	MOVX A,@DPTR	AJMP page 7	MOVX A,@Ri 0	1	CLR A	MOV A,dir	MOV A,@Ri 0	1	MOV A,Rr 0	1	2	3	4	5	6	7
F	MOVX @DPTR,A	ACALL page 7	MOVX @Ri,A 0	1	CPL A	MOV dir,A	MOV @Ri,A 0	1	MOV Rr,A 0	1	2	3	4	5	6	7

RATINGS

Limiting values in accordance with the Absolute Maximum System (IEC 134).

parameter	symbol	min.	max.	unit
Input voltage on any pin with respect to ground (V_{SS})	V_I	−0.5	+7.0	V
Input, output current on any single pin	I_I, I_O	−	± 10	mA
Total power dissipation	P_{tot}	−	1.0	W
Storage temperature range	P_{stg}	−65	+150	°C
Operating ambient temperature range				
MAB8051AH-2	T_{amb}	0	+70	°C
MAF8051AH-2	T_{amb}	−40	+85	°C
MAF80A51AH-2	T_{amb}	−40	+110	°C

DEVELOPMENT DATA

DC CHARACTERISTICS

V_{DD} = 5 V (± 10%); V_{SS} = 0 V; T_{amb} = 0 to + 70 °C (MAB8051AH-2)
All voltages measured with respect to V_{SS} unless otherwise stated.

parameter	conditions	symbol	min.	max.	unit
Supply current operating	all outputs disconnected $\overline{EA} = V_{DD}$	I_{CC}	–	125	mA
Power-down (optional)	available for MAB8051AH-2 only	I_{PD}	–	20	µA
Inputs					
LOW level input voltage		V_{IL}	–0.5	0.8	V
HIGH level input voltage (except RST, XTAL2)		V_{IH}	2.0	V_{CC} + 0.5	V
HIGH level input voltage (RST and XTAL2)	XTAL1 = V_{SS}	V_{IH1}	2.5	V_{CC} + 0.5	V
Input current logic 0 (Ports 1, 2 and 3)	V_{IL} = 0.45 V	$-I_{IL}$	–	–500	µA
Input current logic 1 (RST)	$V_I = V_{CC} - 1.5$ V	I_{IH1}	–	500	µA
Input leakage current (Port 0, $\overline{EA}$,)	$V_{SS} < V_I < V_{CC}$	$\pm I_{LI1}$	–	10	µA
Input current logic 0 (XTAL2)	V_{IL} = 0.45 V, XTAL1 = V_{SS}	I_{IL2}	–	–3.2	mA
I/O pin capacitance	frequency = 1 MHz; T_{amb} = 25 °C	$C_{I/O}$	–	10	pF
Outputs					
LOW level output voltage (Ports 1, 2, and 3)	(note 1) I_{OL} = 1.6 mA	V_{OL}	–	0.45	V
LOW level output voltage (Port 0, ALE, $\overline{PSEN}$)	(note 1) I_{OL} = 3.2 mA	V_{OL1}	–	0.45	V
HIGH level output voltage (Ports 1, 2, and 3)	V_{DD} = 5 V ± 10%| $-I_{OH}$ = 80 µA	V_{OH}	2.4	–	V
HIGH level output voltage (Port 0 in external bus mode, ALE, $\overline{PSEN}$)	$-I_{OH}$ = 400 µA	V_{OH1}	–	–	V

DC CHARACTERISTICS

V_{CC} = 5.0 V ± 10%; V_{SS} = 0 V; T_{amb} = −40 °C to +85 °C (MAF8051AH-2); T_{amb} = −40 to +110 °C (MAF80A51AH-2). All voltages measured with respect to V_{SS} unless otherwise specified.

parameter	conditions	symbol	min.	max.	unit
Supply current	all outputs disconnected $\overline{EA} = V_{CC}$	I_{CC}	–	135	mA
Inputs					
LOW level input voltage		V_{IL}	−0.5	0.8	V
HIGH level input voltage (except RST, XTAL2)		V_{IH}	2.1	V_{CC}+0.5	V
HIGH level input voltage (RST and XTAL2)	XTAL2 = V_{SS}	V_{IH1}	2.6	V_{DD}+0.5	V
Input current logic 0 (Ports 1, 2 and 3)	V_{IL} = 0.45 V	I_{IL}	–	500	µA
Input current logic 0 (XTAL2)	V_{IL} = 0.45 V, XTAL1 = V_{SS}	I_{IL1}	–	−4.0	mA
Input leakage current (Port 0, $\overline{EA}$)	$V_{SS} < V_I < V_{CC}$	± I_{LI}	–	10	µA
Input current logic 1 (RST)	$V_I = V_{CC} - 1.5$ V	I_{IH1}	–	500	µA
I/O pin capacitance	frequency = 1 MHz, T_{amb} = 25 °C	$C_{I/O}$	–	10	pF
Outputs					
LOW level output voltage (Ports 1, 2, and 3)	(note 1) I_{OL} = 1.6 mA	V_{OL}	–	0.45	V
LOW level output voltage (Port 0, ALE, $\overline{PSEN}$)	(note 1) I_{OL} = 3.2 mA	V_{OL1}	–	0.45	V
HIGH level output voltage (Ports 1, 2, and 3)	I_{OH} = −80 µA	V_{OH}	2.4	–	V
HIGH level output voltage (Port 0 in external bus mode, ALE, $\overline{PSEN}$)	$-I_{OH}$ = 400 µA	V_{OH1}	2.4	–	V

Note to the DC characteristics

1. The value of V_{OL} will be degraded when the MAB8051AH-2 rapidly discharges external capacitance. This AC noise is most pronounced during the set up of address data. Therefore, when implementing external memory locate the latch or buffer as close to processor as possible.

datum	active ports	time slot interval	degraded I/O lines	V_{OL} (max.)
address	P0, P2	TS3, TS9	P1, P3	0.8 V
write data	P0	TS6	P1, P3, ALE	0.8 V

DEVELOPMENT DATA

AC CHARACTERISTICS

V_{CC} = 5 V ± 10%, V_{SS} = 0 V; $t_{CKmin} = 1/f_{max}$ (maximum operating frequency)

T_{amb} = 0 to + 70 °C, t_{CKmin} = 83 ns (MAB8051AH-2)

T_{amb} = 0 to + 70 °C, t_{CKmin} = 66 ns (MAB8051AH-2)

T_{amb} = −40 to + 85 °C, t_{CKmin} = 83 ns (MAF8051AH-2)

T_{amb} = −40 to + 110 °C, t_{CKmin} = 83 ns (MAF80A51AH-2)

C_L = 100 pF for Port 0, ALE and $\overline{PSEN}$; C_L = 80 pF for all other outputs unless otherwise specified. (See waveforms Figs 12, 13 and 14).

parameter	symbol	10 MHz		12 MHz		variable clock		unit
		min.	max.	min.	max.	min.	max.	
Program memory								
ALE pulse duration	t_{LL}	160	–	127	–	$2t_{CK}-40$	–	ns
Address set-up time to ALE	t_{AL}	45	–	28	–	$t_{CK}-55$	–	ns
Address hold time after ALE	t_{LA}	65	–	48	–	$t_{CK}-35$	–	ns
Time from ALE to valid instruction input	t_{LIV}	–	300	–	233	–	$4t_{CK}-100$	ns
Time from ALE to control pulse $\overline{PSEN}$	t_{LC}	75	–	58	–	$t_{CK}-25$	–	ns
Control pulse duration $\overline{PSEN}$	t_{CC}	265	–	215	–	$3t_{CK}-35$	–	ns
Time from $\overline{PSEN}$ to valid instruction input	t_{CIV}	–	175	–	125	–	$3t_{CK}-125$	ns
Input instruction hold time after $\overline{PSEN}$	t_{CI}	0	–	0	–	0	–	ns
Input instruction float delay after $\overline{PSEN}$	t_{CIF}	–	80	–	63	–	$t_{CK}-20$	ns
Address valid after $\overline{PSEN}$	t_{AC}	92	–	75	–	$t_{CK}-8$	–	ns
Address to valid instruction input	t_{AIV}	–	385	–	302	–	$5t_{CK}-115$	ns
Address float delay after $\overline{PSEN}$	t_{AFC}	12	–	12	–	12	–	ns

AC CHARACTERISTICS (continued)

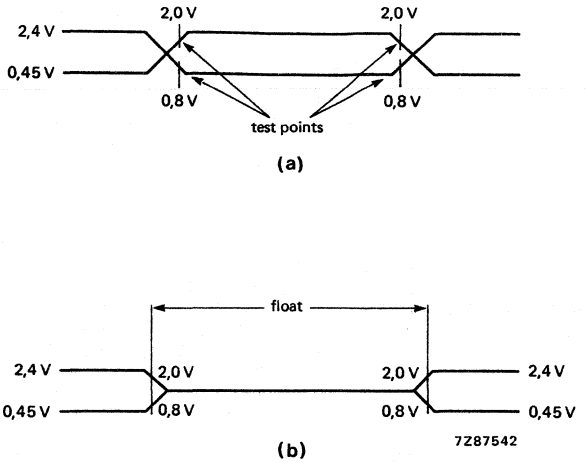
parameter	symbol	10 MHz		12 MHz		variable clock		unit
		min.	max.	min.	max.	min.	max.	
External data memory								
$\overline{RD}$ pulse duration	t_{RR}	500	–	400	–	$6t_{CK}-100$	–	ns
$\overline{WR}$ pulse duration	t_{WW}	500	–	400	–	$6t_{CK}-100$	–	ns
Address hold time after ALE	t_{LA}	65	–	48	–	$t_{CK}-35$	–	ns
$\overline{RD}$ to valid data input	t_{RD}	–	355	–	250	–	$5t_{CK}-165$	ns
Data hold time after $\overline{RD}$	t_{DR}	0	–	0	–	0	–	ns
Data float delay after $\overline{RD}$	t_{DFR}	–	130	–	97	–	$2t_{CK}-70$	ns
Time from ALE to valid data input	t_{LD}	–	650	–	517	–	$8t_{CK}-150$	ns
Address to valid data input	t_{AD}	–	735	–	585	–	$9t_{CK}-165$	ns
Time from ALE to $\overline{RD}$ or $\overline{WR}$	t_{LW}	250	350	200	300	$3t_{CK}-50$	$3t_{CK}+50$	ns
Time from address to $\overline{RD}$ or $\overline{WR}$	t_{AW}	270	–	203	–	$4t_{CK}-130$	–	ns
Time from $\overline{RD}$ or $\overline{WR}$ HIGH to ALE HIGH	t_{WHLH}	60	140	43	123	$t_{CK}-40$	$t_{CK}+40$	ns
Data valid to $\overline{WR}$ transition	t_{DWX}	40	–	23	–	$t_{CK}-60$	–	ns
Data set-up time before $\overline{WR}$	t_{DW}	550	–	433	–	$7t_{CK}-150$	–	ns
Data hold time after $\overline{WR}$	t_{WD}	50	–	33	–	$t_{CK}-50$	–	ns
Address float delay after $\overline{RD}$	t_{AFR}	–	12	–	12	–	12	ns

Notes to AC characteristics:

1. The maximum operating frequency is limited to:

device	maximum clock frequency	
	12	15
MAB8031AH-2	yes	yes
MAB8051AH-2	yes	no
MAF8031AH-2	yes	no
MAF8051AH-2	yes	no
MAF80A31AH-2	yes	no
MAF80A51AH-2	yes	no

2. The minimum operating frequency is limited to 3.5 MHz for all versions.
3. Interfacing the MAB8051AH-2 to devices with float times up to 75 ns is permitted. This limited bus contention will not cause damage to Port 0 drivers.



A.C. testing inputs are driven at 2,4 V for a logic 1 and 0,45 V for a logic 0. Timing measurements are taken at 2,0 V for a logic 1 and 0,8 V for logic 0. The float state is defined as the point at which a Port 0 pin sinks 3,2 mA or sources 400 μA at the voltage test levels.

Fig. 10 AC testing input, output waveform (a) and float waveform (b).

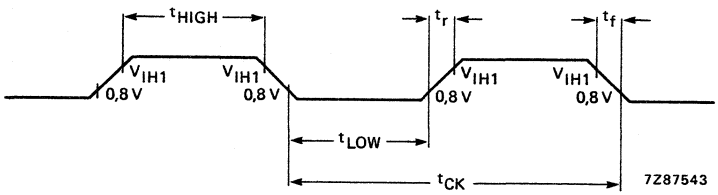


Fig. 11 External clock drive XTAL 2 (see Table 3).

Table 3 External clock drive XTAL 2 (see Fig. 11)

parameter	symbol	variable clock (f = 3,5 to 15 MHz)		unit
		min.	max.	
oscillator clock period	t_{CK}	66,6	286	ns
HIGH time	t_{HIGH}	20	$t_{CK}-t_{LOW}$	ns
LOW time	t_{LOW}	20	$t_{CK}-t_{HIGH}$	ns
rise time	t_r	–	20	ns
fall time	t_f	–	20	ns

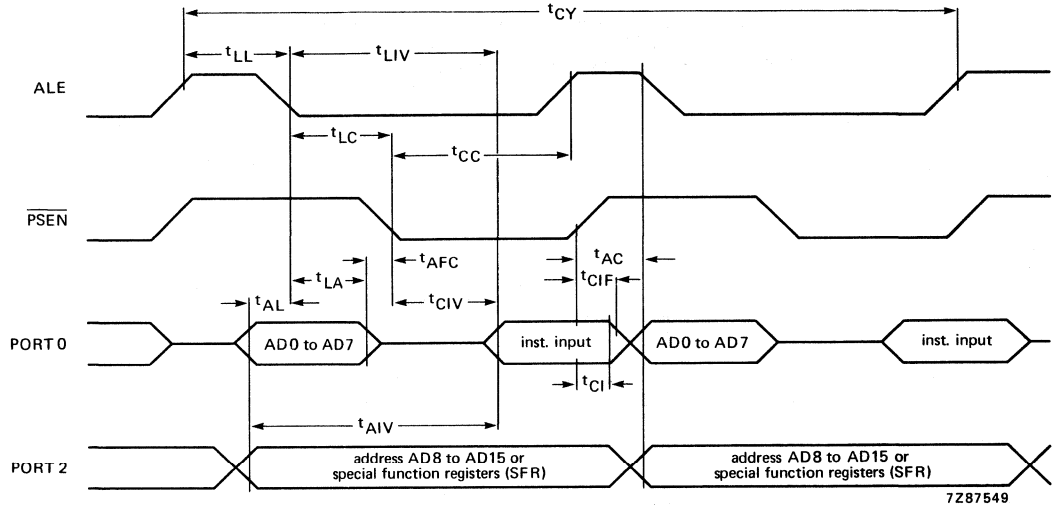


Fig. 12 Read from program memory.

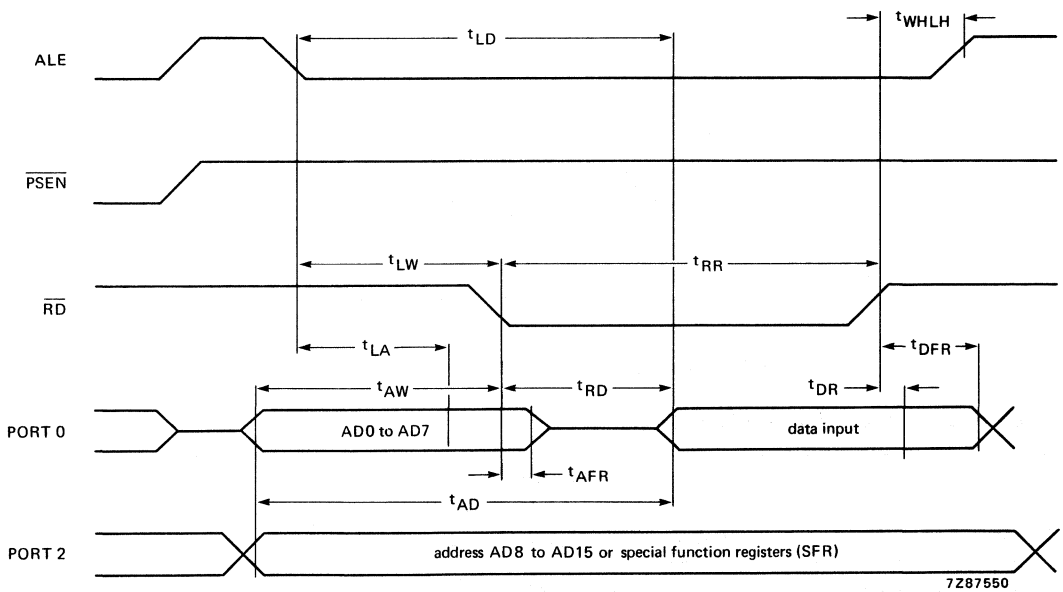


Fig. 13 Read from data memory.

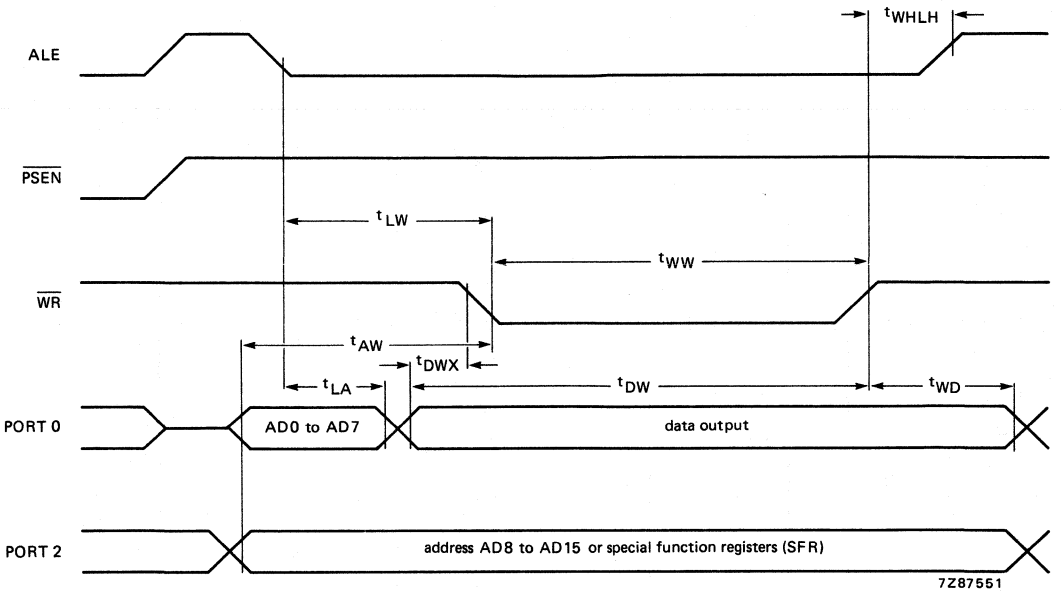


Fig. 14 Write to data memory.

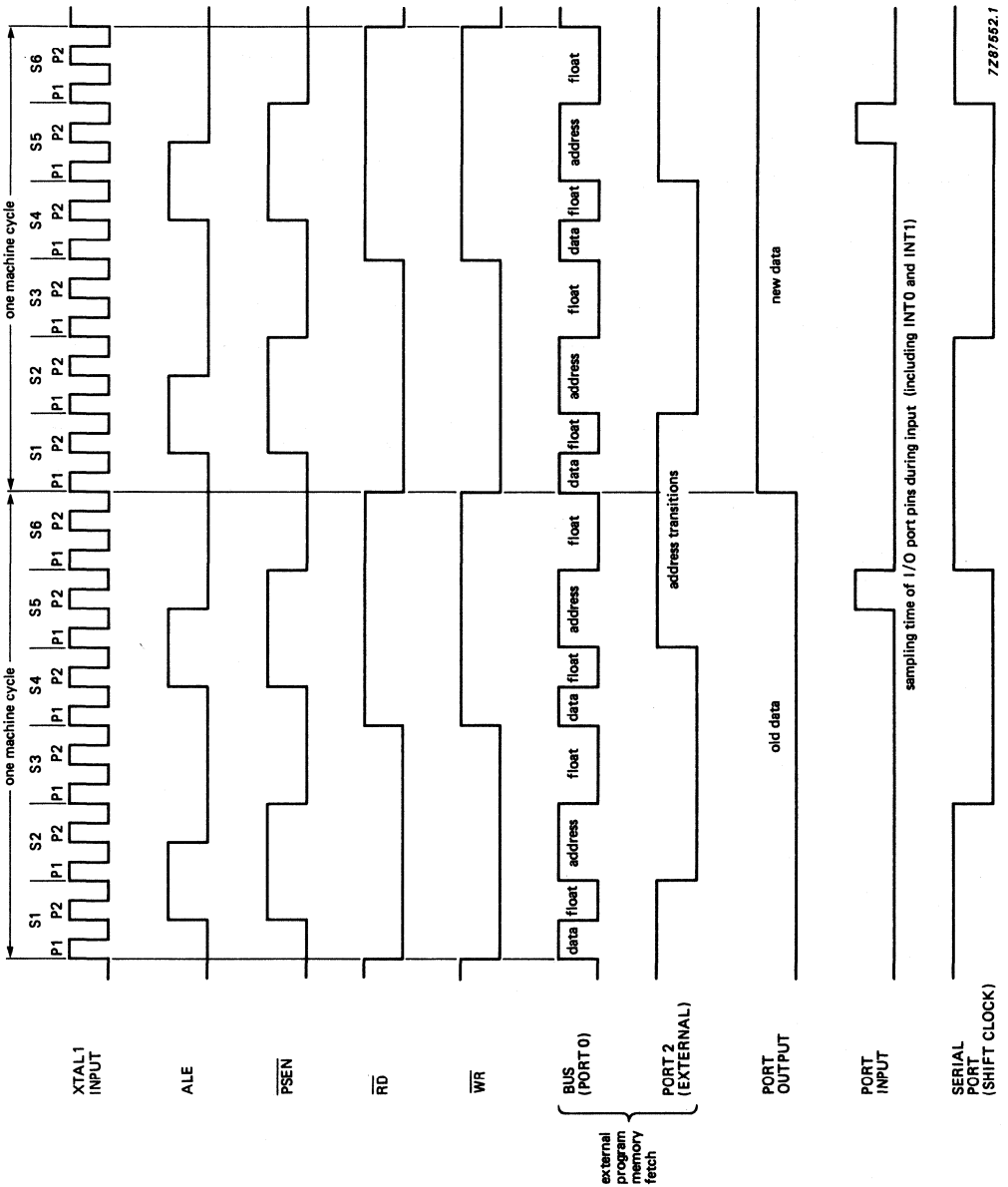


Fig. 15 Instruction cycle timing.

SINGLE-CHIP 8-BIT MICROCONTROLLER

DESCRIPTION

The MAB8052AH is a member of the MAB8051AH family with a higher performance. This single-chip 8-bit microcontroller is manufactured in an advanced 2 μ NMOS process. For this version the following members exist:

- MAB8032AH: ROM-less version of the MAB8052AH
- MAB8052AH: 8 K bytes mask programmable ROM, 256 bytes RAM

Both types are available in 12 MHz versions. In the following, the generic term "MAB8052AH" is used to refer to both family members.

The device provides hardware features, architectural enhancements and new instructions to function as a controller for applications requiring up to 64 K bytes of program memory and/or up to 64 K bytes of data storage.

The MAB8052AH contains a non-volatile 8 K x 8 read-only program memory (not ROM-less version); a volatile 256 x 8 read/write data memory; 32 I/O lines; three 16-bit timer/event counters; a six-source, two-priority-level, nested interrupt structure; a serial I/O port for either multi-processor communications, I/O expansion, or full duplex UART; and on-chip oscillator and timing circuits. For systems that require extra capability, the MAB8052AH can be expanded using standard TTL compatible memories and logic.

The device also functions as an arithmetic processor having facilities for both binary and BCD arithmetic plus bit-handling capabilities. The instruction set consists of 255 instructions; 44% one-byte, 41% two-byte and 15% three-byte. With a 12 MHz crystal, 58% of the instructions are executed in 1 μs and 40% in 2 μs. Multiply and divide instructions require 4 μs.
For further detailed information see the '8051' section of the 'Single-chip 8-bit microcontroller user manual'.

Features

- 8 K x 8 ROM (8052AH only), 256 x 8 RAM
- Four 8-bit ports, 32 I/O lines
- Three 16-bit timer/event counters
- Full-duplex serial port
- External memory expandable to 128 K
- Boolean processing
- 218 bit-addressable locations
- On-chip oscillator
- Six-source interrupt structure with two priority levels
- 58% of instructions executed in 1 μs; multiply and divide in 4 μs
- Enhanced architecture with:
  - non-page-oriented instructions
  - direct addressing
  - four 8-bit register banks
  - stack depth up to 128-bytes
  - multiply, divide, substract and compare
- Upward compatible with MAB8031AH/8051AH
- Extended temperature range with frequency from 3,5 to 10 MHz:
  −40 to + 85 °C MAF8052AH
  −40 to + 100 °C MAF80A52AH

PACKAGE OUTLINES

MAB8032/52AHP: 40-lead DIL, plastic (SOT129).
MAF8032AH/52AH/A32AH/A52AHP: 40-lead DIL, plastic (SOT129).
MAB/MAF8032/52AHWP/A32/A52AHWP: 44-lead plastic leaded chip carrier (PLCC); (SOT187 'pedestal' or SOT187AA 'pocket' versions, these are interchangeable).

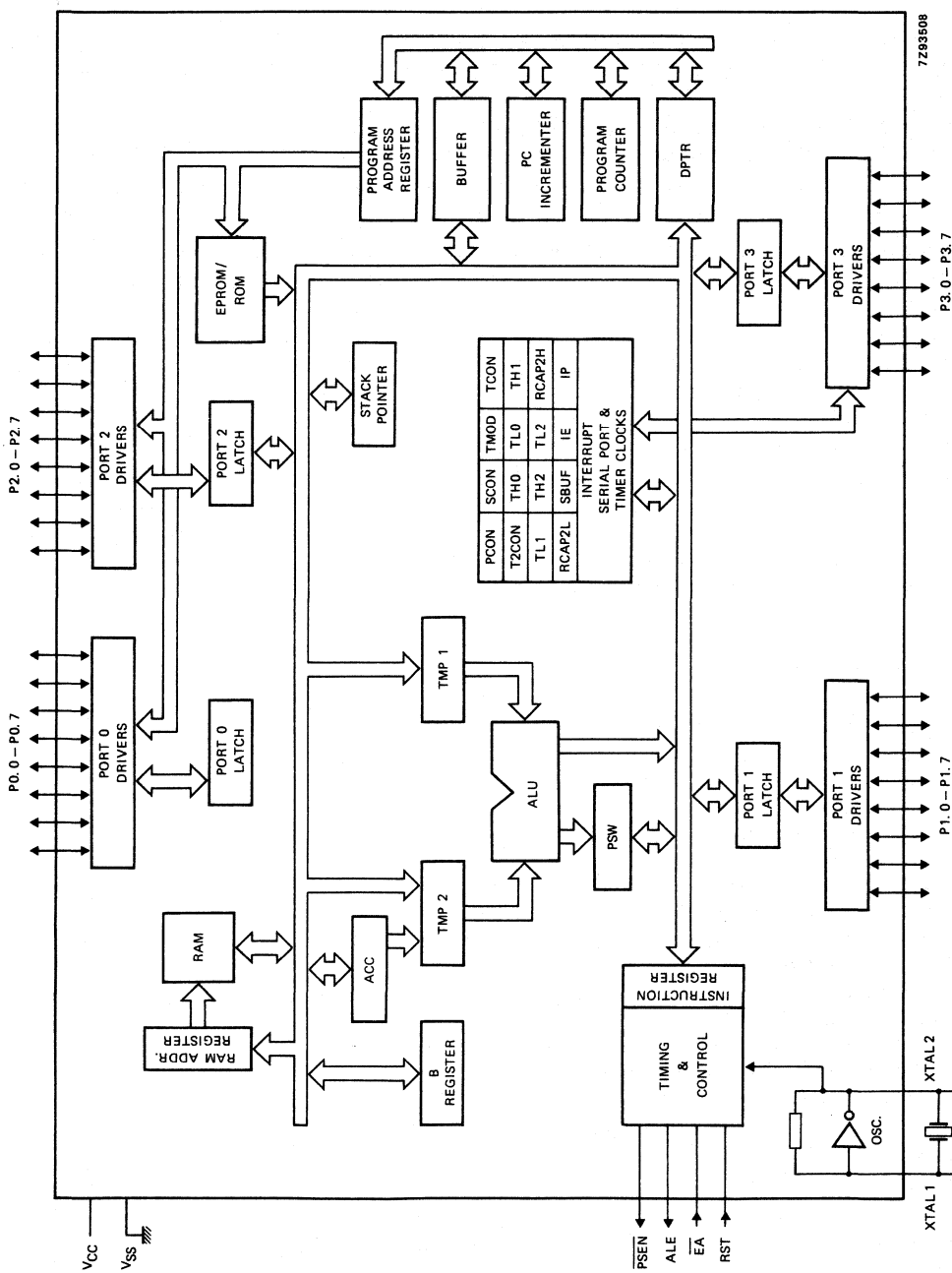


Fig. 1 Block diagram.

DEVELOPMENT DATA

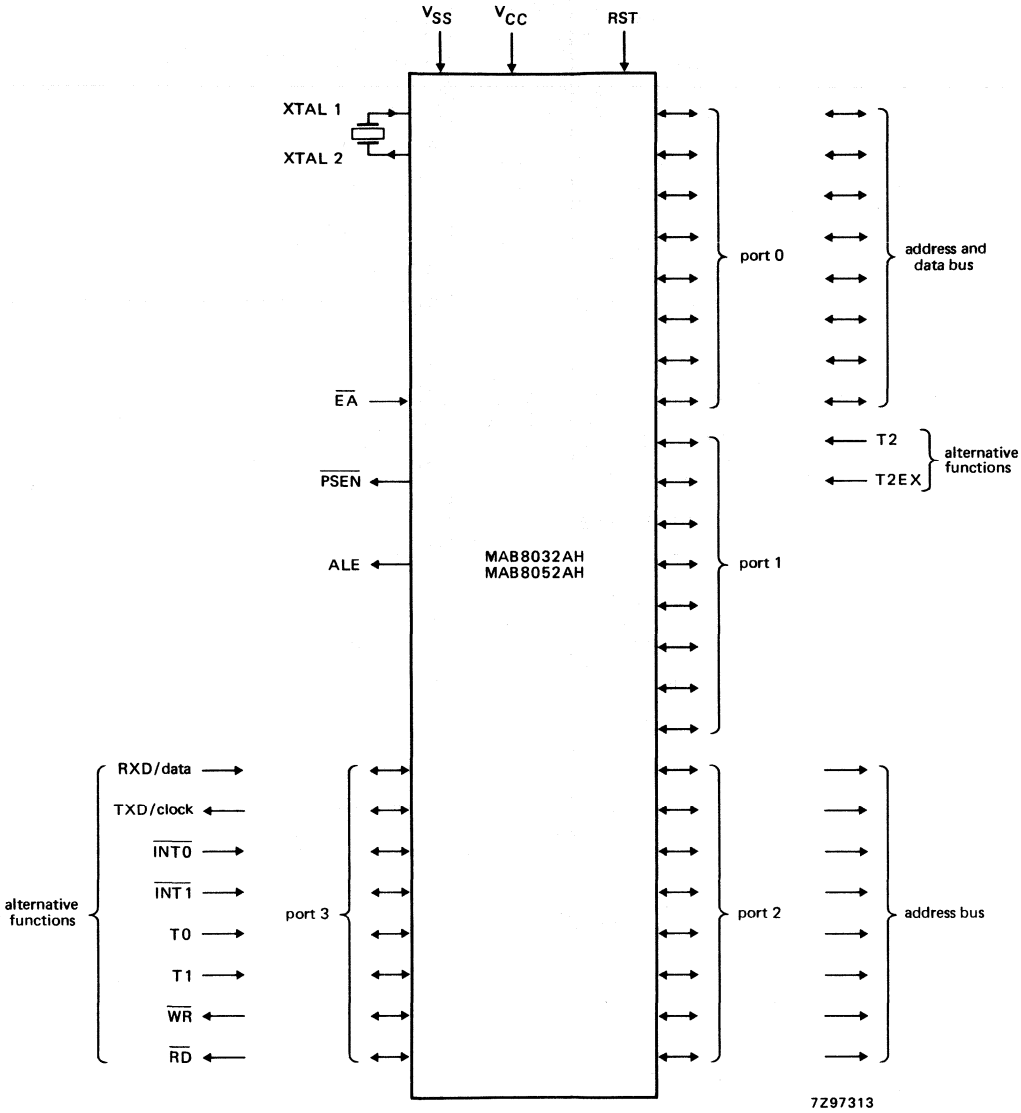


Fig. 2 Functional diagram.

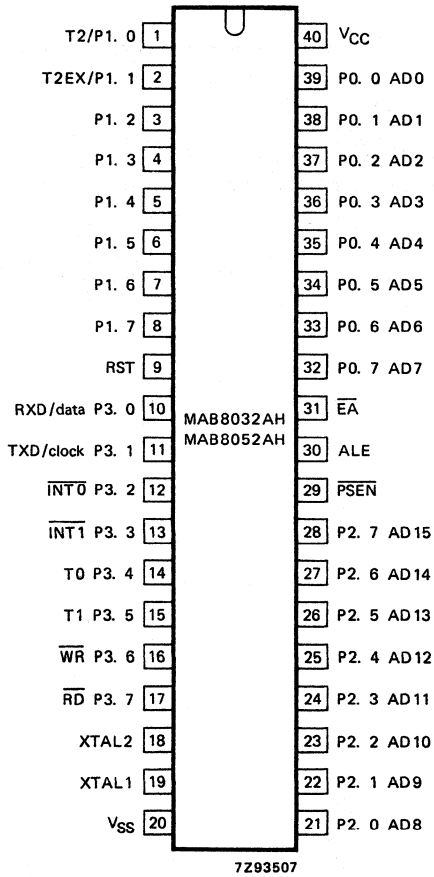


Fig. 3 Pinning diagram.

DEVELOPMENT DATA

PINNING

1-8	P1.0-P1.7	**Port 1:** 8-bit quasi-bidirectional I/O port. It receives the low-order address byte during program verification. Port 1 can sink/source four LS TTL (= 1TTL) inputs. It can drive MOS inputs without external pull-ups. Pins 1 and 2 also supply alternative functions T2 and T2EX. T2 is the counter trigger input for Timer 2; T2EX the external input to Timer 2. Operation of the alternative functions is determined by the relevant output latch programmed to logic 1.
9		**RESET:** a high level on this pin for two machine cycles while the oscillator is running resets the device. An internal pulldown permits Power-On reset using only a capacitor connected to V_{CC} (see Fig. 18).
10-17	P3.0-P3.7	**Port 3:** 8-bit quasi-bidirectional I/O port with internal pull-ups. It also serves the following alternative functions:

Port pin	*Alternative function*
P3.0	**RXD/data:** serial port receiver data input (asynchronous) or data input/output (synchronous)
P3.1	**TXD/clock:** serial port transmitter data output (asynchronous) or clock output (synchronous)
P3.2	**$\overline{\text{INT0}}$:** external interrupt 0 or gate control input for timer/event counter 0
P3.3	**$\overline{\text{INT1}}$:** external interrupt 1 or gate control input for timer/event counter 1
P3.4	**T0** : external input for timer/event counter 0
P3.5	**T1** : external input for timer/event counter 1
P3.6	**$\overline{\text{WR}}$:** external data memory write strobe
P3.7	**$\overline{\text{RD}}$** : external data memory read strobe

Operation of an alternative function is determined by the relevant output latch programmed to logic 1. Port 3 can sink/source four LS TTL inputs. It can drive MOS inputs without external pull-ups.

18	XTAL 2	**Crystal input 2:** output of the inverting amplifier that forms the oscillator, and input to the internal clock generator. Receives the external oscillator signal when an external oscillator is used (see figures 15 and 16).
19	XTAL 1	**Crystal input 1:** input to the inverting amplifier that forms the oscillator. Connected to V_{SS} when an external oscillator is used.
20	V_{SS}	**Ground:** circuit earth potential.
21-28	P2.0-P2.7	**Port 2:** 8-bit quais-bidirectional I/O port with internal pull-ups. It emits the high-order address byte when accessing external memory. It also receives the high-order address bits and control signals during program verification. Port 2 can sink/source four LS TTL inputs. It can drive MOS inputs without external pull-ups.
29	$\overline{\text{PSEN}}$	**Program Store Enable output:** read strobe to the external Program Memory. It is activated twice each machine cycle during fetches from external Program Memory. When executing out of external Program Memory two activations of $\overline{\text{PSEN}}$ are skipped during each access to external Data Memory. $\overline{\text{PSEN}}$ is not activated (remains HIGH) during fetches from internal Program Memory.

PINNING (continued)

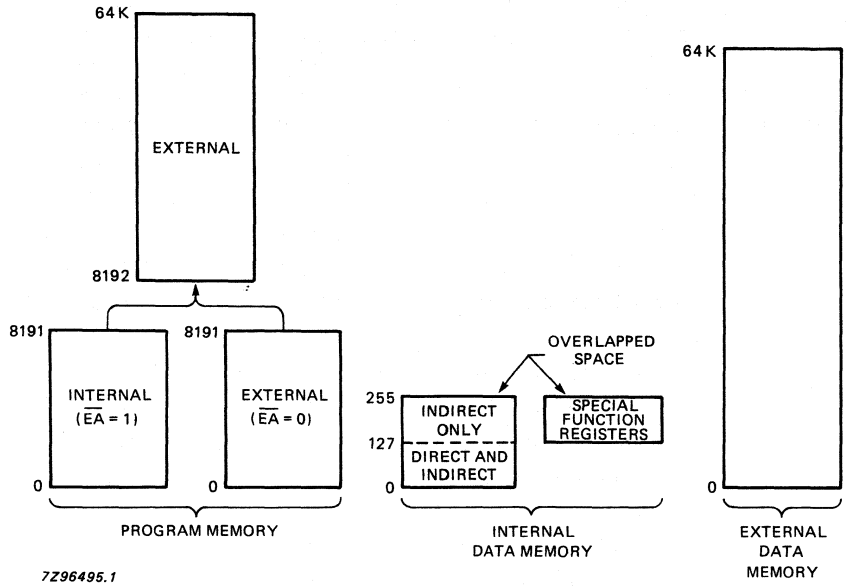
30	ALE	**Address Latch Enable output**: latches the low byte of the address during accesses to external memory in normal operation. It is activated every six oscillator periods except during an external data memory addess.
31	$\overline{EA}$	When $\overline{EA}$ is held at a TTL high level the CPU executes out of the internal Program Memory (ROM), provided the Program Counter is less than 8192. When $\overline{EA}$ is held at a TTL low level the CPU executes out of external Program Memory. $\overline{EA}$ does not float.
32-39	P0.7-P0.0	**Port 0**: 8-bit open drain bidirectional I/O port. It is also the multiplexed low-order address and data bus during accesses to external memory (during these accesses it activates internall pull-ups). It also outputs code bytes during program verification. (External pull-ups are required during program verification). Port 0 can sink (and in bus operations can source) eight LS TTL inputs.
40	V_{CC}	**Power Supply**: + 5 V power supply pin during normal operation.

As the MAB8052AH is based on the MAB8051 the following pages are a description of the additional features.

MEMORIES (see Fig. 4)

ROM: 8 K bytes mask programmable. An external ROM is accessed up to 64 K to accommodate an address higher than 8191.

RAM: 256 bytes (on-chip) plus special function registers detailed in Table 1.



Where:

The internal data memory locations are addressable direct/indirect as follows:

location	addressed
RAM 0 to 127	direct and indirect
RAM 128 to 255	indirect only
SFR 128 to 255	direct only

Fig. 4 Memory map.

SPECIAL FUNCTION REGISTERS

Table 1 Special Function Registers (SFR)

symbol	name	address	contents after reset
ACC*	accumulator	0E0H	00H
B*	B register	0F0H	00H
PSW*	program status word	0D0H	00H
SP	stack pointer	81H	07H
DPTR	data pointer		
	DPH	83H	0000H
	DPL	82H	0000H
P0*	port 0	80H	0FFH
P1*	port 1	90H	0FFH
P2*	port 2	0A0H	0FFH
P3*	port 3	0B0H	0FFH
IP*	interrupt priority control	0B8H	XX000000B
IE*	interrupt enable control	0A8H	0X000000B
TMOD	timer/counter mode control	89H	00H
T2CON*	timer/counter 2 control	0C8H	00H
TCON	timer/counter control	88H	00H
TH0	timer/counter 0 (high byte)	8CH	00H
TL0	timer/counter 0 (low byte)	8AH	00H
TH1	timer/counter 1 (high byte)	8DH	00H
TL1	timer/counter 1 (low byte)	8BH	00H
TH2	timer/counter 2 (high byte)	0CDH	00H
TL2	timer/counter 2 (low byte)	0CCH	00H
RCAP2H	timer/counter 2 capture register (high byte)	0CBH	00H
RCAP2L	timer/counter 2 capture register (low byte)	0CAH	00H
SCON*	serial control	98H	00H
SBUF	serial data buffer	99H	indeterminate
PCON	power control	87H	0XXXXXXXB

Where

H = Hexadecimal
B = Binary

* Registers are both byte and bit addressable.

T2CON

Special Function Register T2CON is the timer/counter 2 control register. T2CON contains the control and status bits shown in Fig. 5 and described in Table 2.

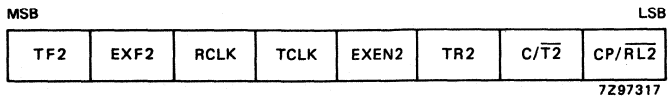


Fig. 5 Timer/counter 2 control register (T2CON).

Table 2 T2CON: Control and status bits

symbol	position	name	operation
TF2	T2CON.7	Timer 2 overflow flag	Set by a Timer 2 overflow and must be cleared by software. TF2 will not be set when either RCLK = 1 or TCLK = 1.
EXF2	T2CON.6	Timer 2 external flag	Set when either a capture or reload is caused by a negative transition on T2EX and EXEN2 = 1. When Timer 2 interrupt is enabled, EXF2 = 1 will cause the CPU to vector to the Timer 2 interrupt routine. EXF2 must be cleared by software.
RCLK	T2CON.5	Receive clock flag	When set, causes the serial port to use Timer 2 overflow pulses for its receive clock in modes 1 and 3. RCLK = 0 causes Timer 1 overflow to be used for the receive clock.
TCLK	T2CON.4	Transmit clock flag	When set causes the serial port to use Timer 2 overflow pulses for its transmit clock in modes 1 and 3. TCLK = 0 causes Timer 1 overflows to be used for the transmit clock.
EXEN2	T2CON.3	Timer 2 external enable flag	When set allows a capture or reload to occur as a result of a negative transition on T2EX if Timer 2 is not being used to clock the serial port. EXEN2 = 0 causes Timer 2 to ignore events at T2EX.
TR2	T2CON.2	Start/stop control for Timer 2	A logic 1 starts the timer.
C/$\overline{T2}$	T2CON.1	Time or counter select (Timer 2)	0 = internal timer (OSC/12) 1 = external event counter (falling edge triggered)
CP/$\overline{RL2}$	T2CON.0	Capture/reload flag	When set capture will occur on negative transitions at T2EX if EXEN2 = 1. When cleared automatic reloads will occur either with Timer 2 overflows or negative transitions at T2EX when EXEN2 = 1. When either RCLK = 1 or TCLK = 1, this bit is ignored and the timer is forced to automatic-reload on Timer 2 overflow.

DEVELOPMENT DATA

SPECIAL FUNCTION REGISTERS (continued)

TH2; TL2

Timer/counter 2 (high byte); Timer/counter 2 (low byte) are the register pair for the 16-bit counting of timer/counter 2.

RCAP2H; RCAP2L

Timer/counter 2 capture (high byte); Timer/counter 2 capture (low byte) are the register pair for the "capture mode" of Timer 2. In this mode, a signal at T2EX initiates a copy of TH2 and TL2 into RCAP2H and RCAP2L. Timer 2 also has an automatic-reload mode. RCAP2H and RCAP2L hold the reload value for this mode.

T2CON

Timer 2 is 16-bit and operates the same as Timer 0 and Timer 1. The operating mode is selected by the C/$\overline{T2}$ bit in the T2CON register. Table 3 shows the operating modes of Timer 2.

Table 3 Timer 2 operating modes.

RCLK + TCLK	CP/$\overline{RL2}$	TR2	mode
0	0	1	16-bit automatic-reload
0	1	1	16-bit capture
1	X	1	baud rate generator
X	X	0	off

Where

X = don't care.

Capture mode (see Fig. 6)

In the capture mode there are two options which are selected by bit EXEN2 in T2CON. If EXEN2 = 0, then Timer 2 is a 16-bit timer or counter which on overflow will set bit TF2 (Timer 2 overflow bit). TF2 can be used to generate an interrupt. If EXEN2 = 1, Timer 2 operates as for EXEN2 = 0 with the added feature that a 1-to-0 transition at the external input T2EX causes the current value in the Timer 2 registers (TL2 and TH2) to be captured into registers RCAP2L and RCAP2H respectively. The 1-to-0 transition of T2EX also causes bit EXF2 in T2CON to be set. EXF2 can be used to generate an interrupt.

Automatic-reload mode (see Fig. 7)

In the automatic-reload mode there are again two options which are selected by bit EXEN2 in T2CON. If EXEN2 = 0, when Timer 2 overflows it sets TF2 and causes the Timer 2 registers to be reloaded with the 16-bit value in registers RCAP2L and RCAP2H, which are preset by software. If EXEN2 = 1, Timer 2 operates as for EXEN2 = 0 with the added feature that a 1-to-0 transition at the external input T2EX triggers the 16-bit reload and sets EXF2.

Baud rate generator mode (see Fig. 8)

The baud rate generator mode is selected by RCLK = 1 and/or TCLK = 1 in T2CON. Thus baud rates for transmit and receive can be simultaneously different. This mode is described in conjunction with the serial port.

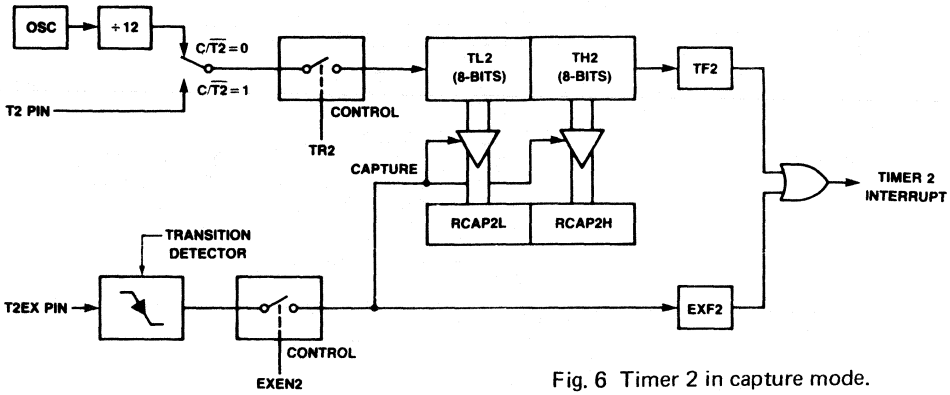


Fig. 6 Timer 2 in capture mode.

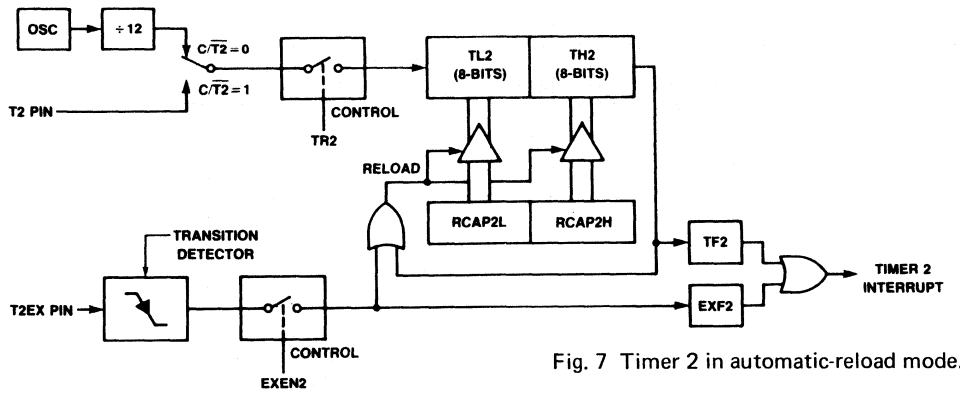


Fig. 7 Timer 2 in automatic-reload mode.

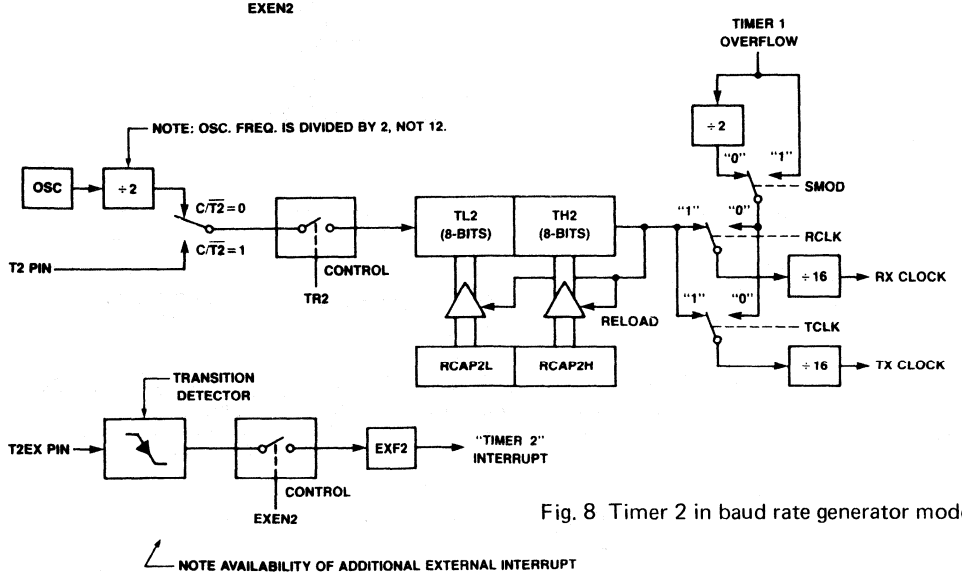


Fig. 8 Timer 2 in baud rate generator mode.

DEVELOPMENT DATA

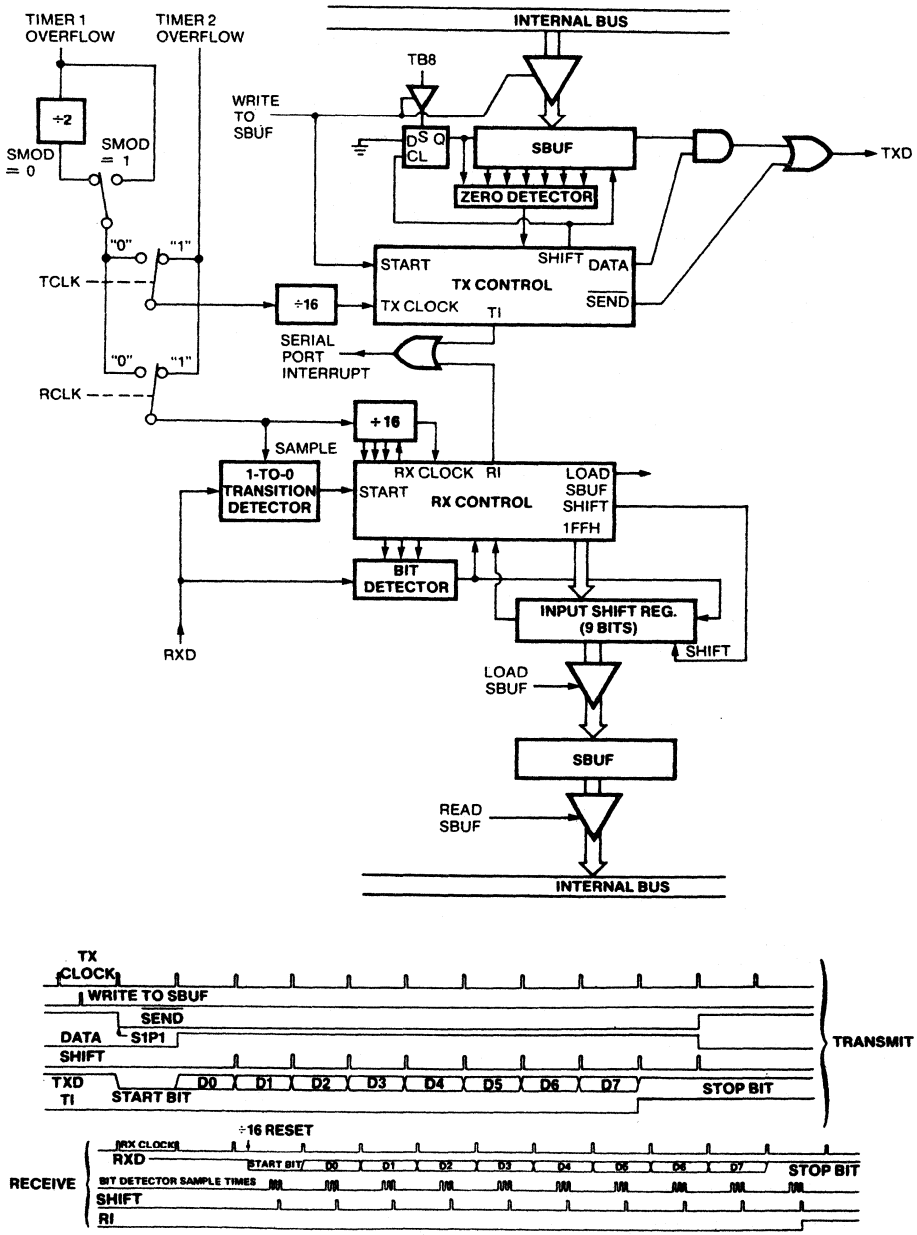


Fig. 9 The serial port in mode 1.

DEVELOPMENT DATA

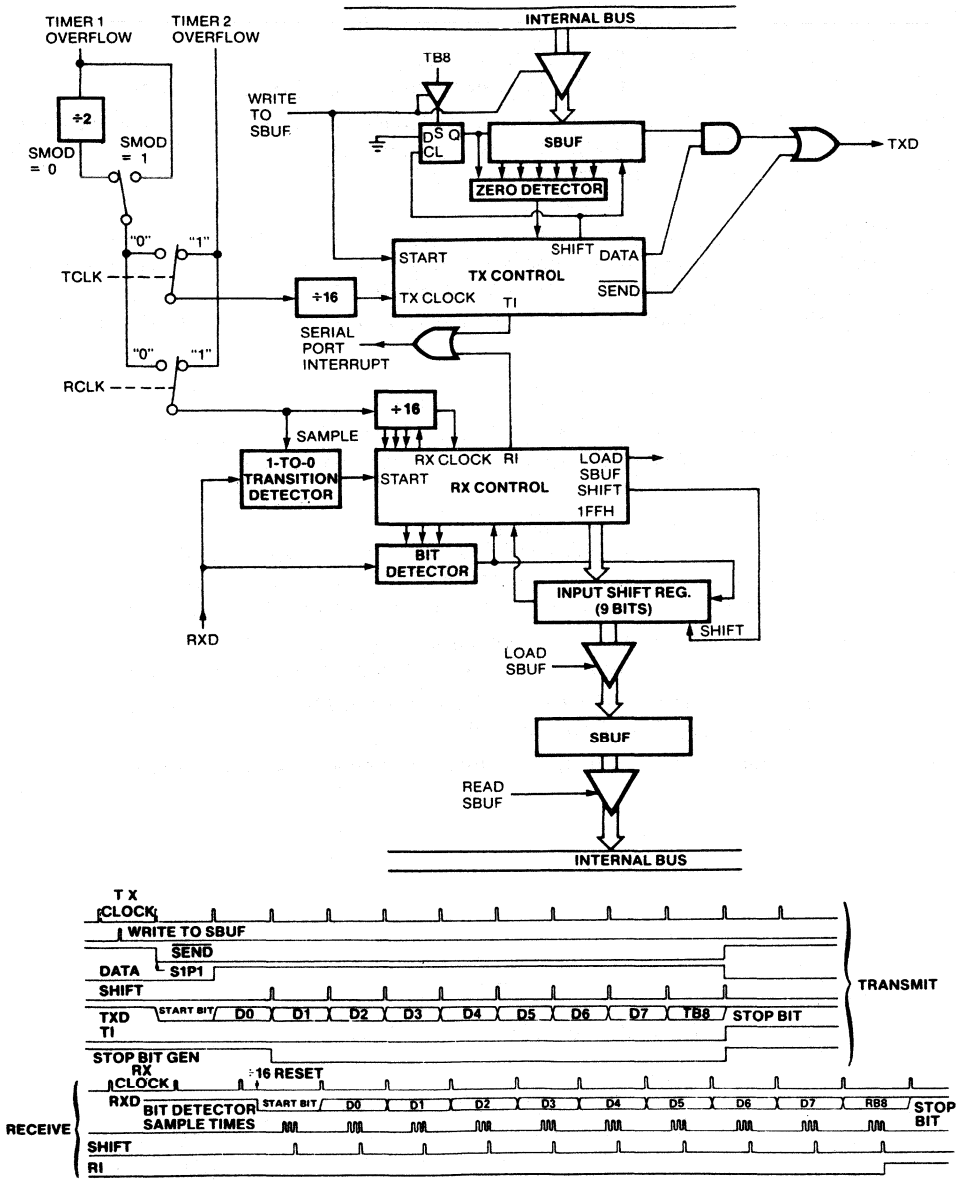


Fig. 10 The serial port in mode 3.

INTERRUPTS

The MAB8052AH has 6 interrupts as shown in Fig. 11.

$\overline{INT0}$; $\overline{INT1}$

The external interrupts $\overline{INT0}$ and $\overline{INT1}$ can be level-activated or transition-activated dependent on bits IT0 and IT1 in register TCON. The flags that generate these interrupts are bits IE0 and IE1.

When a transition-activated external interrupt is generated, the interrupt flag is cleared by the on-chip hardware when the CPU transfers control to the service routine. If the interrupt is level-activated, the request flag is controlled by the external requesting source.

Timer 0; Timer 1

The Timer 0 and Timer 1 interrupts are generated by TF0 and TF1, which are set by an overflow in their respective timer/counter registers. This is not applicable for Timer 0 in mode 3. When a timer interrupt is generated, the interrupt flag is cleared by the on-chip hardware when the CPU transfers control to the service routine.

Timer 2

The interrupt of Timer 2 is generated by the logical OR of TF2 and EXF2. The service routine determines whether it was TF2 or EXF2 that generated the interrupt, and the bits are cleared by software.

Serial port interrupt

The serial port interrupt is generated by the logical OR of R1 and T1 flags. These flags are not cleared by the on-chip hardware when the CPU transfers control to the service routine. The service routine determines whether it was R1 or T1 that generated the interrupt, and the bit is cleared by software.

All of the bits that generate interrupts can be set or cleared by software, with the same result as being set or cleared by hardware. Thus interrupts can be generated, or pending interrupts cancelled, by software.

Each of the interrupt sources can be individually enabled or disabled by setting or clearing a bit in Special Function Register IE (see Fig. 12 and Table 4).

Register IE contains a global disable bit (EA) which disables all interrupts simultaneously.

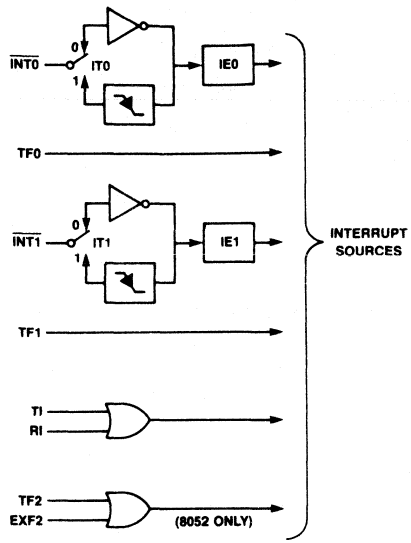


Fig. 11 Interrupt sources.

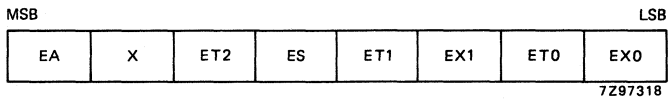


Fig. 12 Interrupt enable register (IE).

Table 4 IE: Control and status bits

symbol	position	function
EA	IE.7	Disables all interrupts. If EA = 0 no interrupt will be acknowledged. If EA = 1 each interrupt source is individually enabled or disabled by setting or clearing its enable bit.
–	IE.6	Reserved.
ET2	IE.5	Enables or disables the Timer 2 overflow or capture interrupt. If ET2 = 0 the Timer 2 interrupt is disabled.
ES	IE.4	Enables or disables the Serial Port interrupt. If ES = 0 the Serial Port interrupt is disabled.
ET1	IE.3	Enables or disables the Timer 1 overflow interrupt. If ET1 = 0 the Timer 1 interrupt is disabled.
EX1	IE.2	Enables or disables external interrupt 1. If EX1 = 0 the external interrupt is disabled.
ET0	IE.1	Enables or disables the Timer 0 overflow interrupt. If ET0 = 0 the Timer 0 interrupt is disabled.
EX0	IE.0	Enables or disables external interrupt 0. If EX0 = 0 the external interrupt is disabled.

DEVELOPMENT DATA

INTERRUPTS (continued)

Priority level structure (see Fig. 13 and Table 5)

Each interrupt source can be programmed to one of two priority levels by setting or clearing a bit in Special Function Register IP. A low-priority interrupt can be interrupted by a high-priority interrupt, but not by another low-priority interrupt. A high-priority interrupt cannot be interrupted by any other interrupt source.

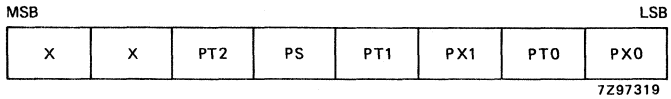


Fig. 13 Interrupt priority register (IP).

Table 5 IP: Control and status bits

symbol	position	function
—.	IP.7	Reserved.
—	IP.6	Reserved.
PT2	IP.5	Defines the Timer 2 interrupt priority level. PT2 = 1 programs it to the higher priority level.
PS	IP.4	Defines the Serial Port interrupt priority level. PS1 = 1 programs it to the higher priority level.
PT1	IP.3	Defines the Timer 1 interrupt priority level. PT1 = 1 programs it to the higher priority level.
PX1	IP.2	Defines the external interrupt 1 priority level. PX1 = 1 programs it to the higher priority level.
PT0	IP.1	Defines the Timer 0 interrupt priority level. PT0 = 1 programs it to the higher priority level.
PX0	IP.0	Defines the external interrupt 0 priority level. PX0 = 1 programs it to the higher priority level.

The interrupt response timing is shown in Fig. 14. The fastest possible response occurs when C2 is the final cycle of an instruction other than "return from interrupt" instruction (RETI), or an access to registers IE or IP.

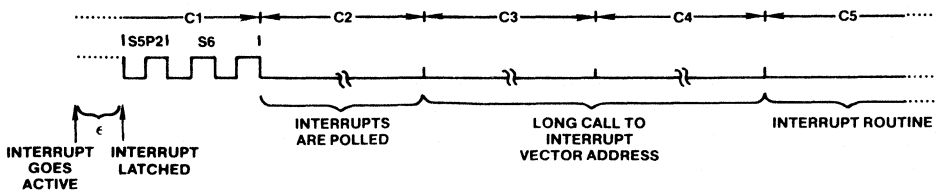


Fig. 14 Interrupt response timing diagram.

Priority level structure (continued)

If two requests of different priority levels are received simultaneously, the higher priority request is serviced. If requests of the *same* priority level are received simultaneously, an internal polling sequence determines which request is serviced. This second priority structure determined by the polling sequence is detailed in Table 6.

Table 6 Vector addresses by interrupt depending on interrupt source and priority

number	source	priority within level	vector address
1	IE0	(highest)	0003H
2	TF0		000BH
3	IE1		0013H
4	TF1		001BH
5	R1 + T1		0023H
6	TF2 + EXF2	(lowest)	002BH

OSCILLATOR CIRCUITRY

The oscillator circuitry of the MAB8052AH is a single-stage inverting amplifier in a Pierce oscillator configuration. The circuitry has a combination of depletion and enhancement mode MOS FETs to produce the inverting characteristics, and passive components. Either a crystal or ceramic resonator can be used as the feedback element to complete the oscillator circuitry. XTAL1, pin 19, is the high gain amplifier input, and XTAL2, pin 18, is the output (see Fig. 15).

To drive the MAB8052AH externally, XTAL1 should be connected to ground and XTAL2 driven from an external source (see Fig. 16).

DEVELOPMENT DATA

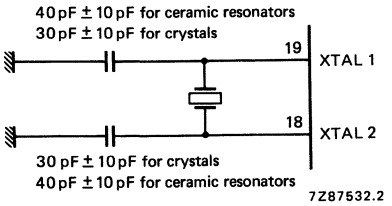


Fig. 15 MAB8052AH oscillator circuit.

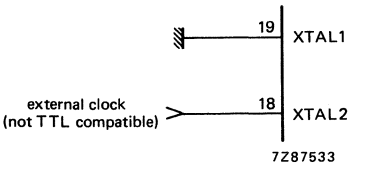


Fig. 16 Driving the MAB8052AH from an external source.

RESET CIRCUITRY

The reset circuitry for the MAB8052AH is connected to the reset pin, RST, as shown in Fig. 17. A Schmitt trigger is used at the input for noise rejection. The output of the Schmitt trigger is sampled by the reset circuitry every machine cycle.

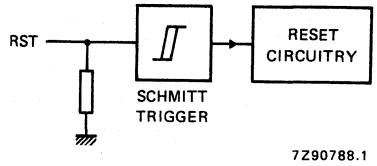


Fig. 17 Reset configuration at RST.

A reset is accomplished by holding the RST pin HIGH for at least two machine cycles (24 oscillator periods), while the oscillator is running. The CPU responds by executing an internal reset. It also configures the ALE and $\overline{PSEN}$ pins as inputs. (They are quasi-bidirectional.) The internal reset is executed during the second cycle in which RST is HIGH and is repeated every cycle until RST goes LOW. It leaves the internal registers as shown in Table 1.

The internal RAM is not affected by reset. When V_{CC} is turned on, the RAM content is indeterminate.

Power-on reset (see Fig. 18)

When V_{CC} is turned on, and provided its rise-time does not exceed 10 ms, an automatic reset can be obtained by connecting the RST pin to V_{CC} via a 10 μF capacitor. When the power is switched on, the voltage on the RST pin (V_{RST}) is the difference between V_{CC} and the capacitor voltage, and decreases from V_{CC} as the capacitor charges through the internal resistor (R_{RST}) to ground. The larger the capacitor, the more slowly V_{RST} decreases. V_{RST} must remain above the lower threshold of the Schmitt trigger long enough to effect a complete reset. The time required is the oscillator start-up time, plus 2 machine cycles.

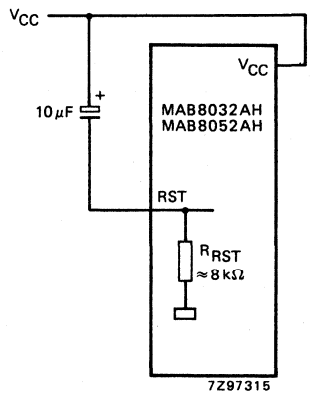


Fig. 18 Power-on reset.

INSTRUCTION SET

The MAB8052AH uses a powerful instruction set to allow expansion of on-chip CPU peripherals and to optimize byte efficiency and execution speed. Reassigned opcodes add new high-power operations and permit new addressing modes to make old operations more orthogonal. The instruction set consists of 49 single-byte, 45 two-byte and 17 three-byte instructions. When using a 12 MHz oscillator, 64 instructions execute in 1 μs and 45 instructions execute in 2 μs. Multiply and divide instructions execute in 4 μs.

Table 7 Instruction set description

mnemonic		description	bytes/ cycles	opcode (hex.)
Arithmetic operation				
ADD	A,Rr	Add register to A	1 1	2*
ADD	A,direct	Add direct byte to A	2 1	25
ADD	A,@Ri	Add indirect RAM to A	1 1	26, 27
ADD	A,#data	Add immediate data to A	2 1	24
ADDC	A,Rr	Add register to A with carry flag	1 1	3*
ADDC	A,direct	Add direct byte to A with carry flag	2 1	35
ADDC	A,@Ri	Add indirect RAM to A with carry flag	1 1	36, 37
ADDC	A,#data	Add immediate data to A with carry flag	2 1	34
SUBB	A,Rr	Subtract register from A with borrow	1 1	9*
SUBB	A,direct	Subtract direct byte from A with borrow	2 1	95
SUBB	A,@Ri	Subtract indirect RAM from A with borrow	1 1	96, 97
SUBB	A,#data	Subtract immediate data from A with borrow	2 1	94
INC	A	Increment A	1 1	04
INC	Rr	Increment register	1 1	0*
INC	direct	Increment direct byte	2 1	05
INC	@Ri	Increment indirect RAM	1 1	06, 07
DEC	A	Decrement A	1 1	14
DEC	Rr	Decrement register	1 1	1*
DEC	direct	Decrement direct byte	2 1	15
DEC	@Ri	Decrement indirect RAM	1 1	16, 17
INC	DPTR	Increment data pointer	1 2	A3
MUL	AB	Multiply A & B	1 4	A4
DIV	AB	Divide A by B	1 4	84
DA	A	Decimal adjust A	1 1	D4

INSTRUCTION SET (continued)

mnemonic		description	bytes/ cycles	opcode (hex.)
Logic operations				
ANL	A,Rr	AND register to A	1 1	5*
ANL	A,direct	AND direct byte to A	2 1	55
ANL	A,@Ri	AND indirect RAM to A	1 1	56, 57
ANL	A,#data	AND immediate data to A	2 1	54
ANL	direct,A	AND A to direct byte	2 1	52
ANL	direct,#data	AND immediate data to direct byte	3 2	53
ORL	A,Rr	OR register to A	1 1	4*
ORL	A,direct	OR direct byte to A	2 1	45
ORL	A,@Ri	OR indirect RAM to A	1 1	46, 47
ORL	A,#data	OR immediate data to A	2 1	44
ORL	direct,A	OR A to direct byte	2 1	42
ORL	direct,#data	OR immediate data to direct byte	3 2	43
XRL	A,Rr	Exclusive-OR register to A	1 1	6*
XRL	A,direct	Exclusive-OR direct byte to A	2 1	65
XRL	A,@Ri	Exclusive-OR indirect RAM to A	1 1	66, 67
XRL	A,#data	Exclusive-OR immediate data to A	2 1	64
XRL	direct, A	Exclusive-OR A to direct byte	2 1	62
XRL	direct,#data	Exclusive-OR immediate data to direct byte	3 2	63
CLR	A	Clear A	1 1	E4
CPL	A	Complement A	1 1	F4
RL	A	Rotate A left	1 1	23
RLC	A	Rotate A left through the carry flag	1 1	33
RR	A	Rotate A right	1 1	03
RRC	A	Rotate A right through the carry flag	1 1	13
SWAP	A	Swap nibbles within A	1 1	C4

mnemonic		description	bytes/ cycles	opcode (hex.)
Data transfer				
MOV	A,Rr	Move register to A	1 1	E*
MOV	A,direct **	Move direct byte to A	2 1	E5
MOV	A,@Ri	Move indirect RAM to A	1 1	E6, E7
MOV	A,#data	Move immediate data to A	2 1	74
MOV	Rr,A	Move A to register	1 1	F*
MOV	Rr,direct	Move direct byte to register	2 2	A*
MOV	Rr,#data	Move immediate data to register	2 1	7*
MOV	direct,A	Move A to direct byte	2 1	F5
MOV	direct,Rr	Move register to direct byte	2 2	8*
MOV	direct,direct	Move direct byte to direct	3 2	85
MOV	direct,@Ri	Move indirect RAM to direct byte	2 2	86, 87
MOV	direct,#data	Move immediate data to direct byte	3 2	75
MOV	@Ri,A	Move A to indirect RAM	1 1	F6, F7
MOV	@Ri,direct	Move direct byte to indirect RAM	2 2	A6, A7
MOV	@Ri,#data	Move immediate data to indirect RAM	2 1	76, 77
MOV	DPTR,#data16	Load data pointer with a 16-bit constant	3 2	90
MOVC	A,@A+DPTR	Move code byte relative to DPTR to A	1 2	93
MOVC	A,@A+PC	Move code byte relative to PC to A	1 2	83
MOVX	A,@Ri	Move external RAM (8-bit address) to A	1 2	E2, E3
MOVX	A,@DPTR	Move external RAM (16-bit address) to A	1 2	E0
MOVX	@Ri,A	Move A to external RAM (8-bit address)	1 2	F2, F3
MOVX	@DPTR,A	Move A to external RAM (16-bit address)	1 2	F0
PUSH	direct	Push direct byte onto stack	2 2	C0
POP	direct	Pop direct byte from stack	2 2	D0
XCH	A,Rr	Exchange register with A	1 1	C*
XCH	A,direct	Exchange direct byte with A	2 1	C5
XCH	A,@Ri	Exchange indirect RAM with A	1 1	C6, C7
XCHD	A,@Ri	Exchange LOW-order digit indirect RAM with A	1 1	D6, D7

**MOV A,ACC is not a valid instruction.

INSTRUCTION SET (continued)

mnemonic		description	bytes/ cycles		opcode (hex.)
Boolean variable manipulation					
CLR	C	Clear carry flag	1	1	C3
CLR	bit	Clear direct bit	2	1	C2
SETB	C	Set carry flag	1	1	D3
SETB	bit	Set direct bit	2	1	D2
CPL	C	Complement carry flag	1	1	B3
CPL	bit	Complement direct bit	2	1	B2
ANL	C,bit	AND direct bit to carry flag	2	2	82
ANL	C,/bit	AND complement of direct bit to carry flag	2	2	B0
ORL	C,bit	OR direct bit to carry flag	2	2	72
ORL	C,/bit	OR complement of direct bit to carry flag	2	2	A0
MOV	C,bit	Move direct bit to carry flag	2	1	A2
MOV	bit,C	Move carry flag to direct bit	2	2	92
Program and machine control					
ACALL	addr11	Absolute subroutine call	2	2	●1addr
LCALL	addr16	Long subroutine call	3	2	12
RET		Return from subroutine	1	2	22
RETI		Return from interrupt	1	2	32
AJMP	addr11	Absolute jump	2	2	▲1addr
LJMP	addr16	Long jump	3	2	02
SJMP	rel	Short jump (relative address)	2	2	80
JMP	@A+DPTR	Jump indirect relative to the DPTR	1	2	73
JZ	rel	Jump if A is zero	2	2	60
JNZ	rel	Jump if A is not zero	2	2	70
JC	rel	Jump if carry flag is set	2	2	40
JNC	rel	Jump if no carry flag	2	2	50
JB	bit,rel	Jump if direct bit is set	3	2	20
JNB	bit,rel	Jump if direct bit is not set	3	2	30
JBC	bit,rel	Jump if direct bit is set and clear bit	3	2	10
CJNE	A,direct,rel	Compare direct to A and jump if not equal	3	2	B5
CJNE	A,#data,rel	Compare immediate to A and jump if not equal	3	2	B4
CJNE	Rr,#data,rel	Compare immed. to reg. and jump if not equal	3	2	B*
CJNE	@Ri,#data,rel	Compare immed. to ind. and jump if not equal	3	2	B6, B7
DJNZ	Rr,rel	Decrement register and jump if not zero	2	2	D*
DJNZ	direct,rel	Decrement direct and jump if not zero	3	2	D5
NOP		No operation	1	1	00

Notes to Table 7

Data addressing modes

Rr	Working register R0-R7.
direct	128 internal RAM locations and any special function register (SFR).
@Ri	Indirect internal RAM location addressed by register R0 or R1.
#data	8-bit constant included in instruction.
#data16	16-bit constant included as bytes 2 and 3 of instruction.
bit	Direct addressed bit in internal RAM or SFR.
addr16	16-bit destination address. Used by LCALL and LJMP. The branch will be anywhere within the 64 K-byte program memory address space.
addr11	11-bit destination address. Used by ACALL and AJMP. The branch will be within the same 2 K-byte page of program memory as the first byte of the following instruction.
rel	Signed (two's complement) 8-bit offset byte. Used by SJMP and all conditional jumps. Range is −128 to + 127 bytes relative to first byte of the following instruction.

Hexadecimal opcode cross-reference to Table 7

*: 8, 9, A, B, C, D, E, F.

●: 11, 31, 51, 71, 91, B1, D1, F1.

▲: 01, 21, 41, 61, 81, A1, C1, E1.

Table 8 Instruction map

first hexadecimal chacter of opcode ↓ — second hexadecimal character of opcode →

	0	1	2	3	4	5	6 | 7	8 | 9 | A	B	C	D	E	F
0	NOP	AJMP page 0	LJMP addr16	RR A	INC A	INC dir	INC @Ri 0 | 1	INC Rr 0 | 1 | 2	3	4	5	6	7
1	JBC bit, addr8	ACALL page 0	LCALL addr16	RRC A	DEC A	DEC dir	DEC @Ri 0 | 1	DEC Rr 0 | 1 | 2	3	4	5	6	7
2	JB bit, addr8	AJMP page 1	RET	RL A	ADD A,#data	ADD A,dir	ADD A,@Ri 0 | 1	ADD A,Rr 0 | 1 | 2	3	4	5	6	7
3	JNB bit, addr8	ACALL page 1	RETI	RLC A	ADDC A,#data	ADDC A,dir	ADDC A,@Ri 0 | 1	ADDC A,Rr 0 | 1 | 2	3	4	5	6	7
4	JC addr8	AJMP page 2	ORL dir,A	ORL dir,#data	ORL A,#data	ORL A,dir	ORL A,@Ri 0 | 1	ORL A,Rr 0 | 1 | 2	3	4	5	6	7
5	JNC addr8	ACALL page 2	ANL dir,A	ANL dir,#data	ANL A,#data	ANL A,dir	ANL A,@Ri 0 | 1	ANL A,Rr 0 | 1 | 2	3	4	5	6	7
6	JZ addr8	AJMP page 3	XRL dir,A	XRL dir,#data	XRL A,#data	XRL A,dir	XRL A,@Ri 0 | 1	XRL A,Rr 0 | 1 | 2	3	4	5	6	7
7	JNZ addr8	ACALL page 3	ORL C,bit	JMP @A+DPTR	MOV A,#data	MOV dir,#data	MOV @Ri,#data 0 | 1	MOV Rr,#data 0 | 1 | 2	3	4	5	6	7
8	SJMP addr8	AJMP page 4	ANL C,bit	MOVC A,@A+PC	DIV AB	MOV dir,dir	MOV dir,@Ri 0 | 1	MOV dir,Rr 0 | 1 | 2	3	4	5	6	7
9	MOV DPTR, #data 16	ACALL page 4	MOV bit,C	MOVC A,@A+DPTR	SUBB A,#data	SUBB A,dir	SUBB A,@Ri 0 | 1	SUBB A,Rr 0 | 1 | 2	3	4	5	6	7
A	ORL C,/bit	AJMP page 5	MOV C,bit	INC DPTR	MUL AB		MOV @Ri,dir 0 | 1	MOV Rr,dir 0 | 1 | 2	3	4	5	6	7
B	ANL C,/bit	ACALL page 5	CPL bit	CPL C	CJNE A, #data, addr8	CJNE A,dir,addr8	CJNE @Ri,#data, addr8 0 | 1	CJNE Rr,#data, addr8 0 | 1 | 2	3	4	5	6	7
C	PUSH dir	AJMP page 6	CLR bit	CLR C	SWAP A	XCH A,dir	XCH A,@Ri 0 | 1	XCH A,Rr 0 | 1 | 2	3	4	5	6	7
D	POP dir	ACALL page 6	SETB bit	SETB C	DA A	DJNZ dir, addr8	XCHD A,@Ri 0 | 1	DJNZ Rr, addr8 0 | 1 | 2	3	4	5	6	7
E	MOVX A,@DPTR	AJMP page 7	MOVX A,@Ri 0 | 1		CLR A	MOV A,dir	MOV A,@Ri 0 | 1	MOV A,Rr 0 | 1 | 2	3	4	5	6	7
F	MOVX @DPTR,A	ACALL page 7	MOVX @Ri,A 0 | 1		CPL A	MOV dir,A	MOV @Ri,A 0 | 1	MOV Rr,A 0 | 1 | 2	3	4	5	6	7

RATINGS

Limiting values in accordance with the Absolute Maximum System (IEC 134)

parameter	symbol	min.	max.	unit
Input voltage on any pin with respect to ground (V_{SS})	V_I	−0,5	+ 7	V
Total power dissipation	P_{tot}	–	2	W
Input, output current	I_I, I_O	–	10	mA
Storage temperature	T_{stg}	−65	+ 150	°C
Operating ambient temperature range				
MAB series	T_{amb}	0	+ 70	°C
MAF series	T_{amb}	−40	+ 86	°C
MAF80A series	T_{amb}	−40	+ 100	°C

DC CHARACTERISTICS FOR MAB8032AH/52AH

V_{CC} = 5 V (± 10%); V_{SS} = 0 V; T_{amb} = 0 to + 70 °C; all voltages with respect to V_{SS} unless otherwise specified

parameter	conditions	symbol	min.	max.	unit
Supply current	all outputs disconnected; $\overline{EA}$ = V_{CC}	I_{CC}	–	175	mA
Inputs					
Input voltage LOW		V_{IL}	−0,5	0,8	V
Input voltage HIGH all inputs except RST and XTAL2		V_{IH}	2	V_{CC} + 0,5	V
Input voltage HIGH to RST and XTAL2	XTAL1 to V_{SS}	V_{IH1}	2,5	V_{CC} + 0,5	V
Outputs					
Output voltage LOW (Ports 1, 2, 3) (note 1)	I_{OL} = 1,6 mA	V_{OL}	–	0,45	V
Output voltage LOW (Port 0, ALE, $\overline{PSEN}$) (note 1)	I_{OL1} = 3,2 mA	V_{OL1}	–	0,45	V
Output voltage HIGH (Ports 1, 2, 3)	I_{OH} = −80 μA	V_{OH}	2,4	–	V
Output voltage HIGH (Port 0, ALE, $\overline{PSEN}$)	I_{OH1} = −400 μA	V_{OH1}	2,4	–	V
Input leakage current (Port 0, $\overline{EA}$)	0,45 V < V_I < V_{CC}	± I_{LI}	–	10	μA
Input current HIGH (RST)	V_I = V_{CC} −1,5 V	I_{IH1}	–	500	μA
Input current logic 0 (Ports 1, 2, 3)	V_{IL} = 0,45 V	I_{IL}	–	−800	μA
Input current logic 0 (XTAL2)	V_{IL} = 0,45 V; XTAL1 to V_{SS}	I_{IL2}	–	−3,2	mA
Capacitance of I/O buffer	f_c = 1 MHz; T_{amb} = 25 °C	$C_{I/O}$	–	10	pF

DEVELOPMENT DATA

DC CHARACTERISTICS FOR MAF8032AH/52AH and MAF80A32AH/52AH

V_{CC} = 5 V (± 10%); V_{SS} = 0 V; T_{amb} = −40 to + 85 °C (MAF8032AH/52AH) and (MAF80A32AH/52AH).
All voltages measured with respect to V_{SS} unless otherwise specified

parameter	conditions	symbol	min.	max.	unit
Supply current	all outputs disconnected; $\overline{EA} = V_{CC}$	I_{CC}	–	180	mA
Inputs					
Input voltage LOW		V_{IL}	−0,5	0,8	V
Input voltage HIGH all inputs except RST and XTAL 2		V_{IH}	2,2	V_{CC} + 0,5	V
Input voltage HIGH to RST and XTAL 2	XTAL 1 to V_{SS}	V_{IH1}	2,7	V_{CC} + 0,5	V
Outputs					
Output voltage LOW (Ports 1, 2, 3) (note 1)	I_{OL} = 1,2 mA	V_{OL}	–	0,45	V
Output voltage LOW (Port 0, ALE, $\overline{PSEN}$) (note 1)	I_{OL1} = 2,4 mA	V_{OL1}	–	0,45	V
Output voltage HIGH (Ports 1, 2, 3)	I_{OH} = −50 μA	V_{OH}	2,4	–	V
Output voltage HIGH (Port 0, ALE, $\overline{PSEN}$)	I_{OH1} = −360 μA	V_{OH1}	2,4	–	V
Input leakage current (Port 0, $\overline{EA}$)	0,45 V < V_I < V_{CC}	± I_{LI}	–	10	μA
Input current HIGH (RST)	$V_I = V_{CC}$−1,5 V	I_{IH1}	–	500	μA
Input current logic 0 (Ports 1, 2, 3)	V_{IL} = 0,45 V	I_{IL}	–	−800	μA
Input current logic 0 (XTAL 2)	V_{IL} = 0,45 V; XTAL 1 to V_{SS}	I_{IL2}	–	−3,5	mA
Capacitance of I/O buffer	f_c = 1 MHz; T_{amb} = 25 °C	$C_{I/O}$	–	10	pF

Note 1

V_{OL} is degraded when the MAB8052AH rapidly discharges external capacitance.
This AC noise is most pronounced during emission of address data. When using external memory, locate the latch or buffer as close to the MAB8052AH as possible.

datum	emitting ports	time slot internal	degraded I/O lines	V_{OL} (max.)
address	P2, P0	TS3, TS9	P1, P3	0,8 V
write data	P0	TS6	P1, P3, ALE	0,8 V

Table 9 Available frequency range with respect to ambient operating temperature.

type	T_{amb}		clock frequency
	min.	max.	
MAB8032AH/52AH	0 °C	70 °C	3,5 to 12 MHz
MAF8032AH/52AH	–40 °C	85 °C	3,5 to 10 MHz
MAF80A32AH/A52AH	–40 °C	100 °C	3,5 to 10 MHz

DEVELOPMENT DATA

AC CHARACTERISTICS

V_{CC} = 5 V (±10%); V_{SS} = 0 V; T_{amb} = same for all three temperature ranges; C_L = 80 pF for all other outputs unless otherwise specified (see waveforms of Figs 21, 22 and 23).

parameter	symbol	10 MHz		12 MHz		variable clock (note 1)		unit
		min.	max.	min.	max.	min.	max.	
Program memory								
ALE pulse duration	t_{LL}	160	–	127	–	$2t_{CK}-40$	–	ns
Address set-up time to ALE	t_{AL}	60	–	43	–	$t_{CK}-40$	–	ns
Address hold time after ALE	t_{LA}	65	–	48	–	$t_{CK}-35$	–	ns
Time from ALE to valid instruction input	t_{LIV}	–	300	–	233	–	$4t_{CK}-100$	ns
Time from ALE to control pulse $\overline{PSEN}$	t_{LC}	75	–	58	–	$t_{CK}-25$	–	ns
Control pulse duration $\overline{PSEN}$	t_{CC}	265	–	215	–	$3t_{CK}-35$	–	ns
Time from $\overline{PSEN}$ to valid instruction input	t_{CIV}	–	175	–	125	–	$3t_{CK}-125$	ns
Input instruction hold time after $\overline{PSEN}$	t_{CI}	0	–	0	–	0	–	ns
Input instruction float delay after $\overline{PSEN}$ (note 2)	t_{CIF}	–	80	–	63	–	$t_{CK}-20$	ns
Address valid after $\overline{PSEN}$ (note 2)	t_{AC}	92	–	75	–	$t_{CK}-8$	–	ns
Address to valid instruction input	t_{AIV}	–	385	–	302	–	$5t_{CK}-115$	ns
Address float time to $\overline{PSEN}$	t_{AFC}	−12	–	−12	–	−12	–	ns

AC CHARACTERISTICS (continued)

parameter	symbol	10 MHz		12 MHz		variable clock (note 1)		unit
		min.	max.	min.	max.	min.	max.	
External data memory								
$\overline{RD}$ pulse duration	t_{RR}	500	–	400	–	$6t_{CK}-100$	–	ns
$\overline{WR}$ pulse duration	t_{WW}	500	–	400	–	$6t_{CK}-100$	–	ns
Address hold time after ALE	t_{LA}	65	–	48	–	$t_{CK}-35$	–	ns
RD to valid data input	t_{RD}	–	335	–	250	–	$5t_{CK}-165$	ns
Data hold time after $\overline{RD}$	t_{DR}	0	–	0	–	0	–	ns
Data float delay after $\overline{RD}$	t_{DFR}	–	130	–	97	–	$2t_{CK}-70$	ns
Time from ALE to valid data input	t_{LD}	–	650	–	517	–	$8t_{CK}-150$	ns
Address to valid data input	t_{AD}	–	735	–	585	–	$9t_{CK}-165$	ns
Time from ALE to $\overline{RD}$ or $\overline{WR}$	t_{LW}	250	350	200	300	$3t_{CK}-50$	$3t_{CK}+50$	ns
Time from address to $\overline{RD}$ or $\overline{WR}$	t_{AW}	270	–	203	–	$4t_{CK}-130$	–	ns
Time from $\overline{RD}$ or $\overline{WR}$ HIGH to ALE HIGH	t_{WHLH}	60	140	43	123	$t_{CK}-40$	$t_{CK}+40$	ns
Data valid to $\overline{WR}$ transition	t_{DWX}	40	–	23	–	$t_{CK}-60$	–	ns
Data set-up time before $\overline{WR}$	t_{DW}	550	–	433	–	$7t_{CK}-150$	–	ns
Data hold time after $\overline{WR}$	t_{WD}	50	–	33	–	$t_{CK}-50$	0	ns
Address float delay after $\overline{RD}$	t_{AFR}	–	12	–	12	–	12	ns

Notes to the AC characteristics

1. $1/t_{CK}$ = 3,5 to 10/12 MHz (see Fig. 20 and Table 10).
2. Interfacing the MAB8052AH to devices with float times up to 75 ns is permitted. This limited bus contention will not cause damage to port 0 drivers.

DEVELOPMENT DATA

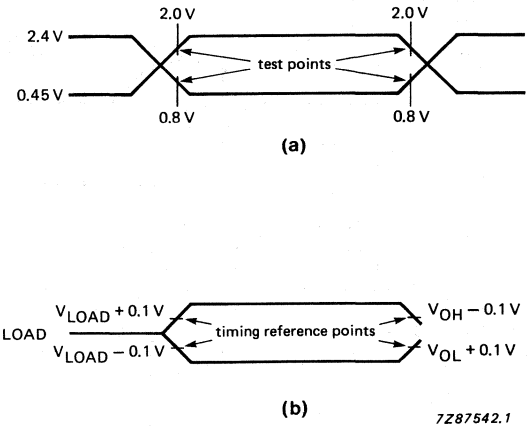


Fig. 19 AC testing input, output waveform (a) and float waveform (b).

AC testing inputs are driven at 2,4 V for a logic 1 and 0,45 V for a logic 0. Timing measurements are taken at 2,0 V for a logic 1 and 0,8 V for logic 0. With reference to Fig. 19(b), for timing purposes a port pin is considered no longer floating when a 100 mV change from load voltage occurs and is said to begin floating when a 100 mV change from the loaded V_{OH}/V_{OL} level occurs; $I_{OL}/I_{OH} \geqslant \pm 20$ mA.

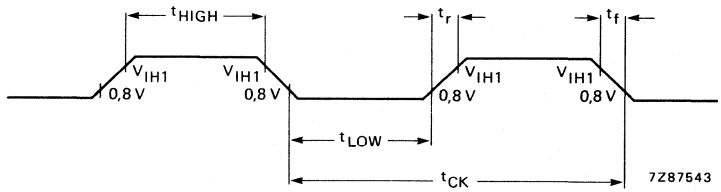


Fig. 20 External clock drive XTAL 2 (see Table 10).

Table 10 External clock drive XTAL2 (see Fig. 20)

parameter	symbol	variable clock * (f = 3,5 to 12 MHz)		unit
		min.	max.	
oscillator clock period	t_{CK}	83,3	286	ns
HIGH time	t_{HIGH}	20	$t_{CK}-t_{LOW}$	ns
LOW time	t_{LOW}	20	$t_{CK}-t_{HIGH}$	ns
rise time	t_r	–	20	ns
fall time	t_f	–	20	ns

* 3,5 - 10 MHz for:
MAF8032AH/52AH
MAF80A32AH/52AH
versions.

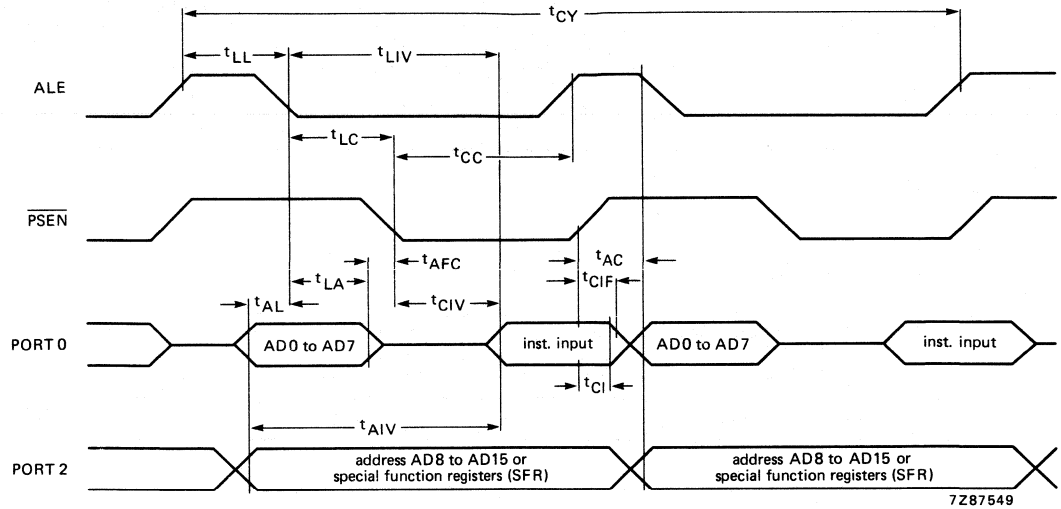


Fig. 21 Read from program memory.

DEVELOPMENT DATA

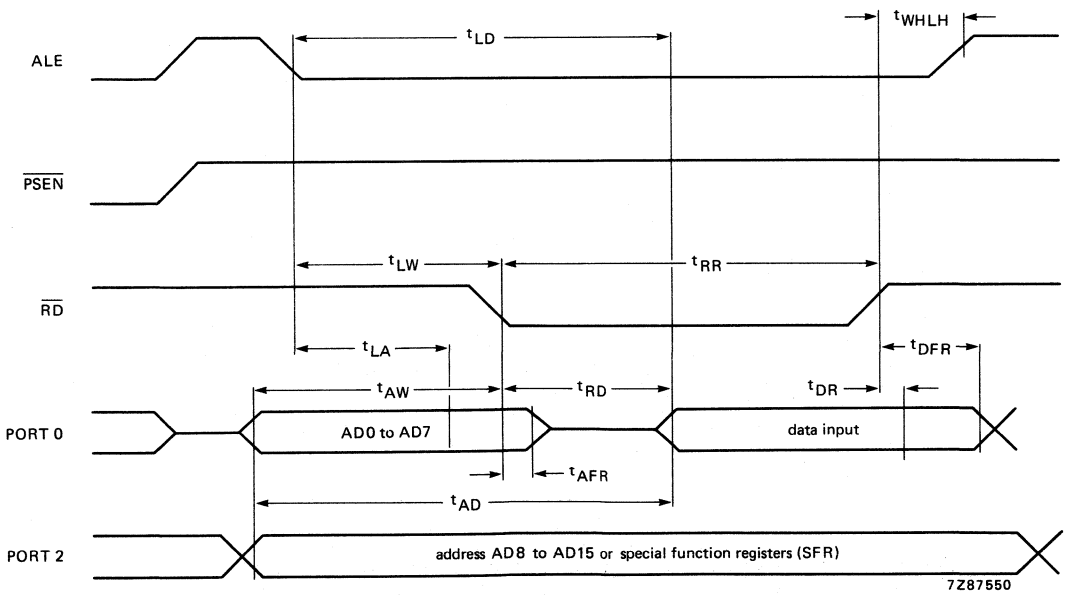


Fig. 22 Read from data memory.

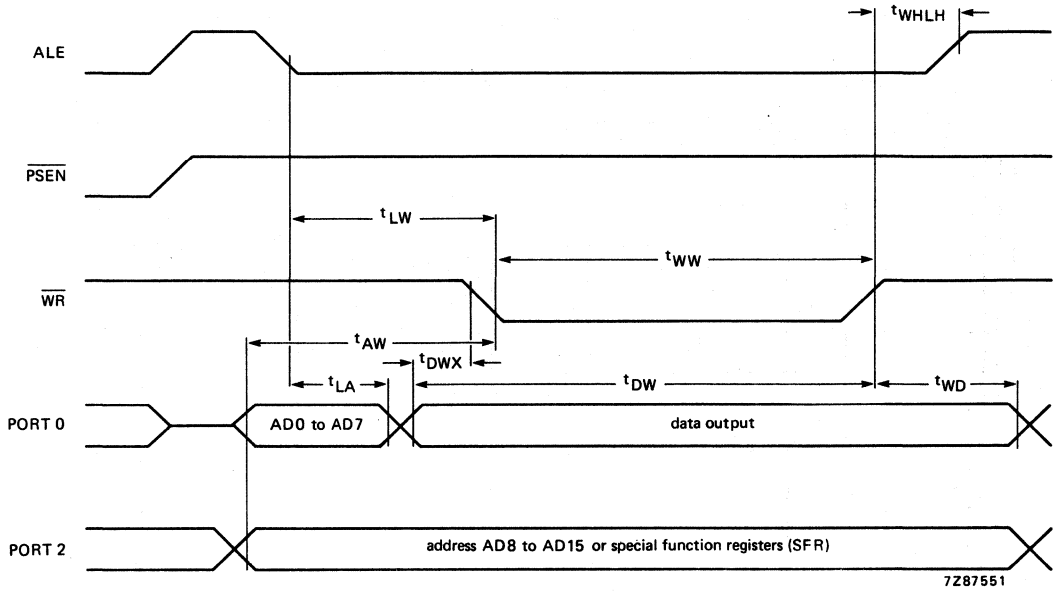


Fig. 23 Write to data memory.

DEVELOPMENT DATA

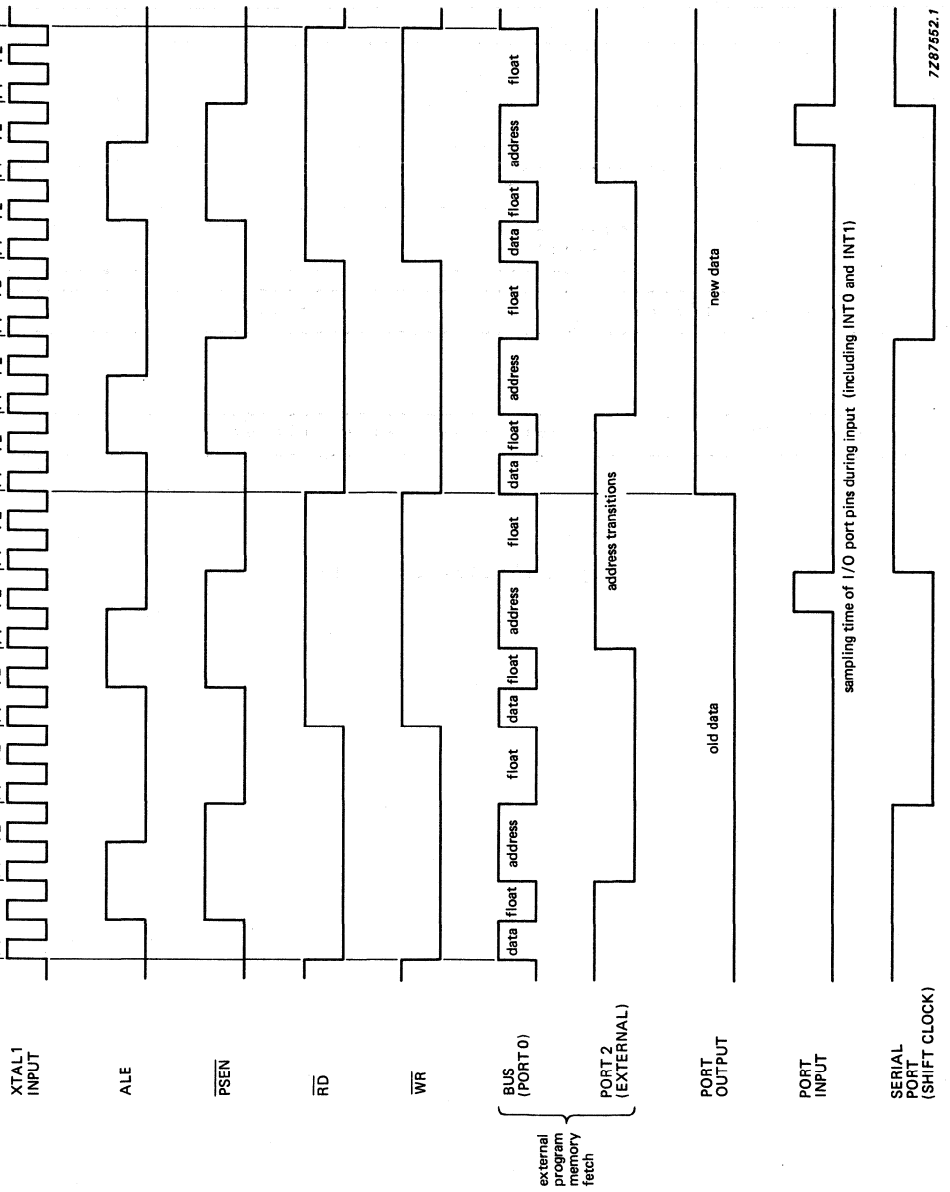


Fig. 24 Instruction cycle timing.

SINGLE-CHIP 8-BIT MICROCONTROLLER

GENERAL DESCRIPTION

The MAB80XXH family of single-chip 8-bit microcontrollers is fabricated in NMOS.
Three interchangeable (pin compatible) versions are available:

- MAB8048H: 1 K bytes mask-programmed ROM, 64 bytes RAM
- MAB8035HL: ROM-less version of the MAB8048H
- MAB8049H: 2 K bytes mask-programmed ROM, 128 bytes RAM
- MAB8039HL: ROM-less version of the MAB8049H
- MAB8050H: 4 K bytes mask-programmed ROM, 256 bytes RAM
- MAB8040HL: ROM-less version of the MAB8050H

These microcontrollers are designed to be efficient control processors as well as arithmetic processors. Their instruction set allows the user to directly set and reset individual I/O lines as well as test individual bits within the accumulator. A large variety of branch and table look-up instructions enable efficient implementation of standard logic functions. Code efficiency is high; over 70% of the instructions are single byte; all others are two byte.

An on-chip 8-bit counter is provided, which can count either machine cycles (÷32) or external events. The counter can be used to generate an interrupt to the processor.
Program and data memories plus input/output capabilities can be expanded using standard TTL compatible memories and logic. For more detailed information see the 'single-chip 8-bit microcontrollers' user manual.

Features

- 8-bit CPU, ROM, RAM and I/O
- 8-bit counter/timer
- On-chip oscillator and clock driver circuits
- Single-level interrupts: external and counter/timer
- 17 internal registers: accumulator, 16 addressable registers
- Over 90 instructions: 70% single byte
- All instructions 1 or 2 cycles
- Easily expandable memory and 27 I/O lines
- TTL compatible inputs and outputs
- Single 5 V supply
- Standard and extended temperature ranges (see Table 5): MAB80XX: 0 to +70 °C
  MAF80XX: −40 to +85 °C
  MAF80AXX: −40 to +110 °C

Applications

- Peripheral interfaces and controllers
- Test and measuring instruments
- Sequencers
- Modems and data enciphering
- Environmental control systems
- Audio/video systems

PACKAGE OUTLINES

All versions: with type no. suffix P (see Table 5): 40-lead DIL; plastic (SOT-129).
MAB8035/8048/8039/8049H/HLWP : 44-lead PLCC; plastic leaded chip-carrier 'pocket' version (SOT-187AA), 'pedestal' version (SOT-187).

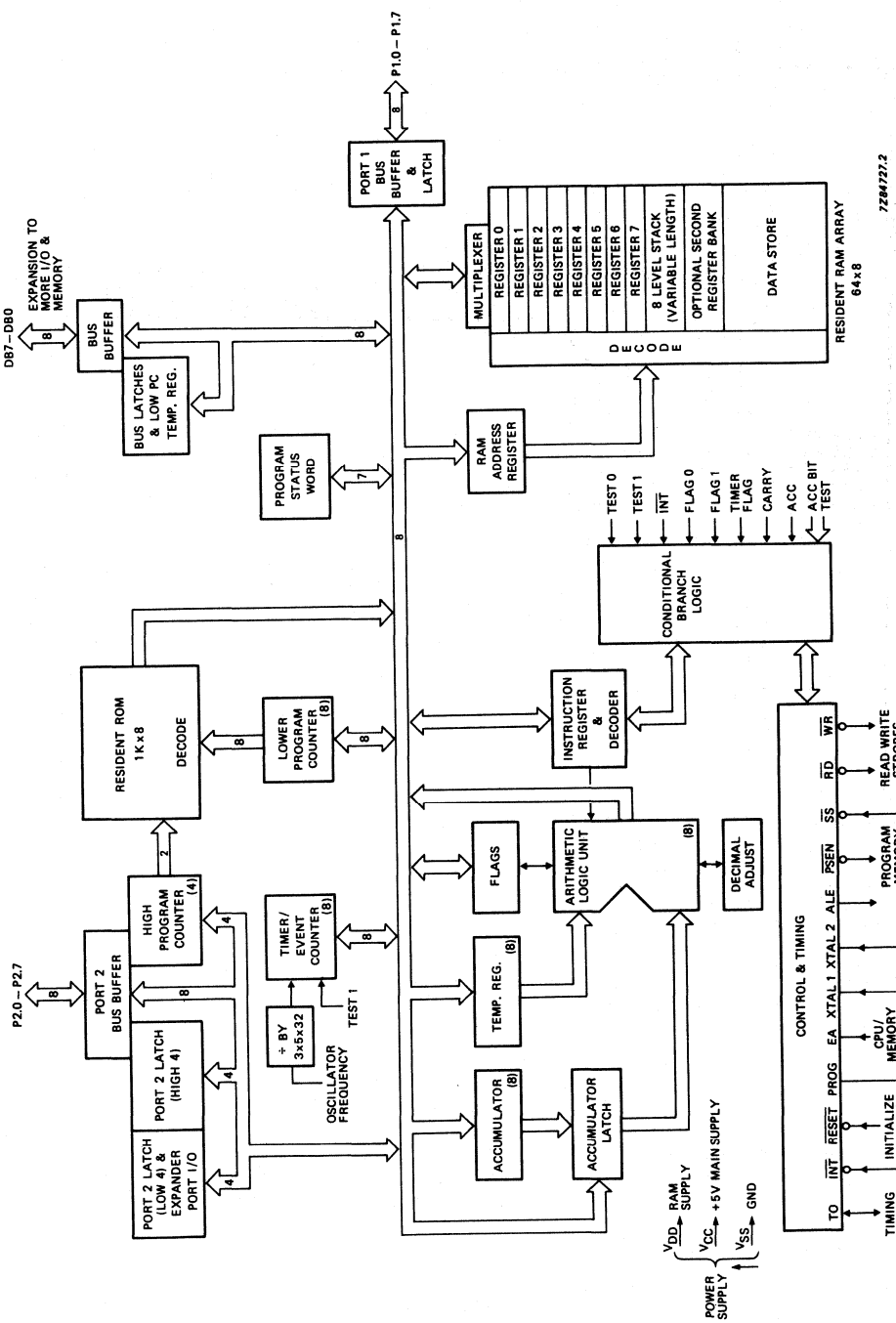


Fig. 1 Block diagram.

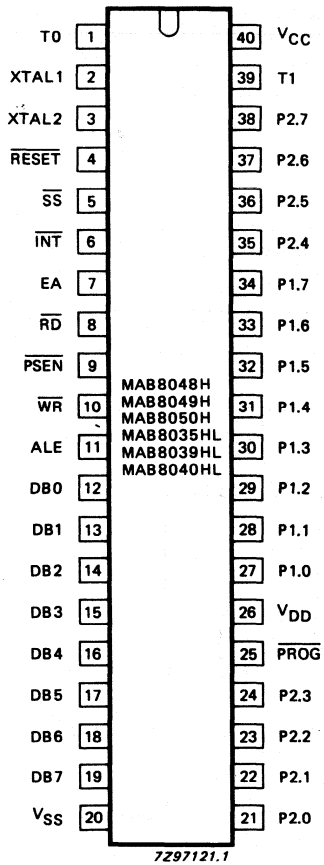


Fig. 2a Pinning diagram; for pin designation see next page.

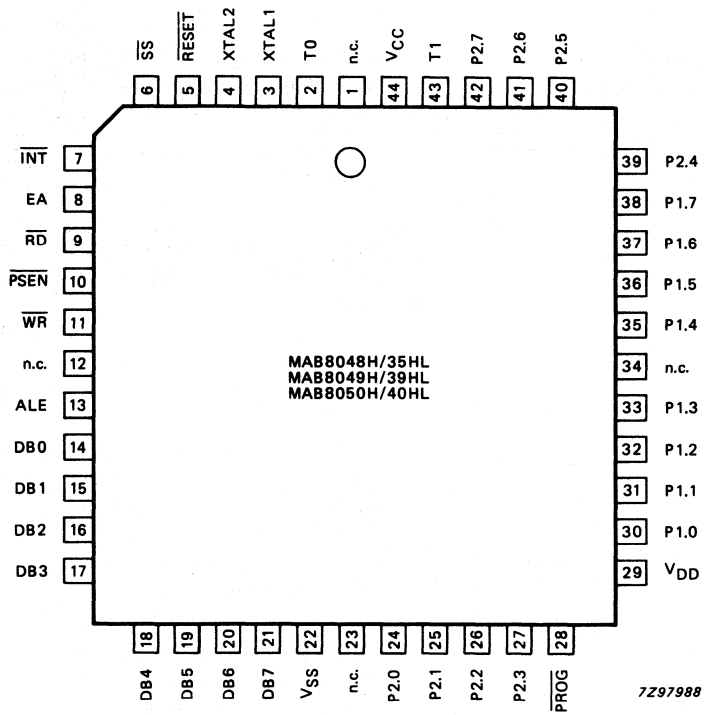


Fig. 2b Pinning diagram for MAB80XXHWP; for pin designation see next page.

PINNING

12–19	DB0–DB7	**Data Bus:** true bidirectional I/O port which can be written or read using the $\overline{RD}$ and $\overline{WR}$ strobes. This port can also be used as an 8-bit latch. It contains the 8 low order address bits during an external memory access and receives the addressed instruction under control of $\overline{PSEN}$. This multiplexed address/data port also contains the address and data during external RAM accesses.
27–34	P1.0–P1.7	**Port 1:** 8-bit quasi-bidirectional I/O port (note 1).
21–24	P2.0–P2.7	**Port 2:** 8-bit quasi-bidirectional I/O port (note 1).
35–38		P2.0–P2.3 contains the 4 higher order address bits during an external program memory access and provides a 4-bit bus for 8243 I/O expanders.
25	$\overline{PROG}$	**Output strobe:** active LOW for 8243 I/O expanders.
1	T0	**Test 0:** input pin which can be tested by the JT0 and JNT0 instructions. **Clock:** T0 can be configured as a clock output using the ENT0 CLK instruction.
39	T1	**Test 1:** input pin which can be tested using the JT1 and JTN1 instructions. T1 can be configured as the timer/counter input using the STRT CNT instruction.
6	$\overline{INT}$	**Interrrupt:** interrupt input pin which can initiate an interrupt if the external interrupt is enabled. Can also be tested using the JNI instruction. Interrupt is disabled during and after $\overline{RESET}$.
4	$\overline{RESET}$	**Reset:** active LOW input used to initialize the microcontroller. During program verification, the address is latched by a 0– to –1 transition on $\overline{RESET}$ and the data at the addressed location is output on BUS (note 2).
11	ALE	**Address latch enable:** occurs each cycle and is useful as a clock output. During an external program or data memory access, ALE is used to latch the address information multiplexed on the DB0 to DB7 outputs.
8	$\overline{RD}$	**Read BUS:** active LOW strobe used to gate data on to BUS lines when reading from an external source.
10	$\overline{WR}$	**Write BUS:** active LOW strobe used to write data from BUS lines to an external designation.
7	EA	**External access input:** when HIGH, all instruction fetches are from external memory.
9	$\overline{PSEN}$	**Program store enable:** active LOW strobe that occurs only during a fetch from external memory.
5	$\overline{SS}$	**Single step:** active LOW input used with ALE to cause the microcontroller to execute a single instruction.
2 3	XTAL 1 XTAL 2	**Crystal inputs:** inputs for a crystal, LC-network or an external timing signal to determine the internal oscillator frequency (note 2).
20	V_{SS}	**Ground:** circuit earth potential.
40	V_{CC}	**Power supply:** + 5 V main power supply pin.
26	V_{DD}	**Power supply:** + 5 V RAM standby power supply; low power

Notes

1. Each port line can be individually configured as an input or an output. A line is designated as an input by first writing a logic 1 to the line. $\overline{\text{RESET}}$ sets all port lines to logic 1.
2. Non-standard TTL V_{IH}.

FUNCTIONAL DESCRIPTION

The following sections provide a detailed functional description of the MAB80XXH microcontroller as shown in Fig. 1. The generic term "MAB80XXH" is used to refer collectively to the MAB8048H/35HL, MAB8049H/39HL and MAB8050H/40HL.

Program memory (see Fig. 3)

The on-chip program memory consists of 1024, 2048 or 4096 bytes of mask programmed ROM (MAB8048H/49H/50H); the MAB8035HL/39HL/40HL versions do not have on-chip program memory. The total addressing capability is 4096 bytes.
The program memory address space is divided into two 2048-byte banks MB0 and MB1. These two 2048 byte banks are each divided into 8 pages of 256 bytes for conditional branches. There are three locations in program memory of special interest. These are:

- Location 0 — contains the first instruction to be executed after a $\overline{\text{RESET}}$
- Location 3 — contains the first instruction of an external interrupt routine
- Location 7 — contains the first instruction of a timer/counter interrupt routine

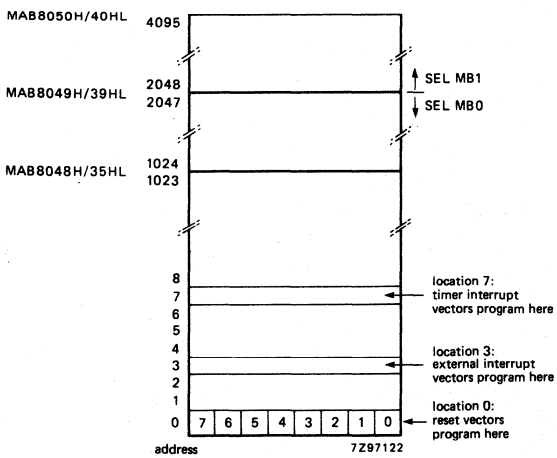


Fig. 3 Program memory map.

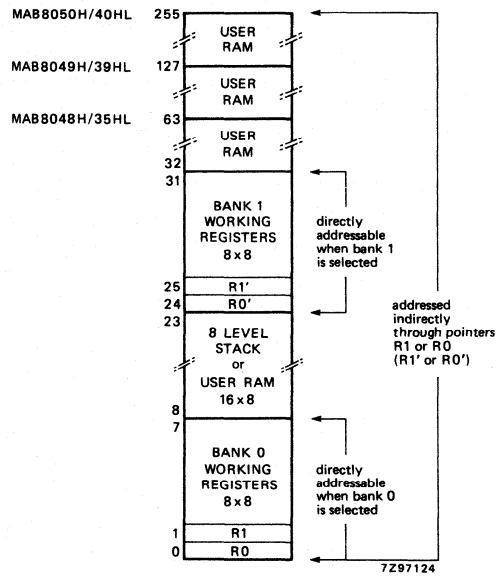


Fig. 4 Data memory map.

FUNCTIONAL DESCRIPTION (continued)

Data memory (see Fig. 4)

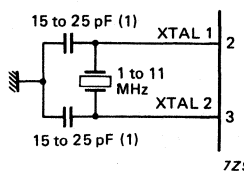
The on-chip data memory consists of a 64, 128 or 256 byte RAM. All locations are indirectly addressable using two RAM pointer registers R0, R1 or R0', R1'. The first 8 RAM locations (0 to 7) are designated as working register bank 0 and are directly addressable. By selecting register bank 1, RAM locations 24 to 31 become the working registers. RAM locations 8 to 23 are designated as the stack. Two bytes are used per CALL allowing up to 8 levels of subroutine nesting.
An extra 256 bytes of RAM may be added and addressed directly using the MOVX instructions.
If more RAM is required, I/O port lines may be used to select additional (256 byte) banks of external memory.

Program counter and stack

The program counter (PC) is a 12-bit counter/register that points to the location from which the next instruction is to be fetched. When EA is logic 0 the PC can address locations 0 to 1023 (8048H), 2047 (8049H) or 4095 (8050H) of internal program memory. At the 1 K (8048H), 2 K (8049H) boundary, an automatic switch-over to external memory occurs. When EA is logic 1 all fetches are from external program memory. The total address space is 4 K bytes. An interrupt or subroutine CALL causes the contents of the program counter to be stored in one of the 8 register pairs of the program counter stack. A 3-bit stack pointer which is part of the program status word (PSW) points to the relevant register pair. Data RAM locations 8 to 23 are available as stack registers and are used to store the program counter and 4 bits of the PSW register. The stack pointer, when initialized to 000, points to RAM locations 8 and 9. The first subroutine jump or interrupt results in the program counter contents being transferred to locations 8 an 9 of the RAM array. The stack pointer is then incremented by one to point to locations 10 and 11. Nesting of subroutines within subroutines can continue up to 8 times without overflowing the stack. If overflow does occur the deepest address stored (locations 8 and 9) will be overwritten and lost since the stack pointer overflows from 111 to 000. It also underflows from 000 to 111. The end of a subroutine, which is signalled by a return instruction (RET or RETR), causes the stack pointer to be decremented and the contents of the appropriate register pair to be transferred to the program counter.

Oscillator and clock (see Figs 5, 6 and 7)

The MAB80XXH has on-chip oscillator and clock driver circuitry. A crystal, LC-network or external timing signal (pulse generator) determines the oscillator frequency. The output of the oscillator is divided-by-three and is available at T0 (pin 1) by executing the ENT0 CLK instruction. This clock signal (CLK) is divided-by-five to define a machine (instruction) cycle. It is available at ALE (pin 11).



(1) Including crystal-socket stray capacities.

Fig. 5 Crystal oscillator mode. Crystal series impedance should be $< 75\ \Omega$ at 6 MHz and $< 180\ \Omega$ at 3,6 MHz. When using a ceramic oscillator both capacitors should be 30 pF.

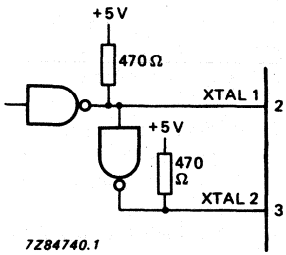
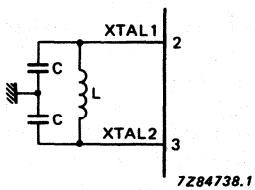


Fig. 6 External clock source. Both XTAL 1 and XTAL 2 should be driven. Resistors to V_{CC} (+ 5 V) are required to ensure V_{IH} = 3,8 V if TTL circuitry is used.
The minimum HIGH and LOW times are 45%.



$$f \approx \frac{1}{2\pi\sqrt{LC'}} \; ; \; C' = \frac{C + 3C_{pp}}{2}$$

L (μH)	C (pF)	nom. f (MHz)
45	20	5,2
120	20	3,2

Fig. 7 LC oscillator. Each capacitor should be ≈ 20 pF including stray capacitance C_{pp} ≈ 5 to 10 pF (pin-to-pin capacitance).

Timer/event counter

An internal counter is available which can count either external events or machine cycles (÷32). The machine cycles are divided-by-32 before they are applied to the input of the 8-bit counter. External events are applied directly to the input of the counter. The maximum clock rate is one third of the machine cycle frequency. The minimum positive duty cycle that can be detected is 0,2 times the cycle period. The counter can be configured to generate an interrupt to the processor when it overflows.

Interrupt

An interrupt may be generated by:

- An external input $\overline{INT}$ (pin 6)
  or
- A timer/counter overflow, when enabled.

In either event, the processor completes execution of the present instruction and then calls the interrupt service routine.
At the end of the interrupt service routine, a RETR instruction restores the machine to the state it was in prior to the interrupt. The external interrupt has priority over the timer/counter interrupt.

Input/output

The MAB80XXH has 27 I/O lines arranged as three 8-bit ports and 3 'test' inputs that can alter program sequences when tested by conditional jump instructions.
Each port line can be individually configured as an input or output.

FUNCTIONAL DESCRIPTION (continued)

Ports 1 and 2 are both 8-bits wide and have identical characteristics. Data written to these ports is latched and remains unchanged until rewritten. In the input mode, these ports are non-latching; inputs must be present until read by an input instruction. Inputs are fully TTL compatible and outputs will drive one standard TTL load.

Ports 1 and 2 are called quasi-bidirectional because they are not high impedance when configured as inputs. Each line is pulled up to + 5 V through a resistor (≈ 50 Ω). This pull-up provides sufficient source current for a TTL HIGH level, yet can be pulled LOW by a standard TTL gate, thus allowing the pin to be used both as an input and an output. To provide fast switching times during a logic 0 - to - 1 transition, transistor TR 2 is switched on for one fifth of a machine cycle when a logic 1 is written to the line. When a logic 0 is written, transistor TR 1 overcomes the pull-up and provides TTL current sinking capability. Since the pull-down transistor is low impedance, a logic 1 must first be written to any line which is to used as an input. $\overline{\text{RESET}}$ initializes all lines to the high impedance logic 1 state. This structure allows input and output on the same pin. Individual port lines can be read and written using the ANL and ORL instructions.

BUS (DB0–DB7)

BUS is a true bidirectional 8-bit port with associated input and output strobes. The BUS port can operate in three different modes: as a latched I/O port, as a bidirectional bus port, or in an expanded system as a program memory address output port. If the bidirectional feature is not needed, BUS can serve as either a statically latched output port or non-latching input port. Input and output lines on this port cannot be mixed. The BUS port lines are either active HIGH, active LOW, or high impedance (floating).

As a static port, data is written and latched using the OUTL instruction and input using the INS in instruction. The INS and OUTL instructions generate pulses on the corresponding $\overline{\text{RD}}$ and $\overline{\text{WR}}$ output strobe lines; however, in the static port mode they are generally not used. As a bidirectional port, the MOVX instructions are used to read and write the port. A write to the port generates a pulse on the $\overline{\text{WR}}$ output line and output data is valid at the trailing edge of $\overline{\text{WR}}$. A read of the port generates a pulse on the $\overline{\text{RD}}$ output line and input data must be valid at the trailing edge of $\overline{\text{RD}}$.

The latched mode (INS, OUTL) is intended for use in the single-chip configuration, where BUS is not being used as an expanded port. OUTL and MOVX instructions can be mixed if required. However, when using a MOVX instruction a previously latched output will be lost and BUS will be left in the high impedance state. INS does not put the BUS in a high impedance state. Therefore, in order to read an external byte (and not the previously latched value) using an INS instruction, it is necessary to precede INS with a MOVX instruction.

OUTL should never be used in a system with external program memory, since latching BUS may cause the next instruction to be incorrectly fetched.

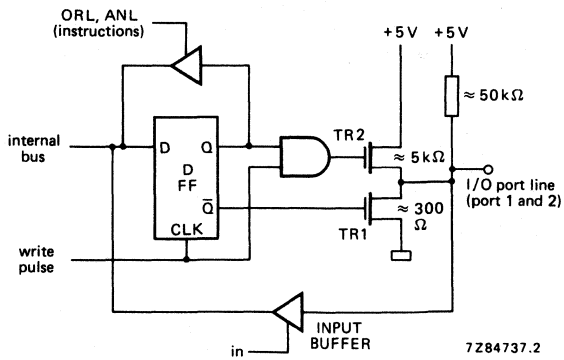


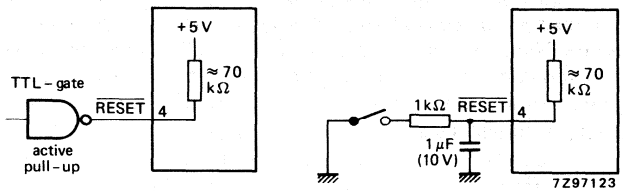
Fig. 8 Quasi-bidirectional port structure.

Test (T0, T1) and $\overline{INT}$

These three pins serve as inputs and may be tested by the conditional jump instruction. They allow inputs to cause program branches without the necessity of loading an input port into the accumulator.

$\overline{RESET}$ (see Fig. 9)

This active LOW input is used to initialize the microcontroller.
This Schmitt-trigger input has an internal pull-up resistor which, in combination with an external 1 μF capacitor, provides an internal reset pulse of sufficient duration to reset all circuitry. If the reset pulse is generated externally, the reset pin must be held at ground (0,45 V) for at least 10 ms after the power supply is within tolerance. Only 5 machine cycles (12,5 μs at 6 MHz) are required if power is already on and the oscillator has stabilized.



(a) (b)

Fig. 9 An external reset is shown in (a) and power-on reset in (b).

Single step ($\overline{SS}$)

This active LOW input when used in combination with ALE will cause the microcontroller to execute a single instruction, then wait until $\overline{SS}$ is reactivated.

Power-down mode (see Fig. 10)

In the MAB80XXH, power can be removed from all but the data RAM array, for low power standby operation. In the power-down mode the contents of the data RAM can be maintained while drawing typically 10% to 15% of the normal operating supply voltage. V_{CC} serves as the + 5 V supply pin for the bulk of the circuitry, while the V_{DD} pin supplies only the RAM array. In normal operation, both pins are at + 5 V. In the standby mode, V_{CC} is at ground and only V_{DD} is maintained at + 5 V. Applying $\overline{RESET}$ to the microcontroller through the reset pin inhibits any access to the RAM and ensures that the RAM cannot be inadvertently altered as power is removed from V_{CC}.

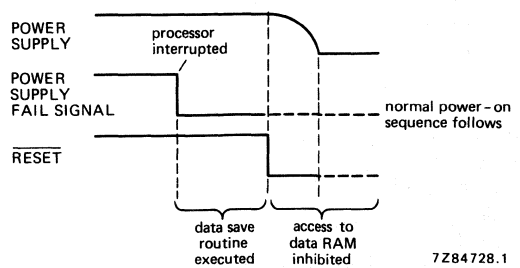


Fig. 10 Power down sequence.

FUNCTIONAL DESCRIPTION (continued)

Instruction set (see Tables 1, 2, 3 and 4)

The MAB80XXH instruction set consists of over 90 one and two-byte instructions. Program code efficiency is high because:

- Working registers and program variables are stored in RAM locations 0 to 127, which require only a single byte to address
- Program memory is divided into pages of 256 bytes, which means that branch destination addresses require only one byte

The instruction set performs logical, arithmetic and test operations on bytes. It also manipulates and tests bits. A set of MOVE instructions operate indirectly on either RAM or ROM, which permits efficient access of pointers and data tables. The indirect jump instruction performs a multi-way branch (up to 256) on the contents of the accumulator to addresses stored in a look-up table. The 'decrement register and jump if not zero' instruction saves a byte each time it is used as opposed to using separate increment and test instructions. The on-chip counter provides the facility for external events or time to be counted by hardware which does not interfere with the main program. The MAB80XXH can either test the counter (under program control) or cause its overflow to generate an interrupt. These features are essential for real-time applications.

Table 1 Symbols and definitions used in Table 2.

symbol	definition description
A	accumulator
addr	program memory address
Bb	bit designation (b = 0—7)
RBS	register bank select
C	carry (bit CY)
CNT	event counter
D	mnemonic for 4-bit digit (nibble)
data	8-bit number or expression
I	interrupt
MB	memory bank
MBFF	memory bank flip-flop
P	mnemonic for 'in-page' operation
PC	program counter
Pp	port designation (p = 0, 1, 2)
PSW	program status word
RB	register bank
Rr	register designation (r = 0—7)
Sn	serial I/O register
SP	stack pointer
T	timer
TF	timer flag
T0	test 0 input
T1	test 1 input
#	immediate data prefix
@	indirect address prefix
(X)	contents of X
((X))	contents of location addressed by X
←	is replaced by
↔	is exchanged with

Table 2 Instruction set

	mnemonic	opcode (hex.)	bytes/ cycles	description	function		notes
ACCUMULATOR	ADD A, Rr	6*	1/1	Add register contents to A	(A)←(A) + (Rr)	r = 0–7	1
	ADD A, @Rr	60 61	1/1	Add RAM data, addressed by Rr, to A	(A)←(A) + ((R0)) (A)←(A) + ((R1))		1
	ADD A, #data	03 data	2/2	Add immediate data to A	(A)←(A) + data		1
	ADDC A, Rr	7*	1/1	Add carry and register contents to A	(A)←(A) + (Rr) + (C)	r = 0–7	1
	ADDC A, @Rr	70 71	1/1	Add carry and RAM data, addresses by Rr, to A	(A)←(A) + ((R0)) + (C) (A)←(A) + ((R1)) + (C)		1
	ADDC A, #data	13 data	2/2	Add carry and immediate data to A	(A)←(A) + data + (C)		1
	ANL A, Rr	5*	1/1	'AND' Rr with A	(A)←(A) AND (Rr)	r = 0–7	
	ANL A, @Rr	50 51	1/1	'AND' RAM data, addressed by Rr, with A	(A)←(A) AND ((R0)) (A)←(A) AND ((R1))		
	ANL A, #data	53 data	2/2	'AND' immediate data with A	(A)←(A) AND data		
	ORL A, Rr	4*	1/1	'OR' Rr with A	(A)←(A) OR (Rr)	r = 0–7	
	ORL A, @Rr	40 41	1/1	'OR' RAM data, addressed by Rr, with A	(A)←(A) OR ((R0)) (A)←(A) OR ((R1))		
	ORL A, #data	43 data	2/2	'OR' immediate data with A	(A)←(A) OR data		
	XRL A, Rr	D*	1/1	'XOR' Rr with A	(A)←(A) XOR (Rr)	r = 0–7	
	XRL A, @Rr	D0 D1	1/1	'XOR' RAM, addressed by Rr, with A	(A)←(A) XOR ((R0)) (A)←(A) XOR ((R1))		
	XRL A, #data	D3 data	2/2	'XOR' immediate data with A	(A)←(A) XOR data		
	INC A	17	1/1	increment A by 1	(A)←(A) + 1		
	DEC A	07	1/1	decrement A by 1	(A)←(A) − 1		
	CLR A	27	1/1	clear A to zero	(A)←0		
	CPL A	37	1/1	one's complement A	(A)←NOT(A)		
	RL A	E7	1/1	rotate A left	$(A_{n+1})\leftarrow(A_n)$ $(A_0)\leftarrow(A_7)$	n = 0–6	

	mnemonic	opcode (hex.)	bytes/ cycles	description	function		notes
ACCUMULATOR (cont.)	RLC A	F7	1/1	rotate A left through carry	$(A_{n+1})\leftarrow(A_n)$ $(A_0)\leftarrow(C)$, $(C)\leftarrow(A_7)$	n = 0–6	2
	RR A	77	1/1	rotate A right	$(A_n)\leftarrow(A_{n+1})$ $(A_7)\leftarrow(A_0)$	n = 0–6	
	RRC A	67	1/1	rotate A right through carry	$(A_n)\leftarrow(A_{n+1})$ $(A_7)\leftarrow(C)$, $(C)\leftarrow(A_0)$	n = 0–6	2
	DA A	57	1/1	decimal adjust A			2
	SWAP A	47	1/1	swap nibbles of A	$(A_{4-7})\leftrightarrow(A_{0-3})$		
DATA MOVES	MOV A, Rr	F*	1/1	move register contents to A	(A)←(Rr)	r = 0–7	
	MOV A, @Rr	F0 F1	1/1	move RAM data, addressed by Rr, to A	(A)←((R0)) (A)←((R1))		
	MOV A, #data	23 data	2/2	move immediate data to A	(A)←data		
	MOV Rr, A	A*	1/1	move accumulator contents to register	(Rr)←(A)	r = 0–7	
	MOV @Rr, A	A0 A1	1/1	move accumulator contents to RAM location addressed by Rr	((R0))←(A) ((R1))←(A)		
	MOV Rr, #data	B* data	2/2	move immediate data to Rr	(Rr)←data	r = 0–7	
	MOV @rR, #data	B0 data B1 data	2/2	move immediate data to RAM location addressed by Rr	((R0))←data ((R1))←data		
	XCH A, Rr	2*	1/1	exchange accumulator contents with Rr	(A)↔(Rr)	r = 0–7	
	XCH A, @Rr	20 21	1/1	exchange accumulator contents with RAM data addressed by Rr	(A)↔((R0)) (A)↔((R1))		
	XCHD A, @Rr	30 31	1/1	exchange lower nibbles of A and RAM data addressed by Rr	$(A_{0-3})\leftrightarrow((R0_{0-3}))$ $(A_{0-3})\leftrightarrow((R1_{0-3}))$		
	MOV A, PSW	C7	1/1	move PSW contents to accumulator	(A)←(PSW)		
	MOV PSW, A	D7	1/1	move accumulator contents to PSW	(PSW)←(A)		3
	MOVP A, @A	A3	1/2	move indirectly addressed data in current page to A	(A)←((A))		
	MOVP3 A,@A	E3	1/2	move data in page 3 to A	(A)←((A))	in page 3	

	MOVX A,@Rr	80 81	1/2	move indirect the contents of external memory to A	(A)←((R))	r = 0–1	
	MOVX @Rr,A	90 91	1/2	move indirect the contents of A to external memory	((R))←(A)	r = 0–1	
FLAGS	CLR C	97	1/1	clear carry bit	(C)←0		2
	CPL C	A7	1/1	complement carry bit	(C)←NOT(C)		2
REGISTER	INC Rr	1*	1/1	increment register by 1	(Rr)←(Rr) + 1	r = 0–7	
	INC @Rr	10 11	1/1	increment RAM data, addressed by Rr, by 1	((R0))←((R0)) + 1 ((R1))←((R1)) + 1		
	DEC Rr	C*	1/1	decrement register by 1	(Rr)←(Rr) –1	r = 0–7	
BRANCH	JMP addr	● 4 address	2/2	unconditional jump within a 2 K bank	(PC_{8-10})←$addr_{8-10}$ (PC_{0-7})←$addr_{0-7}$ (PC_{11-12})←MBFF 0–1		
	JMPP @A	B3	1/2	indirect jump within a page	(PC_{0-7})←((A))		
	DJNZ Rr, addr	E* address	2/2	decrement Rr by 1 and jump if not zero to addr	(Rr)←(Rr) –1 if (Rr) not zero (PC_{0-7})←addr	r = 0–7	
	JF0 addr	B6 address	2/2	jump to addr if F0 = 1	if F0 = 1: (PC_{0-7})←addr		
	JF1 addr	76 address	2/2	jump to addr if F1 = 1	if F1 = 1: (PC_{0-7})←addr		
	JNI addr	86 address	2/2	jump to addr if $\overline{INT}$ = 0	if $\overline{INT}$ = 0: (PC_{0-7})←addr		
	JBb addr	▲ 2 address	2/2	jump to addr if Acc. bit b = 1	if b = 1: (PC_{0-7})←addr	b = 0–7	
	JC addr	F6 address	2/2	jump to addr if C = 1	if C = 1: (PC_{0-7})←addr		
	JNC addr	E6 address	2/2	jump to addr if C = 0	if C = 0: (PC_{0-7})←addr		
	JZ addr	C6 address	2/2	jump to addr if A = 0	if A = 0: (PC_{0-7})←addr		
	JNZ addr	96 address	2/2	jump to addr if A is NOT zero	if A ≠0: (PC_{0-7})←addr		
	JT0 addr	36 address	2/2	jump to addr if T0 = 1	if T0 = 1: (PC_{0-7})←addr		
	JNT0 addr	26 address	2/2	jump to addr if T0 = 0	if T0 = 0: (PC_{0-7})←addr		
	JT1 addr	56 address	2/2	jump to addr if T1 = 1	if T1 = 1: (PC_{0-7})←addr		
	JNT1 addr	46 address	2/2	jump to addr if T1 = 0	if T1 = 0: (PC_{0-7})←addr		
	JTF addr	16 address	2/2	jump to addr if Timer Flag = 1	if TF = 1: (PC_{0-7})←addr		4

	mnemonic	opcode (hex.)	bytes/ cycles	description	function	notes
TIMER/EVENT COUNTER	MOV A, T	42	1/1	move timer/event counter contents to accumulator	(A)←(T)	
	MOV T, A	62	1/1	move accumulator contents to timer/event counter	(T)←(A)	
	STRT CNT	45	1/1	start event counter		
	STRT T	55	1/1	start timer		
	STOP TCNT	65	1/1	stop timer/event counter		
	EN TCNTI	25	1/1	enable timer/event counter interrupt		
	DIS TCNTI	35	1/1	disable timer/event counter interrupt		
CONTROL	EN I	05	1/1	enable external interrupt		
	DIS I	15	1/1	disable external interrupt		
	SEL RB0	C5	1/1	select register bank 0	(RBS)←0	5
	SEL RB1	D5	1/1	select register bank 1	(RBS)←1	5
	SEL MB0	E5	1/1	select program memory bank 0	(MBFF0)←0, (MBFF1)←0	
	SEL MB1	F5	1/1	select program memory bank 1	(MBFF0)←1, (MBFF1)←0	
	ENT0 CLK	75	1/1	enable clock output onto T0		
SUBROUTINE	CALL addr	▲4 address	2/2	jump to subroutine	((SP))←(PC), ($PSW_{4, 6, 7}$) (SP)←(SP) + 1 (PC_{8-10})←$addr_{8-10}$ (PC_{0-7})←$addr_{0-7}$ (PC_{11-12})←MBFF 0–1	6
	RET	83	1/2	return from subroutine	(SP)←(SP) –1 (PC)←((SP))	6
	RETR	93	1/2	return from interrupt and restore bits 4, 6, 7 of PSW	(SP)←(SP) –1 ($PSW_{4, 6, 7}$) + (PC)←((SP))	6

INPUT/OUTPUT	OUTL BUS,A	02	1/2	output accumulator to BUS	(BUS)←A		
	IN A,Pp	09 0A	1/2	input port p data to accumulator	(A)←(P1) (A)←(P2)	p = 1–2	7
	INS A,BUS	08	1/2	input strobed BUS data into accumulator	(A)←(BUS)		
	OUTL Pp,A	39 3A	1/2	output accumulator data to port p	(P1)←(A) (P2)←(A)	p = 1–2	
	ANL BUS, # data	98	2/2	logical AND immediate data with BUS	(BUS)←(BUS) AND data		
	ANL Pp, # data	99 9A	2/2	AND port p data with immediate data	(P1)←(P1) AND data (P2)←(P2) AND data	p = 1–2	
	ORL Pp, # data	89 8A	2/2	OR port p data with immediate data	(P1)←(P1) OR data (P2)←(P2) OR data	p = 1–2	
	ORL BUS, # data	88	2/2	logical OR immediate data with BUS	(BUS)←(BUS) OR data		
	MOVD A,Pp	0C 0D 0E 0F	1/2	move contents of designated port (4–7) to A	(A_{0-3})←(Pp) (A_{4-7})←0	p = 4–7	
	MOVD Pp,A	3C 3D 3E 3F	1/1	move contents of A to designated port (4–7)	(Pp)←(A_{0-3})	p = 4–7	
	ANLD Pp,A	9C 9D 9E 9F	1/2	logical AND contents of A with designated port (4–7)	(Pp)←(Pp) AND (A_{0-3})	p = 4–7	
	ORLD Pp,A	8C 8D 8E 8F	1/1	logical OR contents of A with designated port (4–7)	(Pp)←(Pp) OR (A_{0-3})	p = 4–7	
	NOP	00	1/1	no operation			

Notes to Table 2.

1. PSW CY, AC affected
2. PSW CY affected
3. PSW PS affected
4. Execution of JTF and JNTF instruction resets the Timer Flag (TF).
5. PSW RBS affected
6. PSW SP_0, SP_1, SP_2 affected
7. (A) = 111 P23, P22, P21, P20.
8. (S1) has a different meaning for read and write operation, see serial I/O interface.

* : 8, 9, A, B, C, D, E, F
● : 0, 2, 4, 6, 8, A, C, E
▲ : 1, 3, 5, 7, 9, B, D, F

Table 3 Instruction timing (see also Figs 11 and 12)

	cycle 1					cycle 2				
instruction	S1	S2	S3	S4	S5	S1	S2	S3	S4	S5
IN A,P	fetch instruction	increment program counter	–	increment timer	–	–	read port	* –	–	–
OUTL P,A	¦	¦	–	¦	output to port	–	–	* –	–	–
ANL P,#data	¦	* ¦	–	¦	read port	fetch immediate data	–	*increment program counter	output to port	–
ORL P,#data	¦	* ¦	–	¦	read port	fetch immediate data	–	*increment program counter	output to port	–
INS A,BUS	¦	¦	–	¦	–	–	read port	* –	–	–
OUTL BUS,A	¦	¦	–	¦	output to port	–	–	* –	–	–
ANL BUS,#data	¦	* ¦	–	¦	read port	fetch immediate data	–	*increment program counter	output to port	–
ORL BUS,#data	¦	* ¦	–	¦	read port	fetch immediate data	–	*increment program counter	output to port	–
MOVX @R,A	¦	¦	output RAM address	¦	output data to RAM	–	–	* –	–	–
MOVX A,@R	¦	¦	output RAM address	¦	–	–	read data	* –	–	–
MOVD A,P	fetch instruction	increment program counter	output opcode/ address	increment timer	–	–	read P2 lower	* –	–	–

	cycle 1					cycle 2				
instruction	S1	S2	S3	S4	S5	S1	S2	S3	S4	S5
MOVD P,A	fetch instruction	increment program counter	output opcode/ address	increment timer	output data to P2 lower	–	–	* –	–	–
ANLD P,A	¦		output opcode/ address	¦	output data	–	–	* –	–	–
ORLD P,A	¦		output opcode/ address	¦	output data	–	–	* –	–	–
J (conditional)	¦	*	sample condition	increment timer	–	fetch immediate data	–	* opdata program counter	–	–
STRT CNT STRT T	¦	*	–	–	start counter					
STOP TCNT	¦	*	–	–	stop counter					
EN I	¦	*	–	enable interrupt	–					
DIS I	¦	*	–	disable interrupt	–					
ENT0 CLK	fetch instruction	* increment program counter	–	enable clock	–					

* Valid instruction addresses are output at this time if external program memory is being accessed.

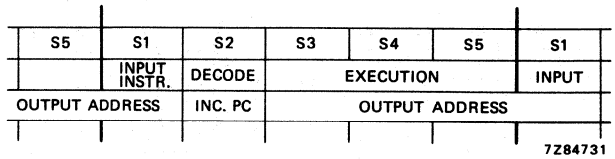


Fig. 11 Instruction cycle.

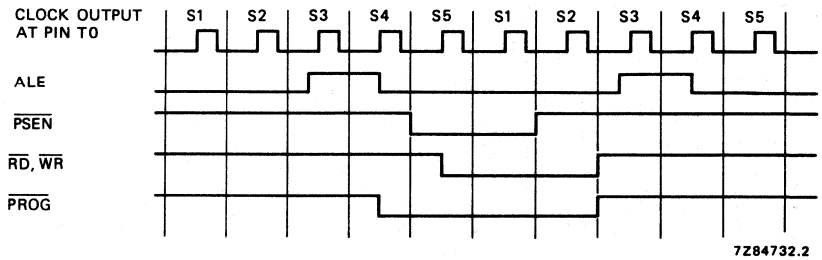


Fig. 12 Instruction cycle timing.

Table 4 Instruction map.

first hex ↓ / second hex →	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	NOP		OUTL BUS,A	ADD A, #data	JMP page 0	EN I		DEC A	INS A,BUS	IN A, Pp	2		MOVD A,Pp			
1	INC @Rr		JB0 addr	ADDC A, #data	CALL page 0	DIS I	JTF addr	INC A	INC Rr							
2	XCH A, @Rr			MOV A, #data	JMP page 1	EN TCNTI	JNT0 addr	CLR A	XCH A,Rr							
3	XCHD A, @Rr		JB1 addr		CALL page 1	DIS TCNTI	JT0 addr	CPL A		OUTL Pp,A			MOVD Pp,A			
4	ORL A, @Rr		MOV A, T	ORL A, #data	JMP page 2	STRT CNT	JNT1 addr	SWAP A	ORL A,Rr							
5	ANL A, @Rr		JB2 addr	ANL A, #data	CALL page 2	STRT T	JT1 addr	DA, A	ANL A,Rr							
6	ADD A, @Rr		MOV T,A		JMP page 3	STOP TCNT		RRC A	ADD A,Rr							
7	ADDC A, @Rr		JB3 addr		CALL page 3	ENT0 CLK	JF1 addr	RR A	ADDC A,Rr							
8	MOVX A, @Rr			RET	JMP page 4	CLR F0	JNI addr		ORL BUS, #data	ORL Pp, #data	2		ORLD Pp,A			
9	MOVX @Rr,A		JB4 addr	RETR	CALL page 4	CPL F0	JNZ addr	CLR C	ANL BUS, #data	ANP Pp, #data	2		ANLD Pp,A			
A	MOV @Rr, A			MOVP A, @A	JMP page 5	CLR F1		CPL C	MOV Rr,A							
B	MOV @Rr, #data		JB5 addr	JMPP @A	CALL page 5	CPL F1	JF0 addr		MOV R, #data							
C					JMP page 6	SEL RB0	JZ addr	MOV A, PSW	DEC Rr							
D	XRL A, @Rr		JB6 addr	XRL A, #data	CALL page 6	SEL RB1		MOV PSW, A	XRL A,Rr							
E				MOVP3 A @A	JMP page 7	SEL MB0	JNC addr	RL A	DJNZ Rr, addr							
F	MOV A, @Rr		JB7 addr		CALL page 7	SEL MB1	JC addr	RLC A	MOV A,Rr							

RATINGS

Limiting values in accordance with the Absolute Maximum System (IEC 134)

Input voltage with respect to V_{SS}			
except input EA	V_I	−0,5 to + 7	V
input EA	V_I	−0,5 to + 12	V
DC current into any input or output	$\pm I_I, \pm I_O$	max. 10	mA
Total power dissipation	P_{tot}	max. 1	W
Storage temperature range	T_{stg}	−65 to +150	°C
Operating ambient temperature range	T_{amb}	see Table 5	

Table 5 MAB80XXH versions.

version	internal memory		RAM st/by.	frequency (MHz)		temperature range (°C)
				min.	max.	
MAB8048H	1 K x 8 ROM	64 byte RAM	yes	1,0	11,0	0 to +70
MAB8035HL	none	64 byte RAM	yes	1,0	11,0	0 to +70
MAF8048H	1 K x 8 ROM	64 byte RAM	yes	1,0	11,0	−40 to +85
MAF8035HL	none	64 byte RAM	yes	1,0	11,0	−40 to +85
MAF80A48H	1 K x 8 ROM	64 byte RAM	yes	1,0	10,0	−40 to +110
MAF80A35HL	none	64 byte RAM	yes	1,0	10,0	−40 to +110
MAB8049H	2 K x 8 ROM	128 byte RAM	yes	1,0	11,0	0 to +70
MAB8039HL	none	128 byte RAM	yes	1,0	11,0	0 to +70
MAF8049H	2 K x 8 ROM	128 byte RAM	yes	1,0	11,0	−40 to +85
MAF8039HL	none	128 byte RAM	yes	1,0	11,0	−40 to +85
MAF80A49H	2 K x 8 ROM	128 byte RAM	yes	1,0	10,0	−40 to +110
MAF80A39HL	none	128 byte RAM	yes	1,0	10,0	−40 to +110
MAB8050H	4 K x 8 ROM	256 byte RAM	yes	1,0	11,0	0 to +70
MAB8040HL	none	256 byte RAM	yes	1,0	11,0	0 to +70
MAF8050H	4 K x 8 ROM	256 byte RAM	yes	1,0	11,0	−40 to +85
MAF8040HL	none	256 byte RAM	yes	1,0	11,0	−40 to +85
MAF80A50H	4 K x 8 ROM	256 byte RAM	yes	1,0	10,0	−40 to +110
MAF80A40HL	none	256 byte RAM	yes	1,0	10,0	−40 to +110

DC CHARACTERISTICS (MAB8048H/35HL; MAB8049H/39HL; MAB8050H/40HL)

$V_{CC} = V_{DD} = 5$ V (± 10%); $V_{SS} = 0$ V; $T_{amb} = 0$ to + 70 °C; all voltages with respect to V_{SS} unless otherwise specified

parameter	symbol	min.	typ.	max.	unit
Supply current at $V_{DD} = 5$ V ± 10%; $V_{SS} = V_{CC} = 0$ V					
MAB8048H/35HL	I_{DD}	–	–	6	mA
MAB8049H/39HL	I_{DD}	–	–	8	mA
MAB8050H/40HL	I_{DD}	–	–	15	mA
Supply current (total) at $V_{DD} = V_{CC} = 5$ V ± 10%; $V_{SS} = 0$ V					
MAB8048H/35HL	$I_{DD} + I_{CC}$	–	–	80	mA
MAB8049H/39HL	$I_{DD} + I_{CC}$	–	–	90	mA
MAB8050H/40HL	$I_{DD} + I_{CC}$	–	–	100	mA
Inputs					
Input voltage LOW all inputs except XTAL 1, XTAL 2, $\overline{\text{RESET}}$	V_{IL}	−0,5	–	0,8	V
Input voltage LOW XTAL 1, XTAL 2, $\overline{\text{RESET}}$	V_{IL1}	−0,5	–	0,6	V
Input voltage HIGH all inputs except XTAL 1, XTAL 2, $\overline{\text{RESET}}$	V_{IH}	2,0	–	V_{CC}	V
Input voltage HIGH XTAL 1, XTAL 2, $\overline{\text{RESET}}$	V_{IH1}	3,8	–	V_{CC}	V
Outputs					
Output voltage LOW (DB0 to DB7) at $I_{OL} = 2$ mA	V_{OL}	–	–	0,45	V
Output voltage LOW ($\overline{\text{RD}}$, $\overline{\text{WR}}$, $\overline{\text{PSEN}}$, ALE) at $I_{OL1} = 1{,}8$ mA	V_{OL1}	–	–	0,45	V
Output voltage LOW ($\overline{\text{PROG}}$) at $I_{OL2} = 1$ mA	V_{OL2}	–	–	0,45	V
Output voltage LOW (all other outputs) at $I_{OL3} = 1{,}6$ mA	V_{OL3}	–	–	0,45	V
Output voltage HIGH (DB0 to DB7) at $-I_{OH} = 400\ \mu$A	V_{OH}	2,4	–	–	V
Output voltage HIGH ($\overline{\text{RD}}$, $\overline{\text{WR}}$, $\overline{\text{PSEN}}$, ALE) at $-I_{OH1} = 100\ \mu$A	V_{OH1}	2,4	–	–	V
Output voltage HIGH (all other outputs) at $-I_{OH} = 40\ \mu$A	V_{OH2}	2,4	–	–	V

DC CHARACTERISTICS (continued)

parameter	symbol	min.	typ.	max.	unit
Input leakage current (T1, $\overline{INT}$) at $V_{SS} < V_I < V_{CC}$	$\pm I_{IL}$	–	–	10	μA
Output leakage current (DB0 to DB7, T0; high impedance) at $V_{SS} + 0{,}45\ V < V_I < V_{CC}$	$\pm I_{OZ}$	–	–	10	μA
Input load current (P1.0 to P1.7, P2.0 to P2.7, EA, $\overline{SS}$) at $V_{SS} + 0{,}45\ V < V_I < V_{CC}$	$-I_{LI}$	–	–	500	μA
Input load current ($\overline{RESET}$) at $V_{SS} < V_I < V_{CC}$	$-I_{LI1}$	20	–	300	μA

DC CHARACTERISTICS

MAF8048H/35HL; MAF8049H/39HL; MAF8050H/40HL (at $T_{amb} = -40$ to + 85 °C)
MAF80A48H/A35HL; MAF80A49H/A39HL; MAF80A50H/A40HL (at $T_{amb} = -40$ to + 110 °C)

$V_{CC} = V_{DD} = 5\ V$ (± 10%); $V_{SS} = 0\ V$; T_{amb} as above; all voltages with respect to V_{SS} unless otherwise specified

parameter	symbol	min.	typ.	max.	unit
Supply current at $V_{DD} = 5\ V \pm 10\%$; $V_{SS} = V_{CC} = 0\ V$					
MAF8048H/35HL; MAF80A48H/A35HL	I_{DD}	–	–	8	mA
MAF8049H/39HL; MAF80A49H/A39HL	I_{DD}	–	–	10	mA
MAF8050H/40HL; MAF80A50H/A40HL	I_{DD}	–	–	18	mA
Supply current (total) at $V_{DD} = V_{CC} = 5\ V \pm 10\%$; $V_{SS} = 0\ V$					
MAF8048H/35HL; MAF80A48H/A35HL	$I_{DD} + I_{CC}$	–	–	90	mA
MAF8049H/39HL; MAF80A49H/A39HL	$I_{DD} + I_{CC}$	–	–	100	mA
MAF8050H/40HL; MAF80A50H/A40HL	$I_{DD} + I_{CC}$	–	–	120	mA
Inputs					
Input voltage HIGH all inputs except XTAL 1, XTAL 2, $\overline{RESET}$	V_{IH}	2,2	–	V_{CC}	V
Input load current (P1.0 to P1.7, P2.0 to P2.7, EA, $\overline{SS}$) at $V_{SS} + 0{,}45\ V < V_I < V_{CC}$	$-I_{LI}$	–	–	0,6	mA

AC CHARACTERISTICS (MAB8048H/35HL; MAB8049H/39HL; MAB8050H/40HL)

$V_{CC} = V_{DD} = 5$ V (± 10%); $V_{SS} = 0$ V; $T_{amb} = 0$ to + 70 °C; note 1.
See waveforms Figs 14, 15, 16, 17 and 18

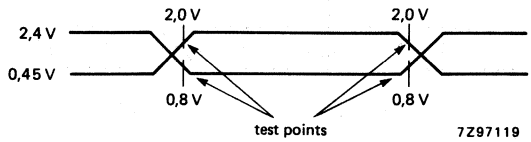
parameter	symbol	$f(t_{CL})$ (note 2)	11 MHz		unit
			min.	max.	
Clock period (note 2)	t_{CL}	$1/(f_{XTAL})$	90,9	1000	ns
ALE pulse duration	t_{LL}	$3{,}5t_{CL}-170$	150	–	ns
Address set-up time to ALE (note 3)	t_{AL}	$2t_{CL}-110$	70	–	ns
Address hold time after ALE	t_{LA}	$t_{CL}-40$	50	–	ns
Control pulse duration $\overline{RD}$, $\overline{WR}$	t_{CC1}	$7{,}5t_{CL}-200$	480	–	ns
Control pulse duration $\overline{PSEN}$	t_{CC2}	$6t_{CL}-200$	350	–	ns
Data set-up time before $\overline{WR}$	t_{DW}	$6{,}5t_{CL}-200$	390	–	ns
Data set-up time after $\overline{WR}$	t_{WD}	$t_{CL}-50$	40	–	ns
Data hold time					
$\overline{RD}$, $\overline{PSEN}$	t_{DR}	$1{,}5t_{CL}-30$	0	110	ns
$\overline{RD}$ to data input	t_{RD1}	$6t_{CL}-170$	–	350	ns
$\overline{PSEN}$ to data input	t_{RD2}	$4{,}5t_{CL}-170$	–	190	ns
Address set-up time to $\overline{WR}$	t_{AW}	$5t_{CL}-150$	300	–	ns
Address set-up time to data input ($\overline{RD}$)	t_{AD1}	$10{,}5t_{CL}-220$	–	730	ns
Address set-up time to data input ($\overline{PSEN}$)	t_{AD2}	$7{,}5t_{CL}-200$	–	460	ns
Address floating to $\overline{RD}$, $\overline{WR}$	t_{AFC1}	$2t_{CL}-40$	140	–	ns
Address floating to $\overline{PSEN}$ (note 3)	t_{AFC2}	$0{,}5t_{CL}-40$	10	–	ns
ALE to control pulse $\overline{RD}$, $\overline{WR}$	t_{LAFC1}	$3t_{CL}-75$	200	–	ns
ALE to control pulse $\overline{PSEN}$	t_{LAFC2}	$1{,}5t_{CL}-75$	60	–	ns
Control pulse to ALE $\overline{RD}$, $\overline{WR}$, $\overline{PROG}$	t_{CA1}	$t_{CL}-40$	50	–	ns
Control pulse to ALE $\overline{PSEN}$	t_{CA2}	$4t_{CL}-40$	320	–	ns
Port control set-up to $\overline{PROG}$	t_{CP}	$1{,}5t_{CL}-80$	50	–	ns
Port control hold to $\overline{PROG}$	t_{PC}	$4t_{CL}-260$	100	–	ns
$\overline{PROG}$ to Port 2 input must be valid	t_{PR}	$8{,}5t_{CL}-120$	–	650	ns
Input data hold time from $\overline{PROG}$	t_{PF}	$1{,}5t_{CL}$	0	150	ns
Output data set-up time	t_{DP}	$6t_{CL}-290$	250	–	ns
Output data hold time	t_{PD}	$1{,}5t_{CL}-90$	40	–	ns
$\overline{PROG}$ pulse duration	t_{PP}	$10{,}5t_{CL}-250$	700	–	ns
Port 2 I/O data set-up time to ALE	t_{PL}	$4t_{CL}-200$	160	–	ns
Port 2 I/O data hold time to ALE	t_{LP}	$1{,}5t_{CL}-120$	15	–	ns

AC CHARACTERISTICS (continued)

parameter	symbol	$f(t_{CL})$ (note 2)	11 MHz		unit
			min.	max.	
Port output from ALE	t_{PV}	$4{,}5t_{CL}+100$	–	510	ns
T0 repetition rate	t_{0PRR}	$3t_{CL}$	270	–	ns
Cycle time	t_{CY}	$15/f_{XTAL}$	1,36	15	μs
MAF8048H/35HL; MAF8049H/39HL; MAF8050H/40HL					
Clock period (note 2)	t_{CL}	$1/(f_{XTAL})$	90,8	1000	ns

Notes to AC characteristics

1. Control outputs: C_L = 80 pF.
   Bus outputs: C_L = 150 pF.
2. $f(t_{CL})$ assumes 50% duty cycle on XTAL 1 and XTAL 2; minimum frequency = 1 MHz.
3. Bus high-impedance load: 20 pF.



A.C. testing inputs are driven at 2,4 V for a logic 1 and 0,45 V for a logic 0. Output timing measurements are taken 2,0 V for a logic 1 and 0,8 V for a logic 0.

Fig. 13 A.C. testing input, output waveform.

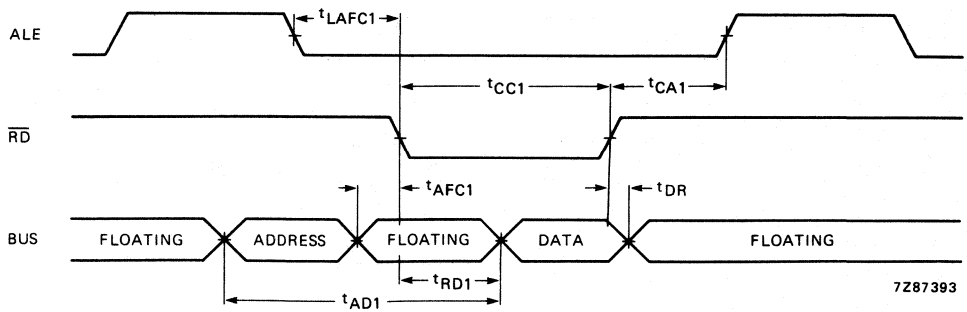


Fig. 14 Read from external data memory.

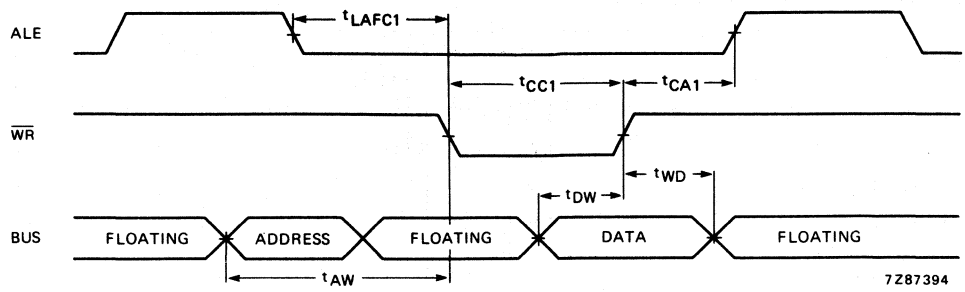


Fig. 15 Write to external memory.

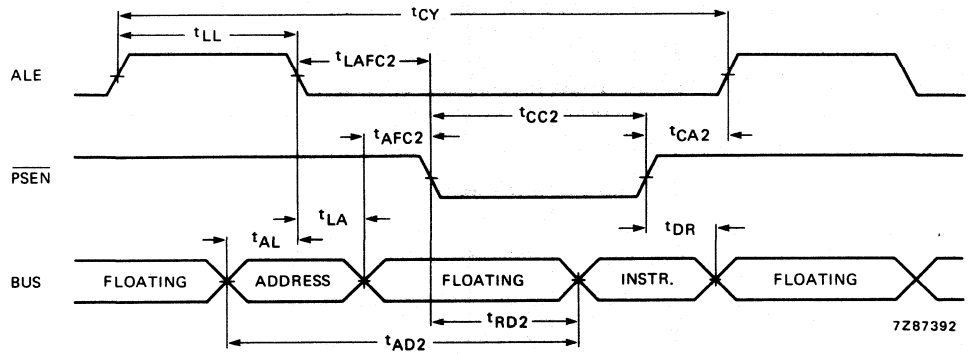


Fig. 16 Instruction fetch from program memory.

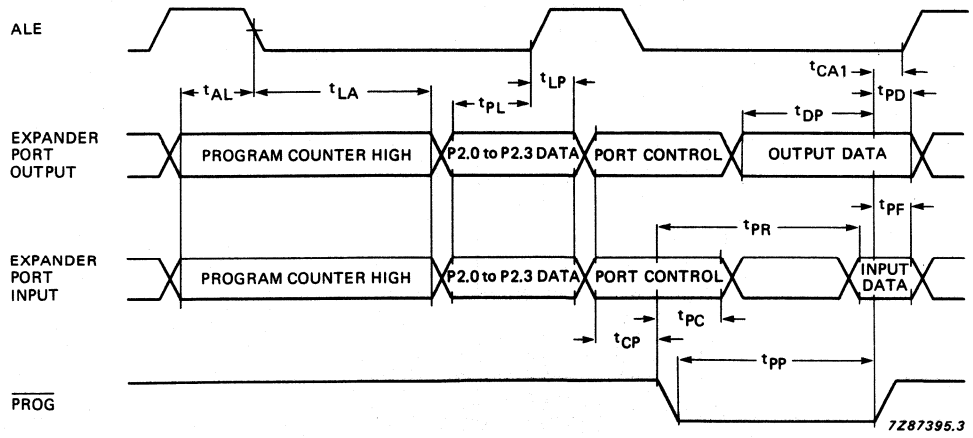


Fig. 17 Port 2 timing.

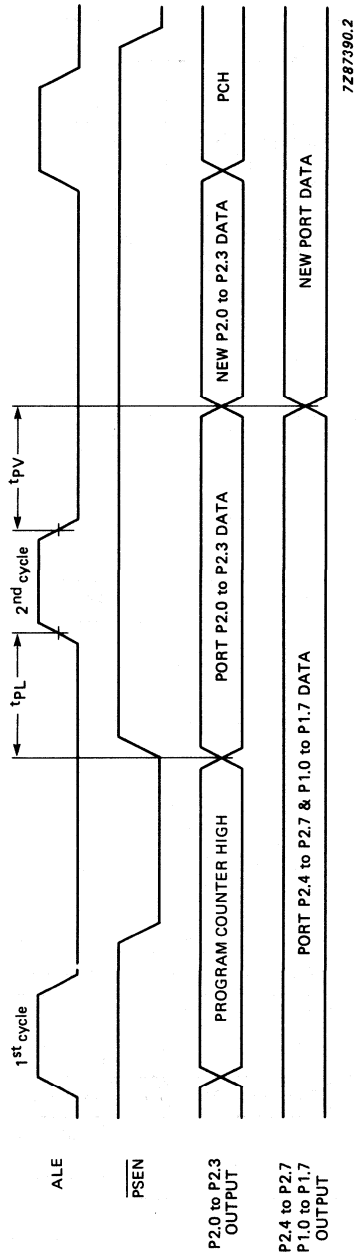


Fig. 18 I/O port timing.

SINGLE-CHIP 8-BIT MICROCONTROLLER

DESCRIPTION

The MAB84X1 family of microcontrollers is fabricated in NMOS. The family consists of 5 devices:

- MAB8401 — 128 bytes RAM, external program memory, with 8-bit LED-driver (10mA), emulation of MAB/F8422/42* possible
- MAB/MAF8411 — 1K byte ROM/64 bytes RAM plus 8-bit LED-driver
- MAB/MAF8421 — 2K bytes ROM/64 bytes RAM plus 8-bit LED-driver
- MAB/MAF8441 — 4K bytes ROM/128 bytes RAM plus 8-bit LED-driver
- MAB/MAF8461 — 6K bytes ROM/128 bytes RAM plus 8-bit LED-driver

Each version has 20 quasi-bidirectional I/O port lines, one serial I/O line, one single-level vectored interrupt, an 8-bit timer/event counter and on-board clock oscillator and clock circuits. Two 20-pin versions, MAB/F8422 and MAB/F8442* are also available.

This microcontroller family is designed to be an efficient controller as well as an arithmetic processor. The instruction set is based on that of the MAB8048. The microcontrollers have extensive bit handling abilities and facilities for both binary and BCD arithmetic.

For detailed information see the "8-bit Single-chip Microcontrollers user manual".

* See data sheet on MAB/F8422/42.

Features

- 8-bit: CPU, ROM, RAM and I/O in a single 28-lead DIL package
- 1K, 2K, 4K or 6K ROM bytes plus a ROM-less version
- 64 or 128 RAM bytes
- 20 quasi-bidirectional I/O port lines
- Two testable inputs: one of which can be used to detect zero cross-over, the other is also the external interrupt input
- Single level vectored interrupts: external, timer/event counter, serial I/O
- Serial I/O that can be used in single or multi-master systems (serial I/O data via an existing port line and clock via a dedicated line)
- 8-bit programmable timer/event counter
- Internal oscillator, generated with inductor, crystal, ceramic resonator or external source
- Over 80 instructions (based on MAB8048) all of 1 or 2 cycles
- Single 5 V power supply (± 10%)
- Operating temperature ranges:
  - 0 to + 70 °C — MAB84X1 family
  - −40 to + 85 °C — MAF84X1 family only
  - −40 to + 110 °C — MAF84AX1 family only

PACKAGE OUTLINES

MAB8401B: 28-lead 'Piggy-back' package (with up to 28-pin EPROM on top).
MAB8401WP: 68-lead plastic leaded chip-carrier (PLCC) (SOT-188).
MAB/MAF8411/21/41/61P: 28-lead DIL; plastic (SOT-117).
MAF84A11/21/41/61P: 28-lead DIL; plastic (SOT-117).
MAB8411/21/41/61T: 28-lead mini-pack; plastic (SO-28; SOT-136A).

PINNING

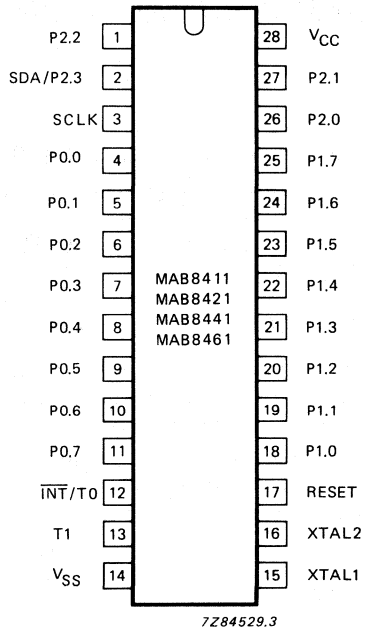
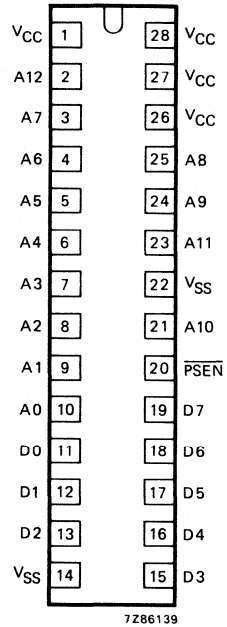


Fig. 1 Pinning diagram for mask-programmable devices MAB8411, MAB8421, MAB8441, MAB8461 and for MAB8401 'Piggy-back' version bottom pinning (for top pinning see Fig. 2).

PINNING DESIGNATION

V_{SS}	14	**Ground**
V_{CC}	28	**Power supply,** + 5 V
P0.0 – P0.7	4 – 11	**Port 0,** 8-bit quasi-bidirectional I/O port
P1.0 – P1.7	18 – 25	**Port 1,** 8-bit quasi-bidirectional I/O port with 8-bit LED driver
P2.0 – P2.3	26, 27, 1, 2	**Port 2,** 4-bit quasi-bidirectional I/O port; SDA/P2.3 is the serial data I/O in serial I/O mode
SCLK	3	Bidirectional clock for serial I/O
$\overline{INT}$/T0	12	External interrupt input (sensitive to a negative-going edge min LOW > 7 clock pulses, min HIGH > 4 clock pulses), testable using the JT0 or JNT0 instructions.
T1	13	Input pin, testable using the JT1 or JNT1 instructions. It can be designated as event counter input using the STRT CNT instruction. It can also be used to detect zero cross-over of slowly moving a.c. inputs.
RESET	17	Input to initialize the processor (active HIGH).
XTAL1	15	Connection to timing component (crystal) that determines the frequency of the internal oscillator. It is also the input for an external clock source.
XTAL2	16	Connection to other side of the timing component.

MAB8401B (top pinning)



PIN DESIGNATION

designation	pin	function
V_{SS}	14, 22	**Ground**
V_{CC}	1, 26–28	**Power supply,** + 5 V
A0–A12	10–3, 25, 24, 21, 23, 2	Address outputs
D0–D7	11–13, 15–19	Data inputs
$\overline{PSEN}$	20	Program store enable

Fig. 2 Pinning diagram for MAB8401B 'Piggy-back' version top pinning (for bottom pinning see Fig. 1); to access a 2732 or 2764 EPROM.

Note

Access times for ROMS/EPROMS to be below 1 μs.

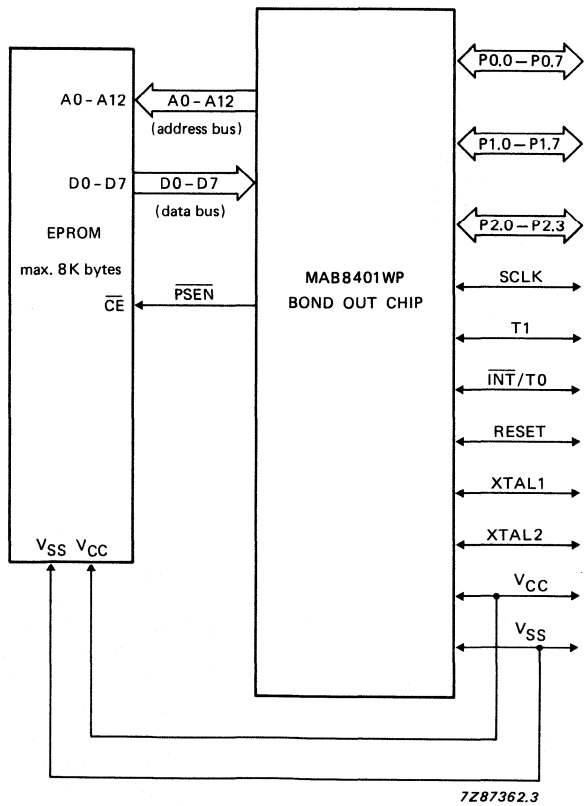


Fig. 2a Connection of EPROM to 'Piggy-back' package MAB8401B.

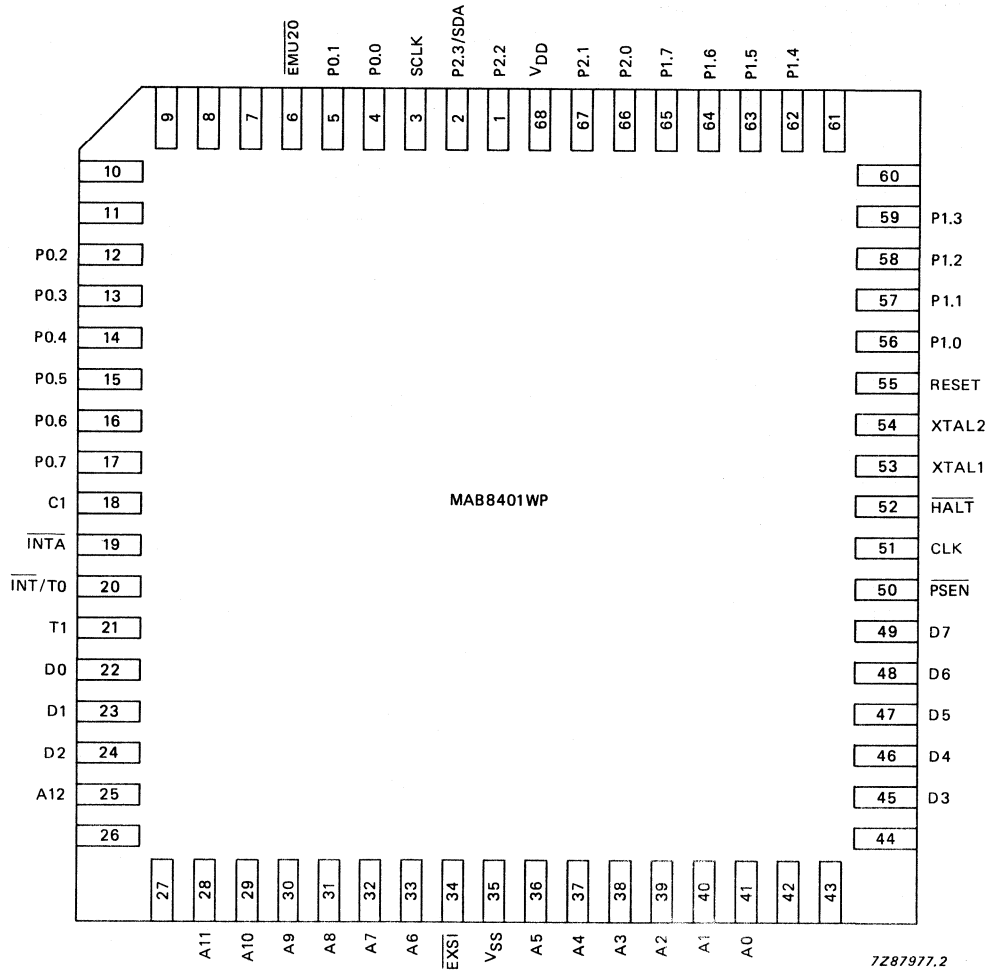


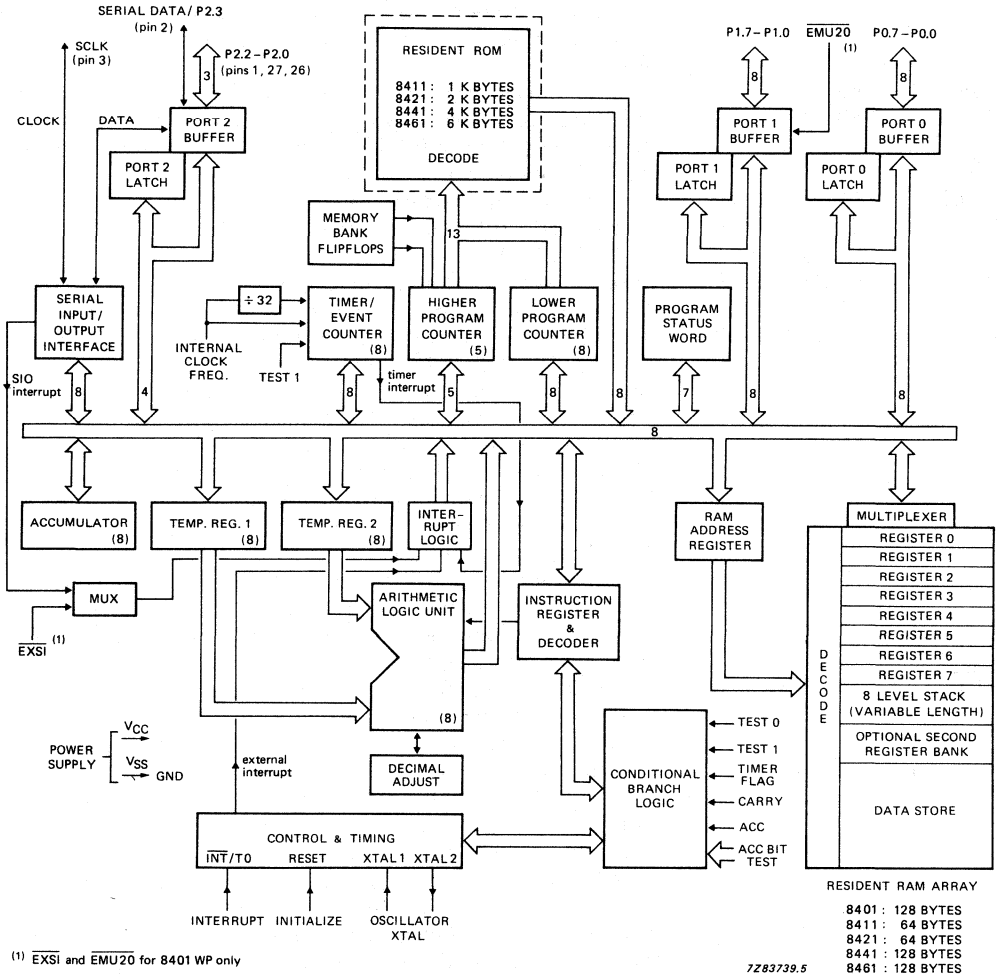
Fig. 3 Pinning diagram; PLCC.

CHIP CARRIER DESIGNATION

designation	pad no.	function
V_{SS}	35	**Ground**
V_{CC}	68	**Power supply,** + 5 V
P0.0 – P0.7	4–5, 12–17	**Port 0,** 8-bit quasi-bidirectional I/O port
P1.0 – P1.7	56–59, 62–65	**Port 1,** 8-bit quasi-bidirectional I/O port with 8-bit LED driver
P2.0 – P2.3	66, 67, 1, 2	**Port 2,** 4-bit quasi-bidirectional I/O port; SDA/P2.3 is the serial data I/O in serial I/O mode
SCLK	3	Bidirectional clock for serial I/O
INT/T0	20	External interrupt input (sensitive to a negative-going edge), testable using the JT0 or JNT0 instructions

T1	21	Input pin, testable using the JT1 or JNT1 instructions. It can be designated as event counter input using the STRT CNT inctruction. It can also be used to detect zero cross-over of slowly moving a.c. inputs.
RESET	55	Input to initialize the processor (active HIGH)
XTAL1	53	Connection to timing component (e.g. crystal) that determines the frequency of the internal oscillator. It is also the input for an external clock source.
XTAL2	54	Connection to other side of the timing component
$\overline{EXSI}$	34	External serial I/O interrupt (active-LOW) for emulation of MAB/F8422/42.
A0–A12	41–36, 33–28 25	Program memory address outputs (active HIGH); A0 = LSB, A12 = MSB. Address output change after begin $\phi 3$ of TS8.
D0–D7	22–24, 45–49	Data input lines (active HIGH) used for reading external program memory. D0 = LSB, D7 = MSB.
CLK	51	Clock output buffered from XTAL2. On the positive-going edge the (internal) ϕ clock goes HIGH.
$\overline{PSEN}$	50	Program store enable. This signal is used for enabling the external EPROM (e.g. on the 'Piggy-back' version). For emulation, it enables the emulation memory and it indicates machine cycles. Active LOW during TS9,*TS10 of each machine cycle and TS1 of the following machine cycle.
$\overline{C1}$	18	Cycle 1 indication output (active LOW). During emulation, this signal indicates the opcode fetch cycle (useful for external instruction decoding, real-time trace). Active from start of TS10 of the cycle preceding cycle 1, until the start of TS10 of cycle 1.
$\overline{HALT}$	52	Halt input (active LOW). If activated, the current instruction is finished and the microcontroller stops execution (HALT mode). The next program counter address is available on the address bus. Program counter and timer/event counter are no longer updated. The serial I/O finishes the current transmit/receive action and goes into the idle state. Interrupts are *not* sampled in the HALT mode, they are only sampled when the microcontroller is running. Interrupt routines can be single-stepped as a normal program.
$\overline{INTA}$	19	Interrupt acknowledge output (active LOW). It indicates any interrupt acception. Active from start of TS8 of the interrupted cycle, until start of TS7 of the second cycle of the (internally forced 'CALL vector address' instruction. During $\overline{INTA}$ active, the address bus shows the address that has been saved in the stack (return address); the C1 output indicates opcode fetch cycles as if a user CALL was executed.
$\overline{EMU20}$	6	Emulate 20-pin version MAB/F8422/42 (active-LOW).

*TS = Time slot, where 10 TS = 1 cycle



(1) EXSI and EMU20 for 8401 WP only

Fig. 4a Block diagram of the MAB84X1 family.

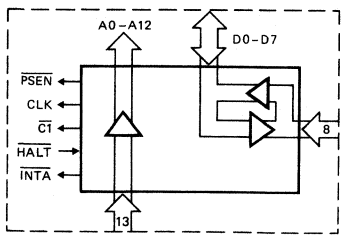


Fig. 4b Replacement for dotted part in Fig. 4a for the MAB8401WP bond-out version.

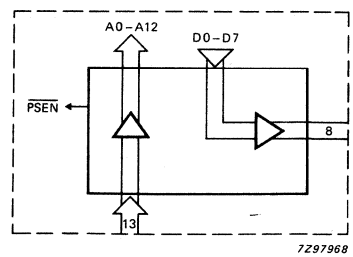


Fig. 4c Replacement of dotted part in Fig. 4a for the MAB8401B 'Piggy-back' version.

FUNCTIONAL DESCRIPTION (for more detail see Microcontroller Users Manual)

Bond-out version MAB8401WP

The bond-out version is a microcontroller that contains no on-board ROM, but has all address and data lines brought out to access an external ROM or EPROM. Thus, this version has more pins than the standard microcontrollers with on-board ROM. It has all the features of the other members of the MAB84X1 family, including emulation facilities for the MAB/F8422/42 (20-pin version). It can address 8K bytes of external ROM. The RAM has 128 bytes.

Piggy-back version MAB8401B

The Piggy-back version is a special package that has standard pinning to the bottom which facilitates insertion as a mask-programmed device. An EPROM is mounted on top in an additional socket. Thus, the total package height is greater than the standard DIL package. Emulation of the 8422/42 is not possible.

Program and data memory

The program memory (ROM) is mask-programmed at our factory. Because the MAB84X1 family offers a range of ROM capacities to suit the application, ROM expansion is not required. Figure 5 shows the program memory map. Program memory is arranged in banks of 2K bytes, that are selected by SEL MB instructions.

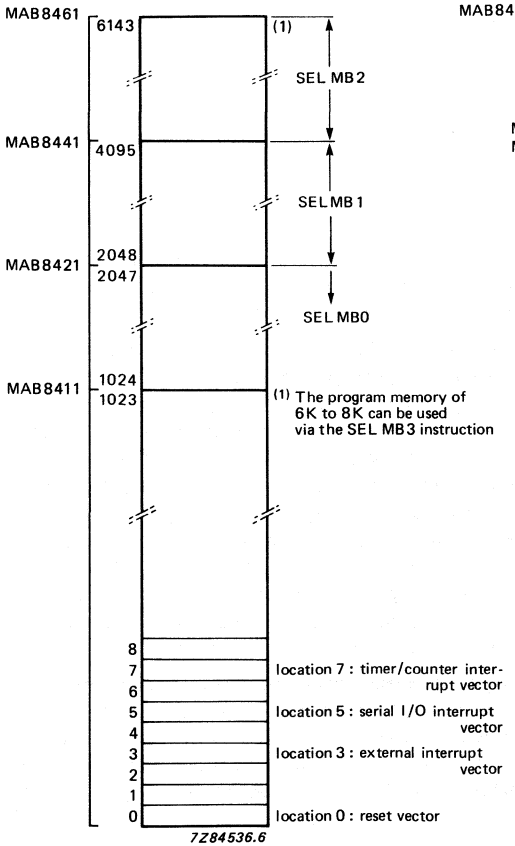


Fig. 5 The program memory map.

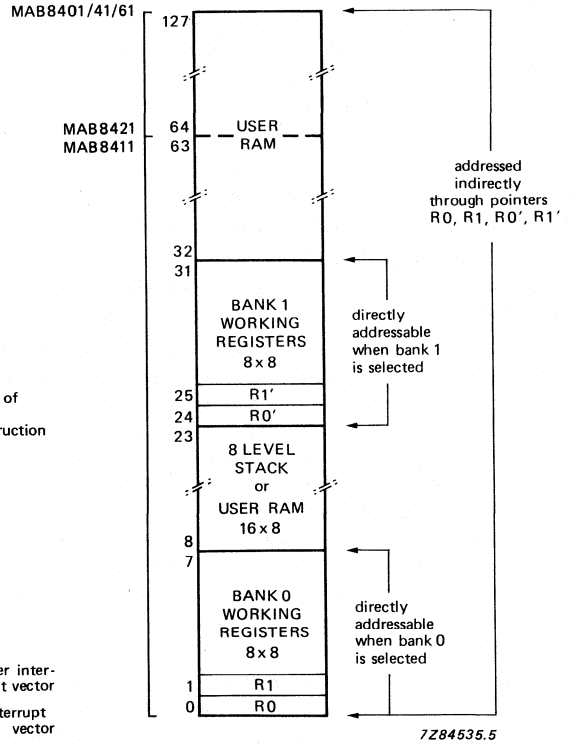


Fig. 6 The data memory map.

FUNCTIONAL DESCRIPTION (continued)

The data memory (RAM) consists of 64 or 128 bytes (8-bit words). All locations are indirectly addressable using RAM pointer registers and up to 16 designated location can be addressed directly. The memory also includes an 8-level program counter stack addressed by a 3-bit stack pointer. Figure 6 shows the data memory map.

On-chip peripheral functions

In addition to the CPU and memories, an interrupt system, I/O facilities, and an 8-bit timer/event counter are integrated on-chip to assist the CPU in repetitions, complicated or time-critical tasks. The I/O facilities include the I/O pins, parallel ports and a serial I/O port, consisting of a data line SDA shared with a parallel port line (P2.3), and a dedicated clock line SCLK.

I/O facilities

The MAB84XX family has 23 I/O lines arranged as:

- Two parallel ports of 8 lines (P0.0–P0.7, P1.0–P1.7). Each line of Port 1 can sink 10 mA.
- A parallel port of 4 lines (P2.0–P2.3).
- A serial I/O consisting of a data line shared with a parallel port line (P2.3) and a separate clock line SCLK;
- An external interrupt and test input $\overline{\text{INT}}$/T0, which when used as a test input can be tested by the conditional jump instructions JT0 or JNT0;
- A test input T1, which can alter program sequences when tested by conditional jump instructions JT1 or JNT1. T1 can also be used as an input to the timer/event counter or to detect zero cross-over of slowly moving AC signals.

All parallel port lines are available in three optional output configurations (except P2.3 – option 1 only):

- Option 1; open drain output without pull-up transistor (Fig. 7(a))
- Option 2; open drain output with pull-up transistor (Fig. 7(b))
- Option 3; push-pull output with pull-up transistor (Fig. 7(c))

If the inputs and outputs on a port are mixed (mixed-mode), the inputs should be options 1 or 2 but not option 3. This prevents cross-currents via TR2 and an external connection to ground, while switching the output on the same port and in parallel, masking the inputs with logic 1 s.

The MAB84X1 family serial I/O interface has been designed to eliminate the heavy processing load imposed upon a normal microcontroller performing serial data transfer. Whereas a normal microcontroller must regularly monitor the serial data bus for the presence of data, the serial I/O interface detects, receives and converts the serial data stream into a parallel format without interrupting the execution of the current program. An interrupt is sent to the microcontroller only when a complete byte is received. Then, the microcontroller reads the data byte in one instruction. Likewise, for transmission, the serial I/O interface performs parallel to serial conversion and subsequent serial output of the data and the microcontroller is only interrupted in the execution of its programmed tasks when a complete byte has been transmitted. The design of the serial I/O interface allows any number of MAB84X1 family devices and peripheral circuits with I^2C bus compatibility to be interconnected by the two-line serial bus. This is achieved by allocating a specific 7-bit address to each device and ensuring that a device reacts only to a message preceded by its own address or the 'general call' address.

Address recognition is performed by the interface hardware so that the microcontroller need only be interrupted when a valid address is received. This saves significant processing time and memory space compared to a conventional microcontroller with a software serial interface. When the address facility is not required, for instance in a system with only two microcontrollers, direct data transfer is possible. In multi-master systems, an automatically invoked arbitration procedure prevents two or more devices transmitting simultaneously.

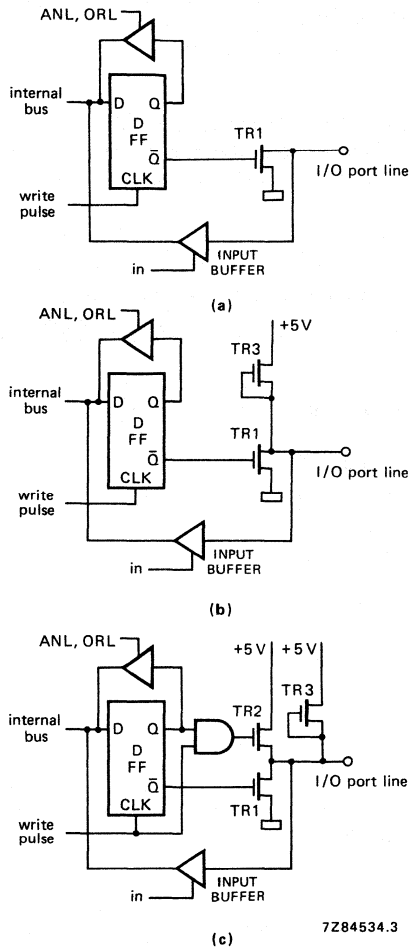


Fig. 7 Quasi-bidirectional I/O interface with (a) open drain output without pull-up transistor, (b) open drain output with pull-up transistor, (c) push-pull output with pull-up transistor.

Serial I/O interface

Figure 8 shows the serial I/O interface. The clock line of the serial bus has exclusive use of pin 3 (SCLK) while the data line shares pin 2 (serial data) with the I/O line P2.3 of port 2. When the serial I/O is enabled, P2.3 is disabled as a parallel port line (P2.3 and SCLK only open drain).

The microcontroller and interface communicate via the internal microcontroller bus and the Serial Interrupt Request line. Data and information controlling the operation of the interface are stored in four registers:

- data shift register S0,
- serial I/O interface status word S1,
- serial clock control word S2,
- address register S0'

FUNCTIONAL DESCRIPTION (continued)

Serial I/O interface (continued)

Data shift register S0

S0 is the shift register that converts serial data to parallel format and vice versa. A pending interrupt is generated only after a complete byte has been transmitted, or after a complete data byte, specific or general call address has been received. The most significant bit is transmitted first.

Serial I/O interface status word S1

S1 provides information about the state of the interface and stores interface control information from the microcontroller. The four most significant bits are common to both read and write instructions, with a separate 4 read-only control bits and 4 write-only interface status bits.

MST and TRX

These bits determine the operating mode of the serial I/O interface (Table 2).

Table 1 Operating modes of the serial I/O interface.

MST	TRX	mode
0	0	slave receiver
1	0	master receiver
0	1	slave transmitter
1	1	master transmitter

BB: Bus Busy

This bit indicates the status of the bus.

PIN: Pending Interrupt Not

PIN = '0' indicates that there is an interrupt pending. This causes a Serial Interrupt Request when the serial interrupt mechanism is enabled.

ESO: Enable Serial Output

The ESO flag enables/disables the serial I/O interface: ESO = logic 1 enables
ESO = logic 0 disables

BC0, BC1 and BC2

These bits indicate the number of bits received or transmitted in a serial data stream.

Bits ES0, BC0, BC1 and BC2 can only be written via software.

AL: Arbitration Lost

The AL flag is set via the hardware when the serial I/O interface, as a master transmitter, loses the bus arbitration procedure.

AAS: Addressed As Slave

This flag is set via the hardware when the interface detects either its own address or the 'general call' address as the first byte of a transfer and if the interface has been programmed to operate in the address recognition mode.

AD0: Address Zero

This flag is set via the hardware after the general call address is detected when the interface is operating in the address recognition mode.

LRB: Last Received Bit

This contains either the last data bit received or, for a transmitting device in the acknowledge mode, the acknowledge from the receiving device.

Bits AL, AAS, AD0 and LRB can only be read via software.

Serial clock control register S2

Bits 0 to 4 of S2 are used to set the frequency of the serial clock signal. When a 4,43 MHz crystal is used, the frequency of the serial clock can be varied between 100 kHz and 720 Hz. An asymmetrical clock with a HIGH to LOW ratio of 3 to 1 is produced by setting bit 5. The asymmetrical clock allows a microcontroller more time per clock period for sampling the data line, making the timing of this action less critical. Bit 6 is used to activate the acknowledge mode of the serial I/O. S2 is a write-only register.

Address register S0'

The address register contains the 7-bit address back-up latches and the bit (ALS) used to enable/disable the address recognition mode. Only when ES0 = 0 can the address register be written using the MOV S0,A and MOV S0,#data instructions.

Serial I/O interrupt logic

The interrupt logic is enabled by the EN SI instruction and disabled by DIS SI. When the interrupt logic is enabled, a pending interrupt results in a serial I/O interrupt to the controller, causing a jump to location 5 in the ROM. When the logic is disabled, the presence of an interrupt is still indicated by the PIN bit in register S1. Therefore, an interrupt can still be serviced but a vectored interrupt will not occur.

Interrupt system

External events and real-time on-chip peripherals require servicing by the CPU asynchronous to the execution of any particular section of code. To tie the asynchronous activities of these functions to normal program execution, three single-level nested interrupts are provided.

Each interrupt vectors to a separate location in the program memory for its service program. Each source can be individually enabled or disabled. When more than one interrupt occurs simultaneously, their priority will be: (1) external, (2) serial I/O and, (3) timer/event counter. An additional external interrupt can be created using the timer/event counter interrupt.

FUNCTIONAL DESCRIPTION (continued)

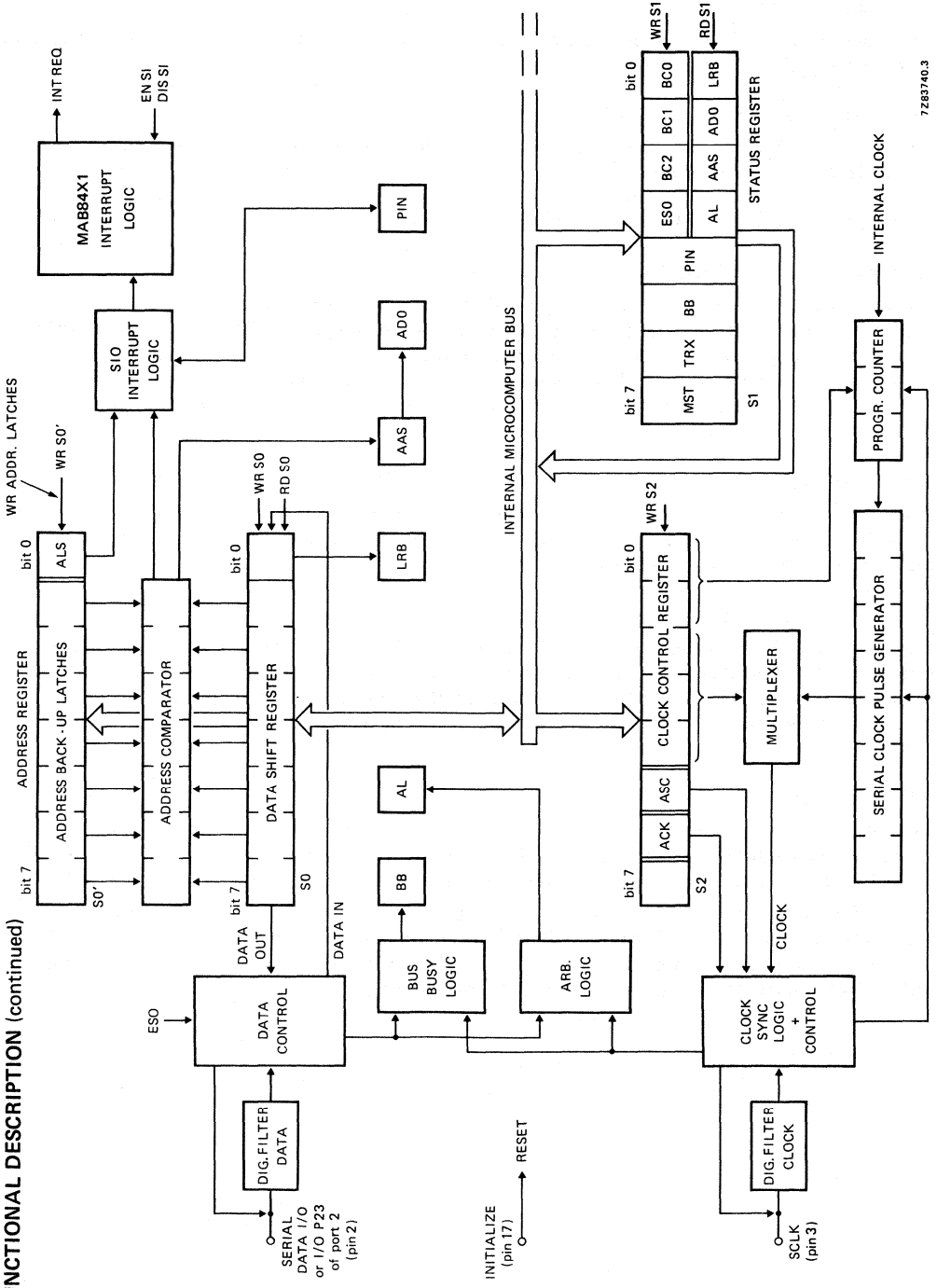


Fig. 8 The serial I/O interface.

Test input T1

The T1 input line can be used as:

- a test input for branch instructions,
- an input for zero voltage cross-over detection,
- an external input to the event counter.

An internal pull-up transistor is provided as a ROM mask option. This is useful when the input is from a switch or standard TTL output.

When T1 is used as a test input, the JT1 or JNT1 instructions test for a HIGH or a LOW respectively.

When used for zero-cross detection purposes, the T1 input must be coupled through a capacitor of typical value 1 μF and operation carried out using the T1 input without the pull-up transistor. The maximum input voltage amplitude is 3 V (peak-to-peak), with a maximum operational frequency of 1 kHz. The T1 input has an on-chip DC offset circuit which self-biases the input to its exact switching level of 1 V. As a consequence a small change will cause a digital transition to occur. The switching level of the T1 input circuit is within the bias voltage of ± 135 mV. Upon each positive cycle on the pin, the event counter is incremented and an overflow will set the timer flag TF. Zero cross-over detection used in conjunction with the timer/event counter interrupt, is useful in thyristor control of power equipment. Figure 9 illustrates, (a) the input waveform, (b) the input diagram and (c) the on-chip self-stabilized bias.

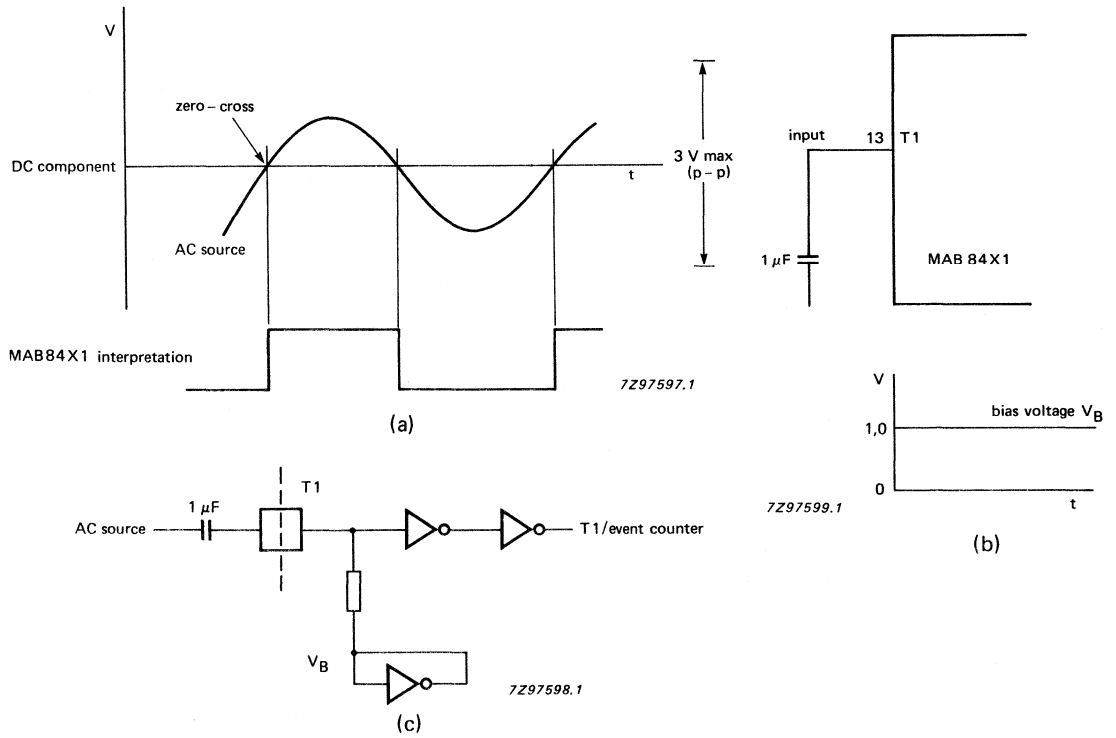


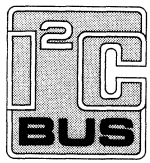
Figure 9 Zero-cross detection circuitry; (a) input waveform, (b) input diagram, (c) on-chip self-stabilized bias.

The operation of T1 as an input to the timer/event counter is described under the heading Timer/event counter.

High current outputs

Ten pins are provided that can sink high currents:

– P2.3 (serial data), pin 2	5 mA at 0,45 V (open drain),
– SCLK, pin 3	5 mA at 0,45 V (open drain),
– P1.0 – P1.7 *	10 mA at 1 V



Purchase of Philips' I²C components conveys a license under the Philips' I²C patent to use the components in the I²C-system provided the system conforms to the I²C specifications defined by Philips.

* P1.0 to P1.7 may be connected in parallel if their logic outputs are always the same.

FUNCTIONAL DESCRIPTION (continued)

Timer/event counter

An 8-bit binary up-counter is provided. This can count external events, machine cycles divided by 32, or machine cycles directly. When used as a timer, the input to the counter is either the overflow or input of a 5-bit prescaler. When used as an event counter, LOW to HIGH transitions on T1 (pin 13) are counted. The maximum rate at which the counter may be incremented is once every machine cycle (200 kHz for a 5 μs machine cycle). Figure 10 illustrates the timer/event counter.

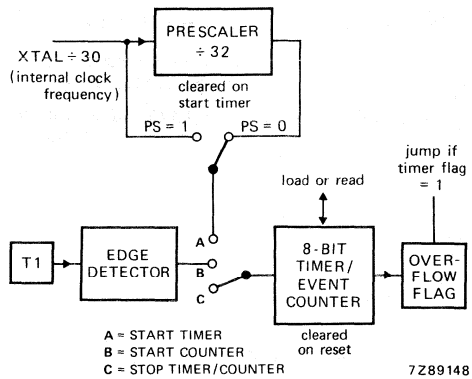


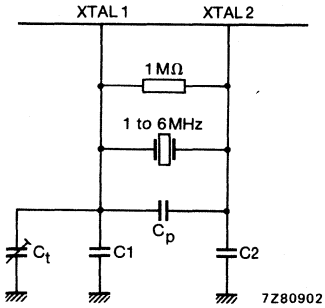
Fig. 10 The timer/event counter.

Differences between the MAB8021 and MAB8048 microcontrollers, and the MAB84X1 family.

	8021	8048	8401, 8411 8421, 8441, 8461
ROM capacity (bytes)	1K	1K	ROMless, 1K, 2K, 4K, 6K
RAM capacity (bytes)	64	64	128, 64, 64, 128, 128
paralel I/O lines	8 + 8 + 4	8 + 8 + 8	8 + 8 + 4
single inputs	1	3	2
serial I/O	no	no	yes, 2-line multi-transmitter
timer	8 bit	8 bit	8 bit
prescaler	mod. 32	mod. 32	mod. 1 & mod. 32
machine cycle time (μs)	10	2,5	5
for clock (MHz)	3	6	6
instruction set	8021	8048	8048 with omissions; 5 new serial I/O instructions; 2 new register instructions; 2 new control instructions; 1 new cond. branch instruction
interrupts	none	2 external timer/ event counter	3 external serial I/O timer/event counter
no. of pins (DIL)	28	40	68 (PLCC), 28

OSCILLATOR CIRCUITRY

Clock frequency is determined by using the internal oscillator or by connecting an external clock to XTAL1. Where the internal oscillator is used, the frequency is set by a crystal between XTAL1 and XTAL2, or by a ceramic resonator or an inductor, each with two associated capacitors, between XTAL1 and XTAL2 (see Fig. 11a). A machine cycle consists of 10 states, each state being 3 oscillator periods. The common 6 MHz crystal gives a 5 μs machine cycle. The MAB84X1 family has dynamic logic, and therefore, for adequate refreshing the oscillator frequency must be at least 1 MHz.



1. Crystal – AT-cut
2. Ceramic resonator
   C1 = C2 = 27 pF
   C1 may be trimmed
   $C_p \leqslant 6{,}75$ pF (parasitic capacitance)

Fig. 11a Quartz crystal or ceramic resonator mode.

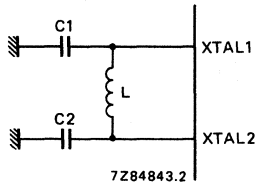


Fig. 11b LC pi-network.

LC oscillator timing

frequency	C1 = C2	L
3,0 MHz	33 pF	100 μH
4,0 MHz	33 pF	56 μH
4,4 MHz	33 pF	47 μH
5,0 MHz	33 pF	33 μH
6,0 MHz	33 pF	22 μH

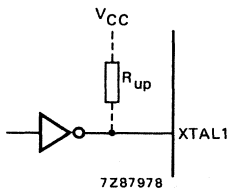


Fig. 11c External drive.

Drive XTAL1
Leave XTAL2 open

Driver may be high-speed CMOS or any TTL
t_r, $t_f < 10$ ns

PROGRAM STATUS WORD

The program status word (PSW) is an 8-bit word in the CPU which stores information about the current status of the microcontroller (Fig. 12). The PSW bits are:

bits 0, 1 and 2 — stack pointer bits (SP_0, SP_1, SP_2);

bit 3 — prescaler select (PS); 0 = divide-by-32; 1 = no prescaling;

bit 4 — working register bank select (RBS):
0 = register bank 0
1 = register bank 1;

bit 5 — not used (1);

bit 6 — auxiliary carry (AC):
half-carry bit is generated by an ADD instruction and used by the decimal adjust instruction DA A;

bit 7 — carry (CY):
the carry flag indicates that the previous operation has resulted in an overflow of the accumulator.

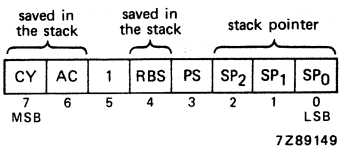


Fig. 12 Program status word.

All bits can be read using MOV A, PSW and bit 3 can be written with MOV PSW, A.

Bits 6 and 7 can be set and cleared by CPU operation. Bit 4 is changed by the SEL RB instruction, bit 3 by the MOV PSW,A instruction, and bits 0, 1 and 2 by the CALL, RET or RETR instructions and when an interrupt occurs. Bits 4, 6 and 7 are stored in the program counter stack during subroutine and interrupt calls. These bits are restored to the PSW with RETR (return and restore) instruction.

Note: The RET instruction has no restore feature and should not be used at the end of an interrupt because this would leave any further interrupts disabled.

The MAB84X1 family has arithmetic, logical and branching capabilities. The DA A, SWAP A, and XCHD instructions simplify BCD arithmetic and the handling of nibbles. The MOVP A,@A instruction permits efficient table look up from the current ROM page.

The conditional branch logic within the processor enables several conditions, internal and external to the processor, to be tested by the user's program. Table 2 lists the conditional branch instructions used to change the program execution sequence. The DJNZ instruction decrements a designated register and branches if the contents are not zero. This instruction makes the register an efficient program loop counter. The JMPP @A instruction allows multiway branches to destinations indirectly addressed by the contents of the accumulator.

Table 2 Conditional branches

TEST	JUMP CONDITION	JUMP INSTRUCTION
accumulator	0 or non-zero	JZ, JNZ
accumulator bit test	1	JB0 to JB7
carry flag	0 or 1	JNC, JC
timer overflow flag	1	JTF
test input $\overline{INT}$	0 or 1	JNT0, JT0
test input T1	0 or 1	JNT1, JT1
test flag 0	1	JF0
test flag 1	1	JF1
register	non-zero	DJNZ

RESET

A positive-going signal on the RESET input:

- sets the program counter to zero,
- selects location 0 of memory bank 0, and register bank 0,
- sets the stack pointer to zero ('000'B); pointing to RAM address 8,
- disable the interrupts (external, timer and serial I/O),
- stops the timer/event counter, then sets it to zero,
- sets the timer prescaler to divide-by-32,
- resets the timer flag,
- sets all ports to logic '1' (input mode),
- sets the serial I/O to slave receiver mode and disables serial I/O.

Automatic reset at power-up may be obtained by connecting the RESET pin to V_{CC} through a 1 μF capacitor C, together with a diode to V_{SS} (cathode to RESET pin). This arrangement is satisfactory, if both the voltage (V_{CC}) rise time and the oscillator start-up time do not exceed either 1 or 10 ms respectively.

The power-on reset circuit is shown in figure 13. At power-on the current drawn by RESET commences to charge the capacitor C. The difference between this increasing capacitor voltage and V_{CC} is known as V_{RESET}. The charging circuit is designed to hold V_{RESET} above the lower threshold of a Schmitt trigger arrangement long enough to effect a complete reset. The minimum time required; is the oscillator start-up time plus two machine cycles.

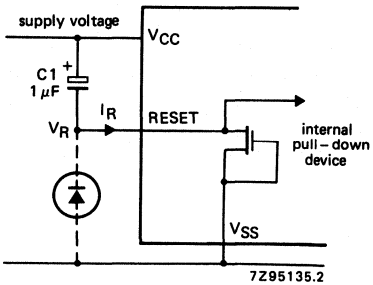


Fig. 13 Typical power-on reset circuitry.

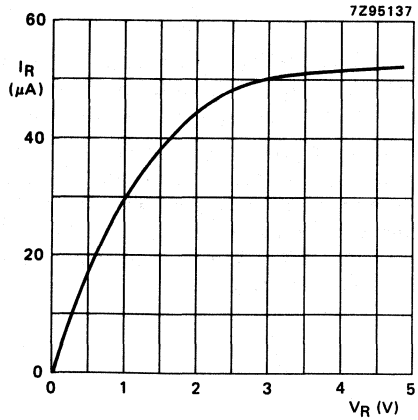


Fig. 14 Power-on reset input characteristics (typical).

INSTRUCTION SET

The instruction set consists of over 80 one and two byte instructions and is based on the MAB8048 instruction set. New instructions include those for serial I/O operation and memory bank selection. Program code efficiency is high because all ROM locations on a 256 byte page require only a single byte address.

Table 3 gives the instruction set of the MAB84X1 family and Table 4 shows the instruction map. The following symbols and abbreviations are used.

Note: During development of software on a PMDS or similar system, it is important to ensure that no jump instruction (direct or indirect), outreaches the final address range of the device.

symbol	description
A	the accumulator
AC	the auxiliary carry flag
addr	program memory address (11-bits)
Bb	bit designation (b = 0–7)
BS	the bank switch
C	carry flag
CLK	clock signal
CNT	event counter
D	nibble designation (4-bits)
DBF	program memory bank flip-flop
data	number or expression (8-bits)
F0, F1	flags 0 and 1
I	interrupt
$\overline{\text{INT}}$	external interrupt
P	'in-page' operation designation
Pp	port designation (p = 1, 2 or 4–7)
PSW	program status word
Rr	register designation (r = 0, 1 or 0–7)
SP	stack pointer
T	timer
TF	timer flag
T0, T1	test 0 and 1 inputs
#	immediate data prefix
@	indirect address prefix
$	current value of program counter
←	is replaced by
↔	is exchanged with

Table 3 MAB84XX family instruction set

	mnemonic	opcode (hex.)	bytes/ cycles	description	function		notes
ACCUMULATOR	ADD A, Rr	6*	1/1	Add register contents to A	(A)←(A) + (Rr)	r = 0–7	1
	ADD A, @Rr	60 61	1/1	Add RAM data, addressed by Rr, to A	(A)←(A) + ((R0)) (A)←(A) + ((R1))		1
	ADD A, #data	03 data	2/2	Add immediate data to A	(A)←(A) + data		1
	ADDC A, Rr	7*	1/1	Add carry and register contents to A	(A)←(A) + (Rr) + (C)	r = 0–7	1
	ADDC A, @Rr	70 71	1/1	Add carry and RAM data, addressed by Rr, to A	(A)←(A) + ((R0)) + (C) (A)←(A) + ((R1)) + (C)		1
	ADDC A, #data	13 data	2/2	Add carry and immediate data to A	(A)←(A) + data + (C)		1
	ANL A, Rr	5*	1/1	'AND' Rr with A	(A)←(A) AND (Rr)	r = 0–7	
	ANL A, @Rr	50 51	1/1	'AND' RAM data, addressed by Rr, with A	(A)←(A) AND ((R0)) (A)←(A) AND ((R1))		
	ANL A, #data	53 data	2/2	'AND' immediate data with A	(A)←(A) AND data		
	ORL A, Rr	4*	1/1	'OR' Rr with A	(A)←(A) OR (Rr)	r = 0–7	
	ORL A, @Rr	40 41	1/1	'OR' RAM data, addressed by Rr, with A	(A)←(A) OR ((R0)) (A)←(A) OR ((R1))		
	ORL A, #data	43 data	2/2	'OR' immediate data with A	(A)←(A) OR data		
	XRL A, Rr	D*	1/1	'XOR' Rr with A	(A)←(A) XOR (Rr)	r = 0–7	
	XRL A, @Rr	D0 D1	1/1	'XOR' RAM, addressed by Rr, with A	(A)←(A) XOR ((R0)) (A)←(A) XOR ((R1))		
	XRL A, #data	D3 data	2/2	'XOR' immediate data with A	(A)←(A) XOR data		
	INC A	17	1/1	increment A by 1	(A)←(A) + 1		
	DEC A	07	1/1	decrement A by 1	(A)←(A) – 1		
	CLR A	27	1/1	clear A to zero	(A)←0		
	CPL A	37	1/1	one's complement A	(A)←NOT(A)		
	RL A	E7	1/1	rotate A left	$(A_{n+1})\leftarrow(A_n)$ $(A_0)\leftarrow(A_7)$	n = 0–6	

	mnemonic	opcode (hex.)	bytes/ cycles	description	function	notes
ACCUMULATOR (cont.)	RLC A	F7	1/1	rotate A left through carry	$(A_{n+1})\leftarrow A_n$ n = 0–6 $(A_0)\leftarrow(C)$, $(C)\leftarrow(A_7)$	2
	RR A	77	1/1	rotate A right	$(A_n)\leftarrow(A_{n+1})$ n = 0–6 $(A_7)\leftarrow(A_0)$	
	RRC A	67	1/1	rotate A right through carry	$(A_n)\leftarrow(A_{n+1})$ n = 0–6 $(A_7)\leftarrow(C)$, $(C)\leftarrow(A_0)$	2
	DA A	57	1/1	decimal adjust A		2
	SWAP A	47	1/1	swap nibbles of A	$(A_{4-7})\leftrightarrow(A_{0-3})$	
DATA MOVES	MOV A, Rr	F*	1/1	move register contents to A	(A)←(Rr) r = 0–7	
	MOV A, @Rr	F0 F1	1/1	move RAM data, addressed by Rr, to A	(A)←((R0)) (A)←((R1))	
	MOV A, #data	23 data	2/2	move immediate data to A	(A)←data	
	MOV Rr, A	A*	1/1	move accumulator contents to register	(Rr)←(A) r = 0–7	
	MOV @Rr, A	A0 A1	1/1	move accumulator contents to RAM location addressed by Rr	((R0))←(A) ((R1))←(A)	
	MOV Rr, #data	B* data	2/2	move immediate data to Rr	(Rr)←data	
	MOV @Rr, #data	B0 data B1 data	2/2	move immediate data to RAM location addressed by Rr	((R0))←data ((R1))←data	
	XCH A, Rr	2*	1/1	exchange accumulator contents with Rr	(A)↔(Rr) r = 0–7	
	XCH A, @Rr	20 21	1/1	exchange accumulator contents with RAM data addressed by Rr	(A)↔((R0)) (A)↔((R1))	
	XCHD A, @Rr	30 31	1/1	exchange lower nibbles of A and RAM data addressed by Rr	$(A_{0-3})\leftrightarrow((R0_{0-3}))$ $(A_{0-3})\leftrightarrow((R1_{0-3}))$	
	MOV A, PSW	C7	1/1	move PSW contents to accumulator	(A)←(PSW)	
	MOV PSW, A	D7	1/1	move accumulator bit 3 to PSW_3	$(PSW_3)\leftarrow(A_3)$	3
	MOVP A, @A	A3	1/2	move indirectly addressed data in current page to A	$(PC_{0-7})\leftarrow(A)$, $(A)\leftarrow((PC))$	

FLAGS	CLR C	97	1/1	clear carry bit	(C)←0		2
	CPL C	A7	1/1	complement carry bit	(C)←NOT(C)		2
REGISTER	INC Rr	1*	1/1	increment register by 1	(Rr)←(Rr) + 1	r = 0–7	
	INC @Rr	10 11	1/1	increment RAM data, addressed by Rr, by 1	((R0))←((R0)) + 1 ((R1))←((R1)) + 1		
	DEC Rr	C*	1/1	decrement register by 1	(Rr)←(Rr) – 1	r = 0–7	
	DEC @Rr	C0 C1	1/1	decrement RAM data, addressed by Rr, by 1	((R0))←((R0)) – 1 ((R1))←((R1)) – 1		
BRANCH	JMP addr	● 4 address	2/2	unconditional jump within a 2K bank	(PC_{8-10})←$addr_{8-10}$ (PC_{0-7})←$addr_{0-7}$ (PC_{11-12})←MBFF 0–1		
	JMPP @A	B3	1/2	indirect jump within a page	(PC_{0-7})←((A))		
	DJNZ Rr, addr	E* address	2/2	decrement Rr by 1 and jump if not zero to addr	(Rr)←(Rr) – 1 if (Rr) not zero (PC_{0-7})←addr	r = 0–7	
	DJNZ @Rr, addr	E0 address	2/2	decrement RAM data, addressed by Rr, by 1 and jump if not zero to addr	((R0))←((R0)) – 1 if ((R0)) not zero (PC_{0-7})←addr		
		E1 address			((R1))←((R1)) – 1 if ((R1)) not zero (PC_{0-7})←addr		
	JBb addr	▲ 2 address	2/2	jump to addr if Acc. bit b = 1	if b = 1: (PC_{0-7})←addr	b = 0–7	
	JC addr	F6 address	2/2	jump to addr if C = 1	if C = 1: (PC_{0-7})←addr		
	JNC addr	E6 address	2/2	jump to addr if C = 0	if C = 0: (PC_{0-7})←addr		
	JZ addr	C6 address	2/2	jump to addr if A = 0	if A = 0: (PC_{0-7})←addr		
	JNZ addr	96 address	2/2	jump to addr if A is NOT zero	if A ≠ 0: (PC_{0-7})←addr		
	JT0 addr	36 address	2/2	jump to addr if T0 = 1	if T0 = 1: (PC_{0-7})←addr		
	JNT0 addr	26 address	2/2	jump to addr if T0 = 0	if T0 = 0: (PC_{0-7})←addr		
	JT1 addr	56 address	2/2	jump to addr if T1 = 1	if T1 = 1: (PC_{0-7})←addr		
	JNT1 addr	46 address	2/2	jump to addr if T1 = 0	if T1 = 0: (PC_{0-7})←addr		
	JTF addr	16 address	2/2	jump to addr if Timer Flag = 1	if TF = 1: (PC_{0-7})←addr		4
	JNTF addr	06 address	2/2	jump to addr if Timer Flag = 0	if TF = 0: (PC_{0-7})←addr		

	mnemonic	opcode (hex.)	bytes/ cycles	description	function	notes
TIMER/EVENT COUNTER	MOV A, T	42	1/1	move timer/event counter contents to accumulator	(A)←(T)	
	MOV T, A	62	1/1	move accumulator contents to timer/event counter	(T)←(A)	
	STRT CNT	45	1/1	start event counter		
	STRT T	55	1/1	start timer		
	STOP TCNT	65	1/1	stop timer/event counter		
	EN TCNTI	25	1/1	enable timer/event counter interrupt		
	DIS TCNTI	35	1/1	disable timer/event counter interrupt		
CONTROL	EN I	05	1/1	enable external interrupt		
	DIS I	15	1/1	disable external interrupt		
	SEL RB0	C5	1/1	select register bank 0	(RBS)←0	5
	SEL RB1	D5	1/1	select register bank 1	(RBS)←1	5
	SEL MB0	E5	1/1	select program memory bank 0	(MBFF0)←0, (MBFF1)←0	
	SEL MB1	F5	1/1	select program memory bank 1	(MBFF0)←1, (MBFF1)←0	
	SEL MB2	A5	1/1	select program memory bank 2	(MBFF0)←0, (MBFF1)←1	
	SEL MB3	B5	1/1	select program memory bank 3	(MBFF0)←1, (MBFF1)←1	
SUBROUTINE	CALL addr	▲ 4 address	2/2	jump to subroutine	((SP))←(PC), ($PSW_{4, 6, 7}$) (SP)←(SP) + 1 (PC_{9-10})←$addr_{8-10}$ (PC_{0-7})←$addr_{0-7}$ (PC_{11-12})←MBFF 0–1	6
	RET	83	1/2	return from subroutine	(SP)←(SP) −1 (PC)←((SP))	6
	RETR	93	1/2	return from interrupt and restore bits 4, 6, 7 of PSW	(SP)←(SP) −1 ($PSW_{4, 6, 7}$) + (PC)←((SP))	6

PARALLEL INPUT/OUTPUT	IN A, Pp	08 09 0A	1/2	input port p data to accumulator	(A)←(P0) (A)←(P1) (A)←(P2)	7
	OUTL Pp, A	38 39 3A	1/2	output accumulator data to port p	(P0)←(A) (P1)←(A) (P2)←(A)	
	ANL Pp, #data	98 data 99 data 9A data	2/2	AND port p data with immediate data	(P0)←(P0) AND data (P1)←(P1) AND data (P2)←(P2) AND data	
	ORL Pp, #data	88 data 89 data 8A data	2/2	OR port p data with immediate data	(P0)←(P0) OR data (P1)←(P1) OR data (P2)←(P2) OR data	
	OUTL PO,A	90	1/2	Output accumulator data to port ϕ	(P0)←(A)	9
SERIAL INPUT/OUTPUT	MOV A, S_n	0C 0D	1/2	move serial I/O register contents to accumulator	(A)←(S0) (A)←(S1)	8
	MOV S_n, A	3C 3D 3E	1/2	move accumulator contents to serial I/O register	(S0)←(A) (S1)←(A) (S2)←(A)	
	MOV S_n, #data	9C data 9D data 9E data	2/2	move immeidate date to serial I/O register	(S0)←data (S1)←data (S2)←data	
	EN SI	85	1/1	enable serial I/O interrupt		
	DIS SI	95	1/1	disable serial I/O interrupt		
	NOP	00	1/1	no operation		

Notes to Table 3.

1. PSW CY, AC affected
2. PSW CY affected
3. PSW PS affected
4. Execution of JTF and JNTF instructions resets the Timer Flag (TF).
5. PSW RBS affected
6. PSW SP_0, SP_1, SP_2 affected
7. (A) = 1111 P2.3, P2.2, P2,1, P2,0.
8. (S1) has a different meaning for read and write operation, see serial I/O interface.
9. Only for software-transfer from the MAB8021.

* : 8, 9, A, B, C, D, E, F
● : 0, 2, 4, 6, 8, A, C, E
▲ : 1, 3, 5, 7, 9, B, D, F

Table 4 MAB84X1 family instruction set

first hexadecimal chracter of opcode ↓ / second hexadecimal chracter of opcode →

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	NOP			ADD A, # data	JMP page 0	EN I	JNTF addr	DEC A	IN A,Pp 0	1	2		0	MOV A,Sn 1		
1	INC @Rr 0	1	JB0 addr	ADDC A, # data	CALL page 0	DIS I	JTF addr	INC A	0	1	2	INC Rr 3	4	5	6	7
2	XCH A, @Rr 0	1		MOV A, # data	JMP page 1	EN TCNTI	JNT0 addr	CLR A	0	1	2	XCH A,Rr 3	4	5	6	7
3	XCHD A, @Rr 0	1	JB1 addr		CALL page 1	DIS TCNTI	JT0 addr	CPL A	OUTL Pp,A 0	1	2		0	MOV Sn,A 1	2	
4	ORL A, @Rr 0	1	MOV A, T	ORL A, # data	JMP page 2	STRT CNT	JNT1 addr	SWAP A	0	1	2	ORL A,Rr 3	4	5	6	7
5	ANL A, @Rr 0	1	JB2 addr	ANL A, # data	CALL page 2	STRT T	JT1 addr	DA A	0	1	2	ANL A,Rr 3	4	5	6	7
6	ADD A, @Rr 0	1	MOV T, A		JMP page 3	STOP TCNT		RRC A	0	1	2	ADD A,Rr 3	4	5	6	7
7	ADDC A, @Rr 0	1	JB3 addr		CALL page 3			RR A	0	1	2	ADDC A,Rr 3	4	5	6	7
8				RET	JMP page 4	EN SI			ORL Pp, # data 0	1	2					
9	OUTL PO, A		JB4 addr	RETR	CALL page 4	DIS SI	JNZ addr	CLR C	ANL Pp, # data 0	1	2		0	MOV Sn, # data 1	2	
A	MOV @Rr, A 0	1		MOVP A, @A	JMP page 5	SEL MB2		CPL C	0	1	2	MOV Rr, A 3	4	5	6	7
B	MOV @Rr, # data 0	1	JB5 addr	JMPP @A	CALL page 5	SEL MB3			0	1	2	MOV Rr, # data 3	4	5	6	7
C	DEC @Rr 0	1			JMP page 6	SEL RB0	JZ addr	MOV A, PSW	0	1	2	DEC Rr 3	4	5	6	7
D	XRL A, @Rr 0	1	JB6 addr	XRL A, # data	CALL page 6	SEL RB1		MOV PSW, A	0	1	2	XRL A, Rr 3	4	5	6	7
E	DJNZ @Rr, addr 0	1			JMP page 7	SEL MB0	JNC addr	RL A	0	1	2	DJNZ Rr, addr 3	4	5	6	7
F	MOV A, @Rr 0	1	JB7 addr		CALL page 7	SEL MB1	JC addr	RLC A	0	1	2	MOV A, Rr 3	4	5	6	7

Table 5 shows the additional MAB84X1 family instructions (including the five for serial I/O operation) that are not part of the MAB8048 instruction set.

Table 5 MAB84X1 family instructions not in the MAB8048 instruction set

serial I/O	register	control	conditional branch

MOV A, S_n
MOV S_n, A
MOV S_n, #data
EN SI
DIS SI
	DEC @Rr
	DJNZ @Rr, addr
		SEL MB2
		SEL MB3
			JNTF addr

Table 6 shows the MAB8048 instructions omitted from the MAB84X1 family instruction set.

Table 6 MAB8048 instructions not in the MAB84X1 family instruction set

data moves	flags	branch	control

MOVX A, @R
MOVX @R, A
MOVP3 A, @A
MOVD A, P
MOVD P, A
ANLD P, A
ORLD P, A
	CLR F0
	CPL F0
	CLR F1
	CPL F1
		* JNI addr
		JF0 addr
		JF1 addr
		
		* replaced by
		JT0 JNT0.
			ENT0 CLK

ABSOLUTE MAXIMUM RATINGS

Limiting values in accordance with the Absolute Maximum System (IEC 134)

Stress above those listed under 'Absolute maximum ratings' may cause permanent damage to the device. This is a stress rating only and functional operation of the device, at these, or any other conditions above those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

parameter	symbol	min.	max.	unit
Input voltage on any pin with respect to ground (V_{SS})	V_I	−0,5	+7	V
Total power dissipation				
SOT-117, 28-lead DIL	P_{tot}	−	1	W
SOT-136, 28-lead DIL	P_{tot}	−	0,6	W
Input/output current for all pins except port 1	I_I, I_O	−	10	mA
Input/output current for port 1	I_I, I_O	−	20	mA
Storage temperature	T_{stg}	−65	+150	°C
Operating temperature				
standard	T_{amb}	0	+70	°C
extended	T_{amb}	−40	+85	°C
automotive	T_{amb}	−40	+110	°C

DC CHARACTERISTICS

V_{CC} = 5 V (10%); V_{SS} = 0 V; all voltages with respect to V_{SS} unless otherwise specified

parameter	conditions	symbol	min.	max.	unit
Supply current					
MAB	0 to + 70 °C	I_{CC}	–	85	mA
MAF	–40 to + 85 °C	I_{CC}	–	100	mA
MAF84A	–40 to + 110 °C	I_{CC}	–	100	mA
Inputs					
Input voltage LOW (except P2.3 and SCLK)		V_{IL}	–0,5	0,8	V
Input voltage LOW (P2.3 and SCLK)		V_{IL1}	–0,5	1,5	V
Input voltage HIGH (all inputs except XTAL1, P2.3 and SCLK)		V_{IH}	2	V_{CC} + 0,5	V
Input voltage HIGH (XTAL1, P2,3 and SCLK)		V_{IH1}	3,0	V_{CC} + 0,5	V
Outputs					
Output voltage LOW (P0.0–P0.7)	I_{OL} = 1,6 mA	V_{OL}	–	0,45	V
Output voltage LOW (P1.0–P1.7 for 8401/11/21/41/61)	I_{OL12} = 10 mA	V_{OL12}	–	1,0	V
Output voltage LOW (P2.0–P2.2)	I_{OL2} = 1,6 mA	V_{OL2}	–	0,45	V
Output voltage LOW (P2.3, SCLK)	I_{OL3} = 5 mA	V_{OL3}	–	0,45	V
Output voltage LOW (non-standard pins of bond-out versions)	I_{OL4} = 0,4 mA	V_{OL4}	–	0,45	V
Output voltage HIGH (all outputs unless open drain)	I_{OH} = –50 μA	V_{OH}	2,4	–	V
Output leakage current	$V_{SS} < V_I < V_{CC}$	$\pm I_{OL}$	–	10	μA

AC CHARACTERISTICS (all versions except bond-out)

V_{CC} = 5 V ± 10%; V_{SS} = 0 V.

parameter	symbol		min.	max.	unit
Frequency	f_{XTAL}	MAB/MAF84X1	1	6	MHz
		MAF84AX1	1	5	MHz
Cycle time	t_{CY}	MAB/MAF84X1	5	30	μs
		MAF84AX1	6	30	μs

AC CHARACTERISTICS (bond-out versions)

V_{CC} = 5 V ± 10%; V_{SS} = 0 V.

parameter	symbol	min.	max.	unit
f_{CL} = 6 MHz				
Control pulse duration $\overline{PSEN}$ (9CP)	t_{CC}	1,5	9	μs
Address to $\overline{PSEN}$ L set-up (1CP)	t_{AS}	167	–	ns
Data to $\overline{PSEN}$ H set-up (1CP + 120 ns)	t_{DS}	600	–	ns
Data hold time	t_{DR}	0	–	ns
Address to data-in (10CP–t_{DS})	t_{AD}	–	1,07	μs
Time from $\overline{PSEN}$ L to C1 (3CP)	t_{PC}	500	–	ns
Time from $\overline{INTA}$ L to $\overline{PSEN}$ (3CP)	t_{IP0}	500	–	ns
Time from $\overline{INTA}$ H to $\overline{PSEN}$ (6CP)	t_{IP1}	1	–	μs
$\overline{HALT}$ set-up to $\overline{PSEN}$ (15CP)	t_{HS}	2,5	–	μs
$\overline{HALT}$ hold time from $\overline{PSEN}$ (3CP)	t_{HH}	500	–	ns

Note: CP = clock pulse.

T1 ZERO-CROSS CHARACTERISTICS

T_{amb} = 0 to + 70 °C; V_{CC} = 5 V ± 10%, V_{SS} = 0 V; C_L = 80 pF

parameter	conditions	symbol	min.	max.	unit
Zero-cross detection input (T1) peak-topeak	AC coupled, C = 1,0 μF	$V_{ZX(p-p)}$	1	3	V
Zero-cross accuracy	50 Hz sine wave	A_{ZX}	–	± 135	mV
Zero-cross detection input frequency (T1)		F_{ZX}	0,05	1	kHz

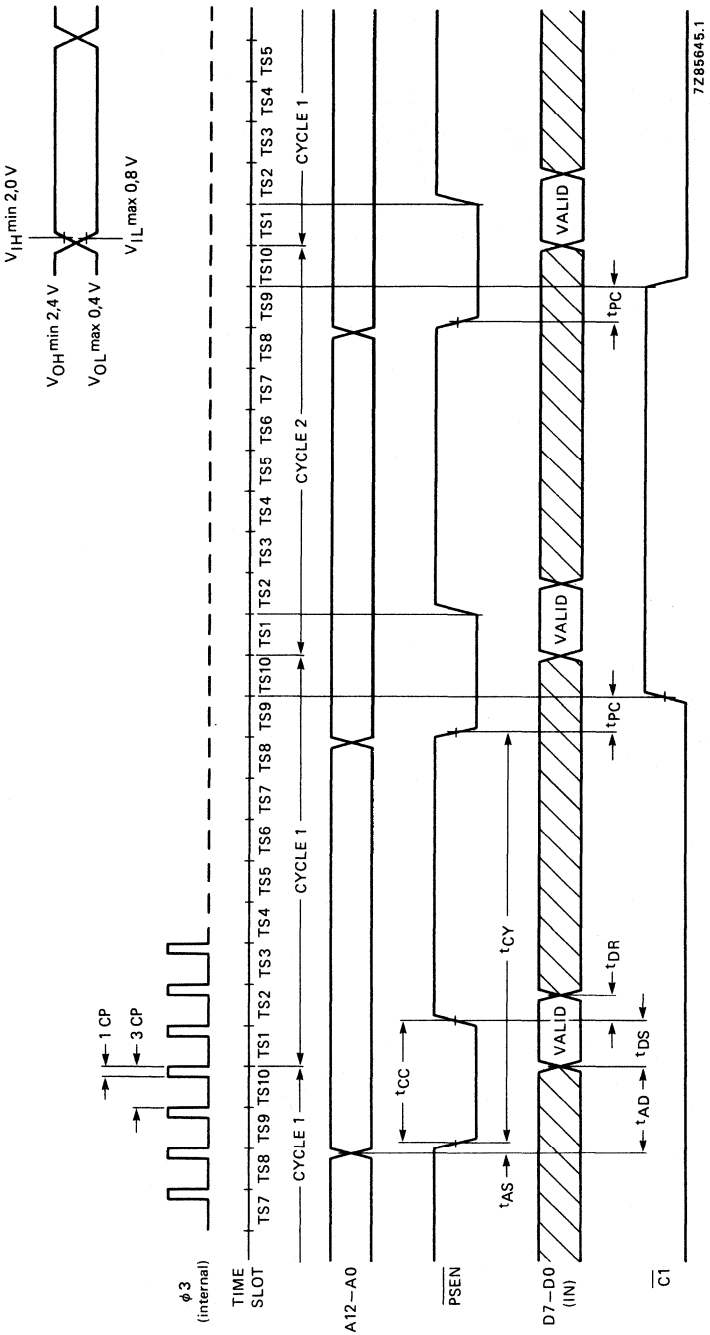


Fig. 15 Memory access timing MAB8401B/WP and I/O voltage parameters.

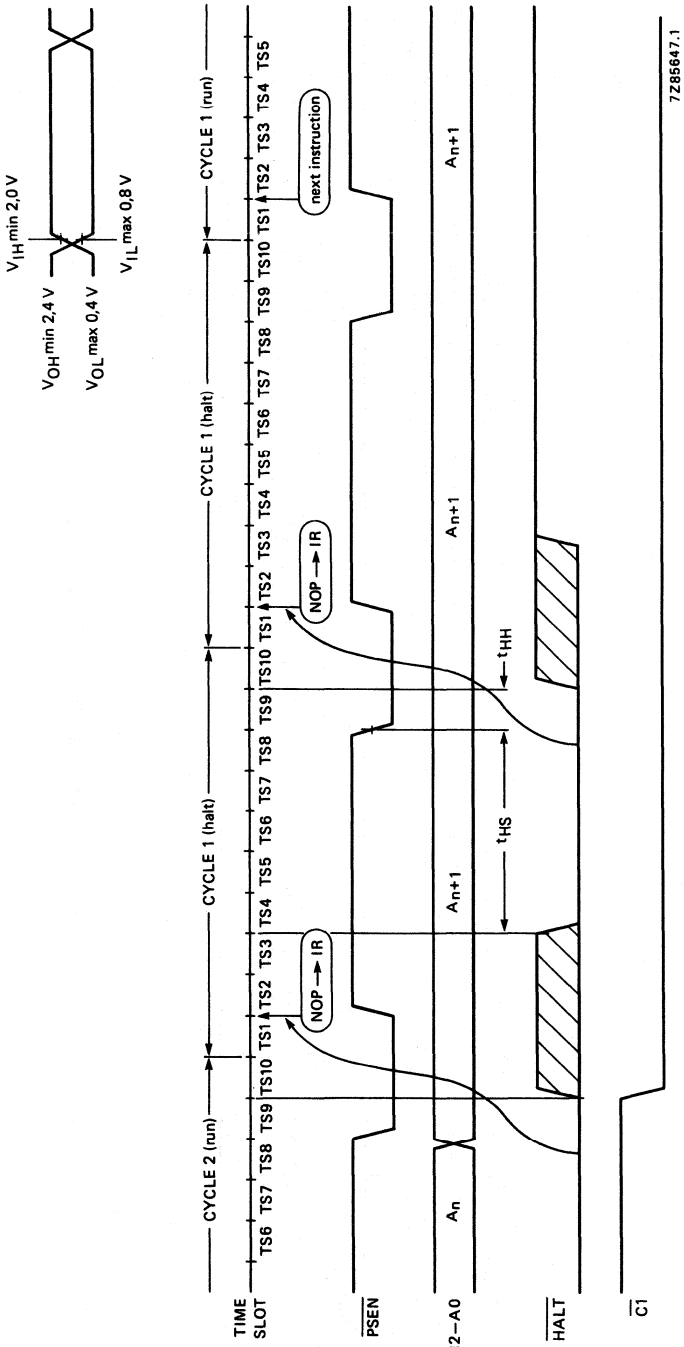


Fig. 16 $\overline{HALT}$ timing MAB8401WP and I/O voltage parameters.

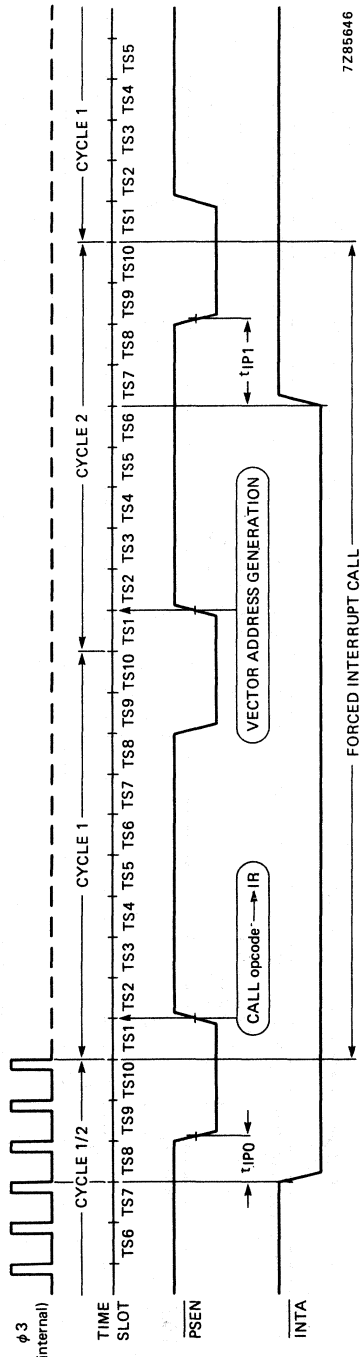


Fig. 17 $\overline{\text{INTA}}$ timing MAB8401WP.

SINGLE-CHIP 8-BIT MICROCONTROLLER

DESCRIPTION

The MAB8422/8442 is a high-performance microcontroller incorporating dedicated hardware, memory capacity and I/O lines. This dedication means a microcontroller can be economically installed in high-volume products where its main function is control.

The MAB8422/8442 is a 20 pin, single-chip 8-bit microcontroller that has been developed from the 28 pin MAB8421/8441 microcontrollers. The versions are:

- MAB8422 - 2K x 8 ROM/64 bytes RAM
- MAB8442 - 4K x 8 ROM/128 bytes RAM

Each version has 15 I/O port lines comprising one 8-bit parallel port (P0), one 2-bit parallel port (P1.0 and P1.1 that are shared with the serial I/O lines SDA and SCL), one 3-bit parallel port (P2.0 - P2.2) and two input lines ($\overline{INT}$/T0 and T1).

The serial I/O interface is I^2C compatible and therefore the MAB8422/8442 can operate as a slave or a master in single and multi-master systems. Conversion from parallel to serial data when transmitting, and vice versa when receiving, is done mainly in software. There is a minimum of hardware for the serial I/O implemented. This hardware is controlled by the status of the SDA and SCL lines and can be read or written under software control. Standard software for I^2C-bus control is available upon request. For detailed information see the user manual 'Single-chip 8-bit microcontrollers'.

Features

- 8-bit: CPU, ROM, RAM and I/O
- 20 pin package
- MAB8422: 2K x 8 ROM/64 bytes RAM
- MAB8442: 4K x 8 ROM/128 bytes RAM
- 13 quasi-bidirectional I/O port lines
- Two testable inputs T1 and $\overline{INT}$/T0
- High current output on P0 (I_{OL} = 10 mA at V_{OL} = 1 V)
- One interrupt line combined with the testable input line $\overline{INT}$/T0
- Single-level interrupts: external, timer/event counter, serial I/O
- I^2C-compatible serial I/O that can be used in single or multi-master systems (serial I/O data and clock via P1.0 and P1.1 port lines, respectively)
- 8-bit programmable timer/event counter
- Internal oscillator, generated with inductor, crystal, ceramic resonator or external source
- Over 80 instructions (based on MAB8048)
- All instructions 1 or 2 cycles, cycle time dependent on oscillator frequency
- Single power supply
- Operating temperature ranges: 0 to +70 °C (MAB84X2)
  −40 to +85 °C (MAF84X2)
  −40 to +110 °C (MAF84AX2)

PACKAGE OUTLINES

MAB/MAF84X2, MAF84AX2: 20-lead DIL; plastic (SOT-146).

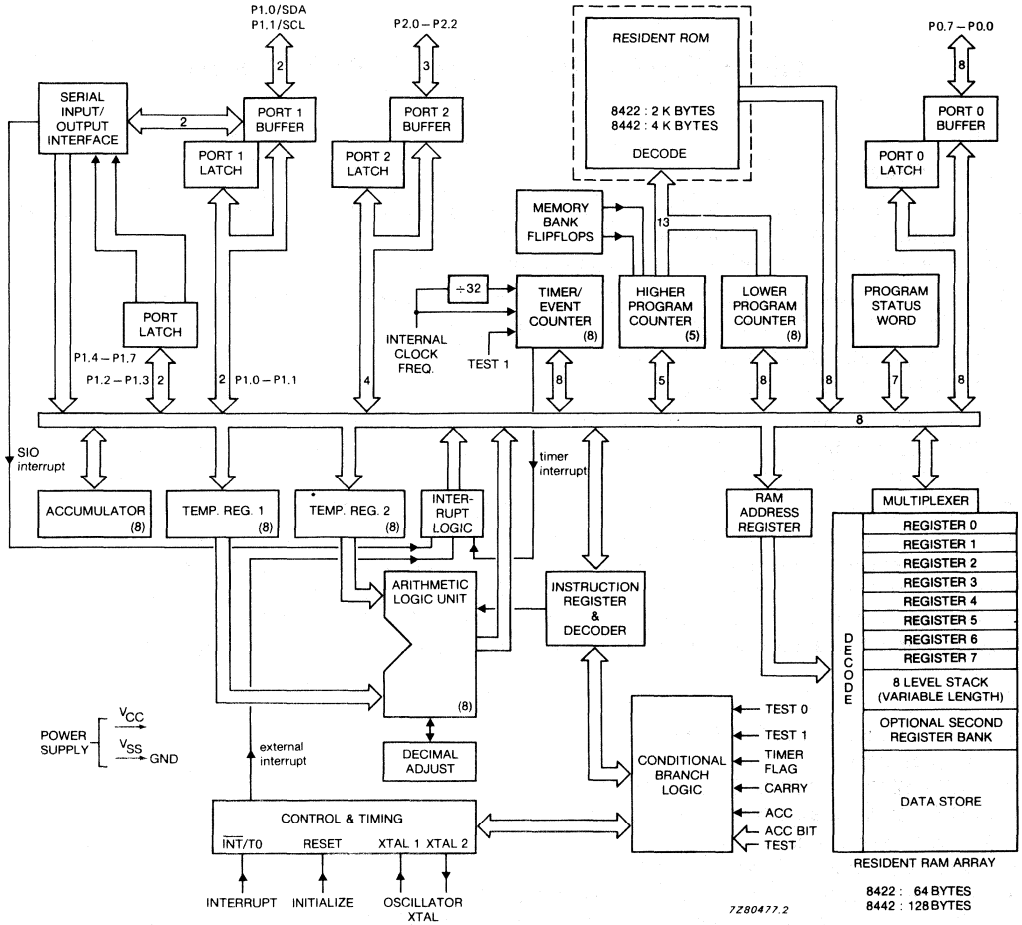


Fig. 1 Block diagram of the MAB8422/8442.

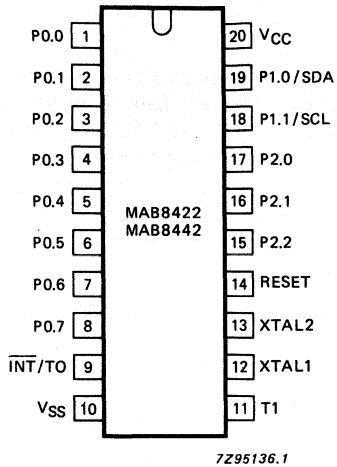


Fig. 2 Pinning diagram.

PINNING

Designation	Pin number	Function
P0.0 - P0.7	1-8	8-bit quasi-bidirectional I/O port (Port 0 high current output).
$\overline{INT}$/T0	9	External interrupt input (sensitive to a negative going edge) and/or input, testable using the JT0 or JNT0 instructions.
V_{SS}	10	Ground.
T1	11	Input pin, testable using the JT1 or JNT1 instructions. It can be designated as event counter input using the STRT CNT instruction. It can also be used to detect zero cross-over of slowly moving a.c. inputs.
XTAL1	12	Connection to timing component that determines the frequency of the internal oscillator. It is also the input for an external clock source.
XTAL2	13	Connection to the other side of the timing component.
RESET	14	Input to initialize the processor (active HIGH).
P2.0-P2.2	17-15	Quasi-bidirectional port.
P1.1/SCL	18	Quasi-bidirectional port in parallel port mode. Serial clock in serial I/O mode.
P1.0/SDA	19	Quasi-bidirectional port in parallel port mode. Serial data I/O in serial I/O mode.
V_{CC}	20	Power supply.

FUNCTIONAL DESCRIPTION

Program and data memory

The non-volatile program memory (ROM), as shown in Fig. 3, is arranged in two banks of 2K bytes, that are selected by SEL MB instructions, and each bank is further divided into 256-byte pages. Only the unconditional jump instructions (JMP and CALL) can be used to cross page boundaries. Memory bank boundaries can also be crossed using these instructions provided that the appropriate memory bank has been selected.

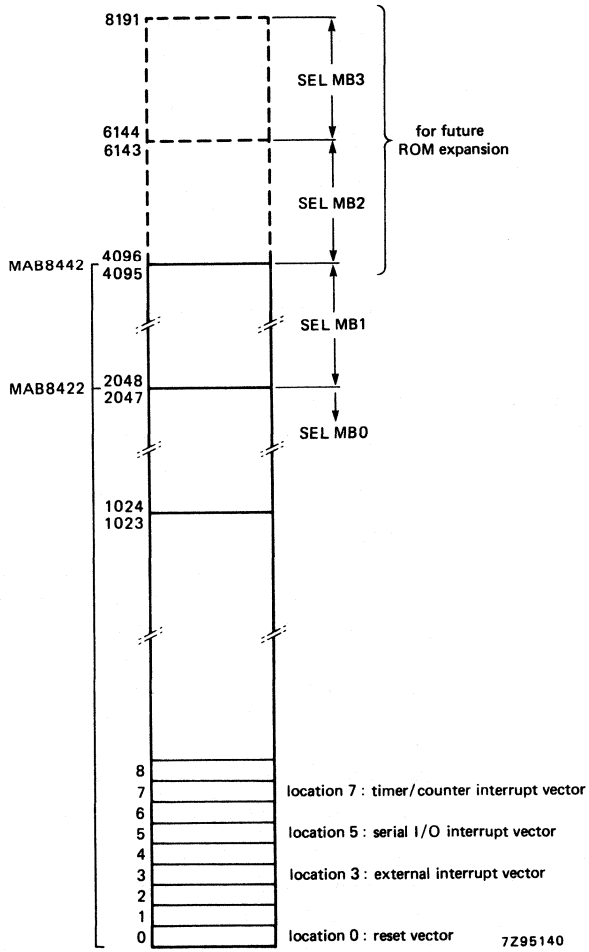


Fig. 3 Program memory map.

In the volatile data memory (RAM), all locations are indirectly addressable using RAM pointer registers and up to 16 designated locations can be addressed directly. The memory also includes an 8-level program counter stack addressed by a 3-bit stack pointer and two register banks, each with 8 registers. The data memory is shown in Fig. 4.

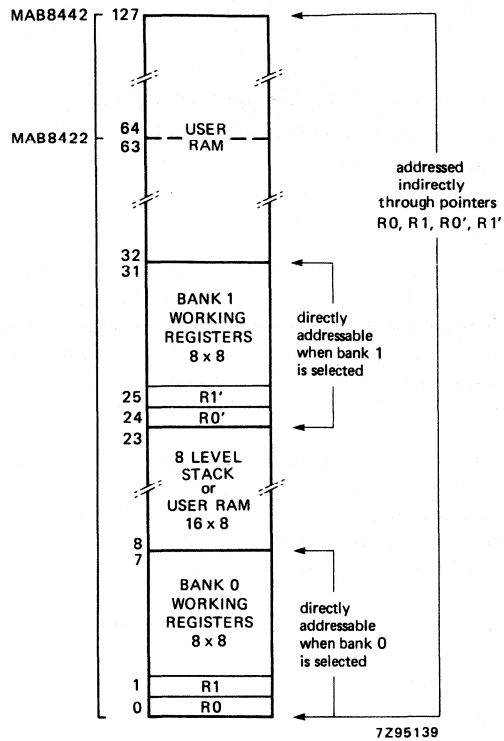


Fig. 4 Data memory map.

Instruction set

The instruction set consists of over 80 one and two byte instructions. It is identical to the MAB84X1 instruction set except that the instructions MOV Sn, A, MOV A, Sn and MOV Sn, #data are not used. Program code efficiency is high because all RAM locations on a 256 byte page require only a single byte address.

On-chip peripheral functions

In addition to the CPU and memories, an interrupt system, I/O facilities, and an 8-bit timer/event counter are integrated on-chip to assist the CPU in repetitious, complicated or time-critical tasks. The I/O facilities include the I/O pins, parallel ports and a serial I/O port sharing the two pins of the parallel port P1.

FUNCTIONAL DESCRIPTION (continued)

I/O facilities (see Fig. 5)

The MAB8422/8442 has 13 I/O lines and 2 testable inputs arranged as:

- An 8 line parallel port P0.0-P0.7, high current outputs with three optional output configurations:
  - A push-pull output with pull-up (Fig. 5 (a))
  - Open drain with pull-up (Fig. 5 (b))
  - Open drain without pull-up (Fig. 5 (c))
- A 2 line parallel/serial port P1.0/SDA and P1.1/SCL, open-drain without pull-up, as output configuration. Schmitt-trigger input. After RESET, P1.0/SDA and P1.1/SCL will be in the parallel port mode. To stay in this mode the internal port latches, P1.4 and P1.3 must be kept in the logic '1' state. Inputs P1.2-P1.7 are not valid in the parallel port mode. After a RESET, the microcontroller remains in the parallel port mode until the serial I/O mode is enabled.
- A 3 line parallel port P2.0-P2.2 with the same output configurations as P0.0-P0.7 but without high-current output;
- An external interrupt and test input INT/T0, which when used as a test input can be tested by the conditional jump instructions JT0 and JNT0;
- A test input T1, tested by the conditional jump instructions JT1 and JNT1. T1 can also be used as an input to the timer/event counter or to detect zero cross-over of slowly moving AC signals. This test or input line can be ordered with port option 5(b) or 5(c).

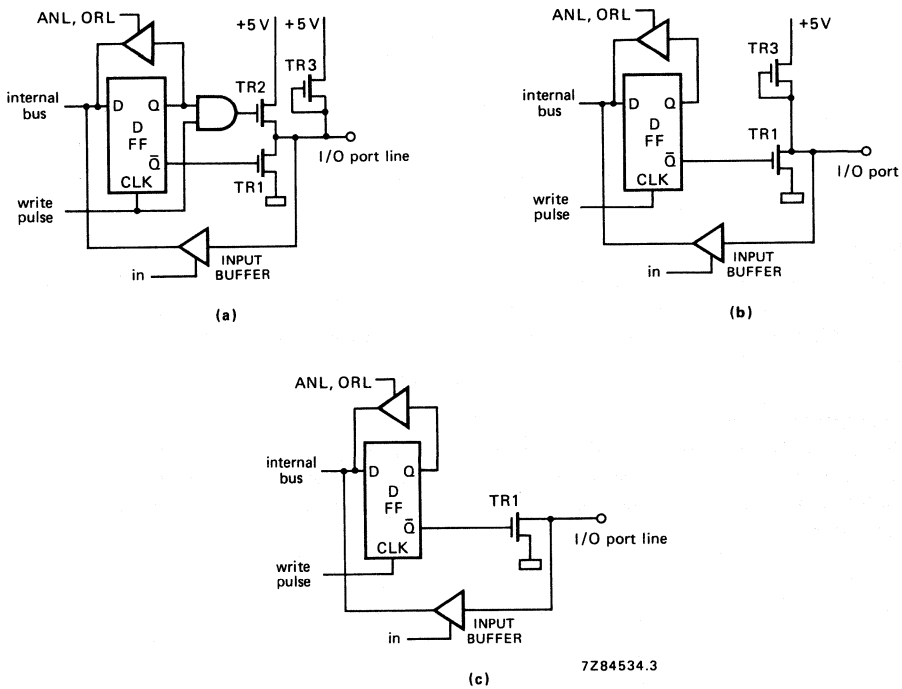


Fig. 5 Port option configurations.

Test input T1

The T1 input line can be used as:

- a test input for branch instructions,
- an input for zero voltage cross-over detection,
- an external input to the event counter.

An internal pull-up transistor is provided as a ROM mask option. This is useful when the input is from a switch or standard TTL output.

When T1 is used as a test input, the JT1 or JNT1 instructions test for a HIGH or a LOW respectively.

Zero cross-over detection

When used for zero-cross detection purposes, the T1 input must be coupled through a capacitor of typical value 1 μF and operation carried out using the T1 input without the pull-up transistor. The maximum input voltage amplitude is 3 V (peak-to-peak), with a maximum operational frequency of 1 kHz. The T1 input has an on-chip DC offset circuit which self-biases the input near to its exact switching level of 1 V. As a consequence a small change will cause a digital transition to occur. The switching level of the T1 input circuit is within the bias voltage of ±135 mV. Upon each positive cycle on the pin, the event counter is incremented and an overflow will set the timer flag TF. Zero cross-over detection used in conjunction with the timer/event counter interrupt, is useful in thyristor control of power equipment. Figure 6 illustrates, (a) the input waveform, (b) the input diagram and (c) the on-chip self-stabilized bias.

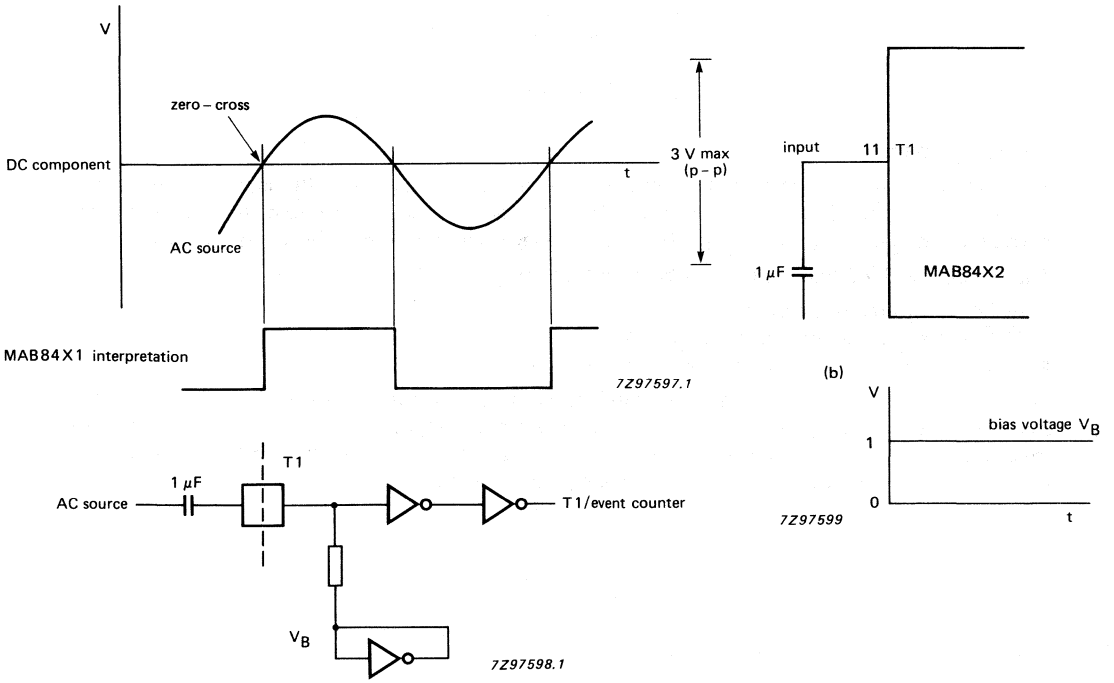


Fig. 6 Zero-cross voltage detection circuitry.

FUNCTIONAL DESCRIPTION (continued)

Timer/event counter

An 8-bit binary up-counter is provided. This can count external events, modulo-32 machine cycles, or machine cycles directly. When used as a timer, the input to the counter is either the overflow or input of a 5-bit prescaler. When used as an event counter, LOW to HIGH transitions on T1 (pin 13) are counted. The maximum rate at which the counter may be incremented is once every machine cycle (200 kHz for a 5 μs machine cycle). Fig. 7 illustrates the timer/event counter.

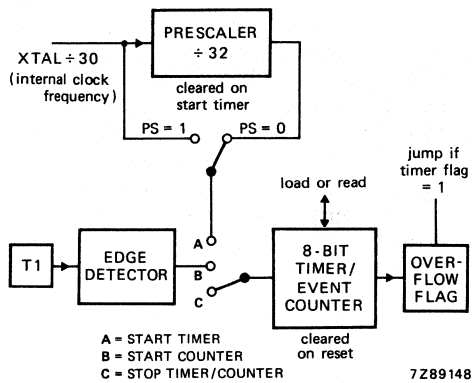


Fig. 7 Timer/event counter.

Interrupt system

External events and real-time on-chip peripherals require servicing by the CPU asynchronous to the execution of any particular section of code. To tie the asynchronous activities of these functions to normal program execution a multiple-source, single-level nested interrupt system is provided.

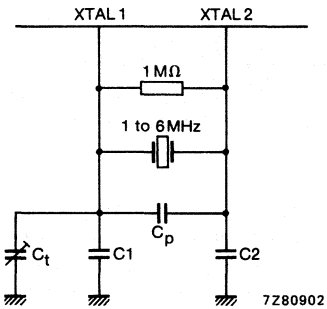
The MAB8422/8442 handles interrupts from three sources as follows:

- $\overline{\text{INT}}$/T0; externally via pin 9
- $\overline{\text{SIOINT}}$; from the internal serial I/O port
- TCNT interrupt; from the internal timer/event counter.

Each interrupt vectors to a separate location in the program memory for its service program. Each source can be individually enabled or disabled. When more than one interrupt occurs simultaneously, their priority will be: (1) external, (2) serial I/O and, (3) timer/event counter. An additional external interrupt can be created using the timer/event counter interrupt.

OSCILLATOR CIRCUITRY (see Fig. 8)

The clock frequency is determined by using the internal oscillator or by connecting an external clock to XTAL1. Where the internal oscillator is used the frequency is set by a crystal, a ceramic resonator or an inductor (each with associated capacitors) between XTAL1 and XTAL2. A machine cycle consists of 10 states, each state being 3 oscillator periods. The MAB8422/8442 has a dynamic logic, and therefore, for adequate refreshing the oscillator frequency must be at least 1 MHz.



1. Crystal - AT-cut
2. Ceramic resonator
   $C_1 = C_2 = 27$ pF
   C_t is optional
   $C_p < 6{,}75$ pF (parasitic capacitance)

Fig. 8(a) Quartz crystal or ceramic resonator mode.

If the frequency has to be trimmed, then a trimmer capacitor C_t should be connected in parallel with the fixed capacitor C_1.

Table 1 shows the LC values for timing generation with the LC oscillator.

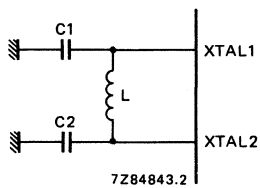
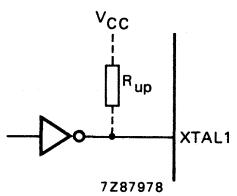


Fig. 8(b) LC pi-network.

Table 1 LC oscillator timing

frequency	C1 = C2	L
3,0 MHz	33 pF	100 μH
4,0 MHz	33 pF	56 μH
4,4 MHz	33 pF	47 μH
5,0 MHz	33 pF	33 μH
6,0 MHz	33 pF	22 μH



Drive XTAL1
Leave XTAL2 open

Driver may be high-speed CMOS or any TTL
t_r, $t_f < 10$ ns

Fig. 9 External drive.

RESET

A positive-going signal on the RESET input:

- sets the program counter to zero,
- selects location 0 of memory bank 0, and register bank 0,
- sets the stack pointer to zero (000); pointing to RAM address 8,
- disables the interrupts (external), timer and serial I/O),
- stops the timer/event counter, then sets it to zero,
- sets the timer prescaler to modulo-32,
- resets the timer flag,
- sets all ports to logic '1' (input mode),
- sets ports P1.0/SDA and P1.1/SCL to the parallel part mode and disables the serial I/O.

Automatic reset at power-up may be obtained by connecting the RESET pin to V_{CC} through a 1 μF capacitor C1, together with a diode to V_{SS} (cathode to RESET pin). This arrangement is satisfactory, if both the voltage (V_{CC}) rise time and the oscillator start-up time do not exceed either 1 or 10 ms respectively.

The power-on reset circuit is shown in Fig. 11; the input characteristics are shown in Fig. 12. At power-on the current drawn by RESET commences to charge the capacitor C1. The difference between this increasing capacitor voltage and V_{CC} is known as V_{RESET}. The charging circuit is designed to hold V_{RESET} above the lower threshold of a schmitt trigger arrangement, long enough to effect a complete reset. The minimum time required is the oscillator start-up time, plus two machine cycles.

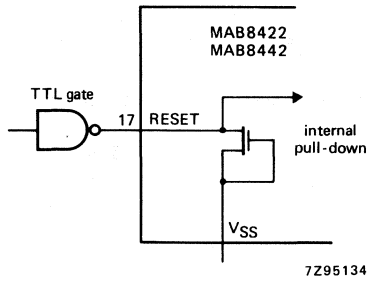


Fig. 10 External reset.

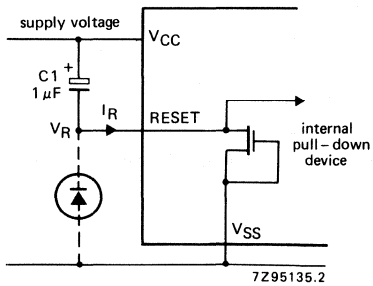


Fig. 11 Power-on reset circuitry.

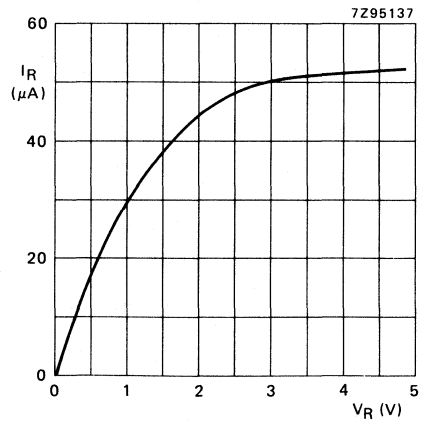


Fig. 12 Power-on reset input characteristics.

INSTRUCTION SET

The instruction set consists of over 80 one and two byte instructions and is based on the MAB8048 instruction set. New instructions include those for serial I/O operation and memory bank selection. Program code efficiency is high because all ROM locations on a 256 byte page require only a single byte address.

Table 2 gives the instruction set of the MAB84X2 family and Table 3 shows the instruction map. The following symbols and abbreviations are used.

symbol	description
A	the accumulator
AC	the auxiliary carry flag
addr	program memory address (11-bits)
Bb	bit designation (b = 0–7)
BS	the bank switch
C	carry flag
CLK	clock signal
CNT	event counter
D	nibble designation (4-bits)
DBF	program memory bank flip-flop
data	number of expression (8-bits)
F0, F1	flags 0 and 1
I	interrupt
$\overline{INT}$	external interrupt
P	'in-page' operation designation
Pp	port designation (p = 1, 2 or 4–7)
PSW	program status word
Rr	register designation (r = 0, 1 or 0–7)
SP	stack pointer
T	timer
TF	timer flag
T0, T1	test 0 an 1 inputs
#	immediate data prefix
@	indirect address prefix
S	current value of program counter
←	is replaced by
↔	is exchanged with

Table 2 Instruction set

	mnemonic	opcode (hex.)	bytes/ cycles	description	function		notes
ACCUMULATOR	ADD A, Rr	6*	1/1	Add register contents to A	(A)←(A) + (Rr)	r = 0–7	1
	ADD A, @Rr	60 61	1/1	Add RAM data, addressed by Rr, to A	(A)←(A) + ((R0)) (A)←(A) + ((R1))		1
	ADD A, #data	03 data	2/2	Add immediate data to A	(A)←(A) + data		1
	ADDC A, Rr	7*	1/1	Add carry and register contents to A	(A)←(A) + (Rr) + (C)	r = 0–7	1
	ADDC A, @Rr	70 71	1/1	Add carry and RAM data, addressed by Rr, to A	(A)←(A) + ((R0)) + (C) (A)←(A) + ((R1)) + (C)		1
	ADDC A, #data	13 data	2/2	Add carry and immediate data to A	(A)←(A) + data + (C)		1
	ANL A, Rr	5*	1/1	'AND' Rr with A	(A)←(A) AND (Rr)	r = 0–7	
	ANL A, @Rr	50 51	1/1	'AND' RAM data, addressed by Rr, with A	(A)←(A) AND ((R0)) (A)←(A) AND ((R1))		
	ANL A, #data	53 data	2/2	'AND' immediate data with A	(A)←(A) AND data		
	ORL A, Rr	4*	1/1	'OR' Rr with A	(A)←(A) OR (Rr)	r = 0–7	
	ORL A, @Rr	40 41	1/1	'OR' RAM data, addressed by Rr, with A	(A)←(A) OR ((R0)) (A)←(A) OR ((R1))		
	ORL A, #data	43 data	2/2	'OR' immediate data with A	(A)←(A) OR data		
	XRL A, Rr	D*	1/1	'XOR' Rr with A	(A)←(A) XOR (Rr)	r = 0–7	
	XRL A, @Rr	D0 D1	1/1	'XOR' RAM, addressed by Rr, with A	(A)←(A) XOR ((R0)) (A)←(A) XOR ((R1))		
	XRL A, #data	D3 data	2/2	'XOR' immediate data with A	(A)←(A) XOR data		
	INC A	17	1/1	increment A by 1	(A)←(A) + 1		
	DEC A	07	1/1	decrement A by 1	(A)←(A) −1		
	CLR A	27	1/1	clear A to zero	(A)←0		
	CPL A	37	1/1	one's complement A	(A)←NOT(A)		
	RL A	E7	1/1	rotate A left	$(A_{n+1})\leftarrow(A_n)$ $(A_0)\leftarrow(A_7)$	n = 0–6	

	mnemonic	opcode (hex.)	bytes/ cycles	description	function		notes
ACCUMULATOR (cont.)	RLC A	F7	1/1	rotate A left through carry	$(A_{n+1})\leftarrow A_n$ $(A_0)\leftarrow(C)$, $(C)\leftarrow(A_7)$	n = 0–6	2
	RR A	77	1/1	rotate A right	$(A_n)\leftarrow(A_{n+1})$ $(A_7)\leftarrow(A_0)$	n = 0–6	
	RRC A	67	1/1	rotate A right through carry	$(A_n)\leftarrow(A_{n+1})$ $(A_7)\leftarrow(C)$, $(C)\leftarrow(A_0)$	n = 0–6	2
	DA A	57	1/1	decimal adjust A			2
	SWAP A	47	1/1	swap nibbles of A	$(A_{4-7})\leftrightarrow(A_{0-3})$		
DATA MOVES	MOV A, Rr	F*	1/1	move register contents to A	(A)←(Rr)	r = 0–7	
	MOV A, @Rr	F0 F1	1/1	move RAM data, addressed by Rr, to A	(A)←((R0)) (A)←((R1))		
	MOV A, #data	23 data	2/2	move immediate data to A	(A)←data		
	MOV Rr, A	A*	1/1	move accumulator contents to register	(Rr)←(A)	r = 0–7	
	MOV @Rr, A	A0 A1	1/1	move accumulator contents to RAM location addressed by Rr	((R0))←(A) ((R1))←(A)		
	MOV Rr, #data	B* data	2/2	move immediate data to Rr	(Rr)←data		
	MOV @Rr, #data	B0 data B1 data	2/2	move immediate data to RAM location addressed by Rr	((R0))←data ((R1))←data		
	XCH A, Rr	2*	1/1	exchange accumulator contents with Rr	$(A)\leftrightarrow(Rr)$	r = 0–7	
	XCH A, @Rr	20 21	1/1	exchange accumulator contents with RAM data addressed by Rr	$(A)\leftrightarrow((R0))$ $(A)\leftrightarrow((R1))$		
	XCHD A, @Rr	30 31	1/1	exchange lower nibbles of A and RAM data addressed by Rr	$(A_{0-3})\leftrightarrow((R0_{0-3}))$ $(A_{0-3})\leftrightarrow((R1_{0-3}))$		
	MOV A, PSW	C7	1/1	move PSW contents to accumulator	(A)←(PSW)		
	MOV PSW, A	D7	1/1	move accumulator bit 3 to PSW_3	$(PSW_3)\leftarrow(A_3)$		3
	MOVP A, @A	A3	1/2	move indirectly addressed data in current page to A	$(PC_{0-7})\leftarrow(A)$, $(A)\leftarrow((PC))$		

Table 2 Instruction set (continued)

FLAGS	CLR C	97	1/1	clear carry bit	(C)←0	2
	CPL C	A7	1/1	complement carry bit	(C)←NOT(C)	2
REGISTER	INC Rr	1*	1/1	increment register by 1	(Rr)←(Rr) + 1 r = 0–7	
	INC @Rr	10 11	1/1	increment RAM data, addressed by Rr, by 1	((R0))←((R0)) + 1 ((R1))←((R1)) + 1	
	DEC Rr	C*	1/1	decrement register by 1	(Rr)←(Rr) – 1 r = 0–7	
	DEC @Rr	C0 C1	1/1	decrement RAM data, addressed by Rr, by 1	((R0))←((R0)) – 1 ((R1))←((R1)) – 1	
BRANCH	JMP addr	• 4 address	2/2	unconditional jump within a 2 K bank	(PC_{8-10})←$addr_{8-10}$ (PC_{0-7})←$addr_{0-7}$ (PC_{11-12})←MBFF 0–1	
	JMPP @A	B3	1/2	indirect jump within a page	(PC_{0-7})←((A))	
	DJNZ Rr, addr	E* address	2/2	decrement Rr by 1 and jump if not zero to addr	(Rr)←(Rr) – 1 r = 0–7 if (Rr) not zero (PC_{0-7})←addr	
	DJNZ @Rr, addr	E0 address	2/2	decrement RAM data, addressed by Rr by 1 and jump if not zero to addr	((R0))←((R0)) – 1 if ((R0)) not zero (PC_{0-7})←addr	
		E1 address			((R1))←((R1)) – 1 if ((R1)) not zero (PC_{0-7})←addr	
	JBb addr	▲ 2 address	2/2	jump to addr if Acc. bit b = 1	if b = 1 : (PC_{0-7})←addr b = 0–7	
	JC addr	F6 address	2/2	jump to addr if C = 1	if C = 1 : (PC_{0-7})←addr	
	JNC addr	E6 address	2/2	jump to addr if C = 0	if C = 0 : (PC_{0-7})←addr	
	JZ addr	C6 address	2/2	jump to addr if A = 0	if A = 0 : (PC_{0-7})←addr	
	JNZ addr	96 address	2/2	jump to addr if A is NOT zero	if A≠0 : (PC_{0-7})←addr	
	JT0 addr	36 address	2/2	jump to addr if T0 = 1	if T0 = 1: (PC_{0-7})←addr	
	JNT0 addr	26 address	2/2	jump to addr if T0 = 0	if T0 = 0: (PC_{0-7})←addr	
	JT1 addr	56 address	2/2	jump to addr if T1 = 1	if T1 = 1: (PC_{0-7})←addr	
	JNT1 addr	46 address	2/2	jump to addr if T1 = 0	if T1 = 0: (PC_{0-7})←addr	
	JTF addr	16 address	2/2	jump to addr if Timer Flag = 1	if TF = 1: (PC_{0-7})←addr	4
	JNTF addr	06 address	2/2	jump to addr if Timer Flag = 0	if TF = 0: (PC_{0-7})←addr	

	mnemonic	opcode (hex.)	bytes/ cycles	description	function	notes
TIMER/EVENT COUNTER	MOV A, T	42	1/1	move timer/event counter contents to accumulator	(A)←(T)	
	MOV T, A	62	1/1	move accumulator contents to timer/event counter	(T)←(A)	
	STRT CNT	45	1/1	start event counter		
	STRT T	55	1/1	start timer		
	STOP TCNT	65	1/1	stop timer/event counter		
	EN TCNTI	25	1/1	enable timer/event counter interrupt		
	DIS TCNTI	35	1/1	disable timer/event counter interrupt		
CONTROL	EN I	05	1/1	enable external interrupt		
	DIS I	15	1/1	disable external interrupt		
	SEL RB0	C5	1/1	select register bank 0	(RBS)←0	5
	SEL RB1	D5	1/1	select register bank 1	(RBS)←1	5
	SEL MB0	E5	1/1	select program memory bank 0	(MBFF0)←0, (MBFF1)←0	
	SEL MB1	F5	1/1	select program memory bank 1	(MBFF0)←1, (MBFF1)←0	
	SEL MB2	A5	1/1	select program memory bank 2	(MBFF0)←0, (MBFF1)←1	
	SEL MB3	B5	1/1	select program memory bank 3	(MBFF0)←1, (MBFF1)←1	
SUBROUTINE	CALL addr	▲4 address	2/2	jump to subroutine	((SP))←(PC), ($PSW_{4, 6, 7}$) (SP)←(SP) + 1 (PC_{8-10})←$addr_{8-10}$ (PC_{0-7})←$addr_{0-7}$ (PC_{11-12})←MBFF 0–1	6
	RET	83	1/2	return from subroutine	(SP)←(SP) −1 (PC)←((SP))	6
	RETR	93	1/2	return from interrupt and restore bits 4, 6, 7 of PSW	(SP)←(SP) −1 ($PSW_{4, 6, 7}$) + (PC)←((SP))	6

Table 2 Instruction set (continued)

PARALLEL INPUT/OUTPUT	IN A, Pp	08 09 0A	1/2	input port p data to accumulator	(A)←(P0) (A)←(P1) (A)←(P2)	7
	OUTL Pp, A	38 39 3A	1/2	output accumulator data to port p	(P0)←(A) (P1)←(A) (P2)←(A)	
	ANL Pp, #data	98 data 99 data 9A data	2/2	AND port p data with immediate data	(P0)←(P0) AND data (P1)←(P1) AND data (P2)←(P2) AND data	
	ORL Pp, #data	88 data 89 data 8A data	2/2	OR port p data with immediate data	(P0)←(P0) OR data (P1)←(P1) OR data (P2)←(P2) OR data	
	OUTL PO,A	90	1/2	Output accumulator data to port ϕ	(P0)←(A)	8
SERIAL INPUT/OUTPUT	EN SI	85	1/1	enable serial I/O interrupt		
	DIS SI	95	1/1	disable serial I/O interrupt		
	NOP	00	1/1	no operation		

Notes to Table 2.

1. PSW CY, AC affected
2. PSW CY affected
3. PSW PS affected
4. Execution of JTF and JNTF instructions resets the Timer Flag (TF).
5. PSW RBS affected
6. PSW SP_0, SP_1, SP_2 affected
7. (A) = 1111 P2.3, P2.2, P2.1, P2.0.
8. Only for software-transfer from the MAB8021.

* : 8, 9, A, B, C, D, E, F
● : 0, 2, 4, 6, 8, A, C, E
▲ : 1, 3, 5, 7, 9, B, D, F

TABLE 3 MAB8422/8442 INSTRUCTION MAP

first hexadecimal character of opcode ↓

second hexadecimal character of opcode →

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	NOP			ADD A, # data	JMP page 0	EN I	JNTF addr	DEC A	IN A,Pp 0	1	2					
1	INC ∂Rr 0	1	JB0 addr	ADDC A, # data	CALL page 0	DIS I	JTF addr	INC A	0	1	2	INC Rr 3	4	5	6	7
2	XCH A, ∂Rr 0	1		MOV A, # data	JMP page 1	EN TCNTI	JNT0 addr	CLR A	0	1	2	XCH A,Rr 3	4	5	6	7
3	XCHD A,∂Rr 0	1	JB1 addr		CALL page 1	DIS TCNTI	JT0 addr	CPL A	OUTL Pp,A 0	1	2					
4	ORL A, ∂Rr 0	1	MOV A, T	ORL A, # data	JMP page 2	STRT CNT	JNT1 addr	SWAP A	0	1	2	ORL A,Rr 3	4	5	6	7
5	ANL A, ∂Rr 0	1	JB2 addr	ANL A, # data	CALL page 2	STRT T	JT1 addr	DA A	0	1	2	ANL A,Rr 3	4	5	6	7
6	ADD A, ∂Rr 0	1	MOV T, A		JMP page 3	STOP TCNT		RRC A	0	1	2	ADD A,Rr 3	4	5	6	7
7	ADDC A, ∂Rr 0	1	JB3 addr		CALL page 3			RR A	0	1	2	ADDC A,Rr 3	4	5	6	7
8				RET	JMP page 4	EN SI			ORL Pp,# data 0	1	2					
9	OUTL PO, A		JB4 addr	RETR	CALL page 4	DIS SI	JNZ addr	CLR C	ANL Pp,# data 0	1	2					
A	MOV ∂Rr, A 0	1		MOVP A, ∂A	JMP page 5	SEL MB2		CPL C	0	1	2	MOV Rr, A 3	4	5	6	7
B	MOV ∂Rr, # data 0	1	JB5 addr	JMPP ∂A	CALL page 5	SEL MB3			0	1	2	MOV Rr, # data 3	4	5	6	7
C	DEC ∂Rr 0	1			JMP page 6	SEL RB0	JZ addr	MOV A, PSW	0	1	2	DEC Rr 3	4	5	6	7
D	XRL A, ∂Rr 0	1	JB6 addr	XRL A, # data	CALL page 6	SEL RB1		MOV PSW, A	0	1	2	XRL A,Rr 3	4	5	6	7
E	DJNZ ∂Rr, addr 0	1			JMP page 7	SEL MB0	JNC addr	RL A	0	1	2	DJNZ Rr, addr 3	4	5	6	7
F	MOV A, ∂Rr 0	1	JB7 addr		CALL page 7	SEL MB1	JC addr	RLC A	0	1	2	MOV A, Rr 3	4	5	6	7

ABSOLUTE MAXIMUM RATINGS

Limiting values in accordance with the Absolute Maximum System (IEC 134)

parameter	symbol	min.	max.	unit
Input voltage on any pin with respect to ground (V_{SS})	V_I	−0,5	+7	V
Total power dissipation	P_{tot}	–	1	W
Input/output current for all pins except port 0	I_I, I_O	–	10	mA
Input/output current for port 0	I_I, I_O	–	20	mA
Storage temperature	T_{stg}	−65	+150	°C
Operating temperature				
standard	T_{amb}	0	+70	°C
extended	T_{amb}	−40	+85	°C
automotive	T_{amb}	−40	+110	°C

DC CHARACTERISTICS

V_{CC} = 5 V (± 10%); V_{SS} = 0 V

parameter	conditions	symbol	min.	max.	unit
Supply current					
MAB8422/42	at 0 °C	I_{CC}	–	70	mA
MAF8422/42	at −40 °C	I_{CC}	–	90	mA
MAF84A22/42	at −40 °C	I_{CC}	–	90	mA
Inputs					
Input voltage LOW (except P1.0/SDA and P1.1/SCL)		V_{IL}	−0,5	0,8	V
Input voltage LOW (P1.0/SDA and P1.1/SCL)		V_{IL1}	−0,5	1,5	V
Input voltage HIGH all inputs except XTAL1, P1.0/SDA and P1.1/SCL		V_{IH}	2,0	V_{CC} + 0,5	V
Input voltage HIGH to XTAL1, P1.0/SDA and P1.1/SCL		V_{IH1}	3,0	V_{CC} + 0,5	V
Outputs					
Output voltage LOW (P0 only)	I_{OL} = 10 mA	V_{OL}		1,0	V
Output voltage LOW (P1.0/SDA and P1.1/SCL)	I_{OL} = 5 mA	V_{OL1}		0,45	V
Output voltage LOW (P0 and P2)	I_{OL} = 1,6 mA	V_{OL2}		0,45	V
Output voltage HIGH (all outputs unless open-drain)	I_{OH} = −50 μA	V_{OH}	2,4		V
Output leakage current	$V_{CC} > V_I > V_{SS}$	± I_{OL}		10	μA

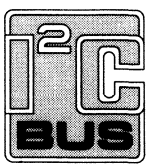
AC CHARACTERISTICS

parameter	symbol	min.	max.	unit
Frequency				
MAB/MAF8422/42	f_{XTAL}	1	6	MHz
MAF84A22/42	f_{XTAL}	1	5	MHz
Cycle time				
MAB/MAF8422/42	t_{CY}	5	30	μs
MAF84A22/42	t_{CY}	6	30	μs

T1 ZERO-CROSS CHARACTERISTICS

T_{amb} = 0 to + 70 °C; V_{CC} = 5 V ± 10%; V_{SS} = 0 V; C_L = 80 pF

parameter	conditions	symbol	min.	max.	unit
Zero-cross detection input (T1) peak-to-peak	AC coupled, C = 1 μF	$V_{ZX(p-p)}$	1	3	V
Zero-cross accuracy	50 Hz sine wave	A_{ZX}	–	± 135	mV
Zero-cross detection input frequency (T1)		F_{ZX}	0,05	1	kHz



Purchase of Philips' I²C components conveys a license under the Philips' I²C patent to use the components in the I²C-system provided the system conforms to the I²C specifications defined by Philips.

SUPERSEDES DATA OF AUGUST 1987

SINGLE-CHIP 8-BIT MICROCONTROLLER

DESCRIPTION

The PCB80C51 family of single-chip 8-bit microcontrollers is manufactured in an advanced CMOS process. The family consists of the following members:

- PCB80C51BH-3: 4 K bytes mask-programmable ROM, 128 bytes RAM
- PCB80C31BH-3: ROM-less version of the PCB80C51BH-3

In the following text, the generic term "PCB80C51BH-3" is used to refer to both family members.

The device provides hardware features, architectural enhancements and new instructions to function as a controller for applications requiring up to 64 K bytes of program memory and/or up to 64 K bytes of data memory.

The PCB80C51BH-3 contains a non-volatile 4 K x 8 read-only program memory; a volatile 128 x 8 read/write data memory; 32 I/O lines; two 16-bit timer/event counters; a five-source, two-priority level, nested interrupt structure; a serial I/O port for either multi-processor communications, I/O expansion, or full duplex UART; and on-chip oscillator and timing circuits. For systems that require extra capability, the PCB80C51BH-3 can be expanded using standard TTL compatible memories and logic.

The PCB80C51BH-3 has two software selectable modes of reduced activity for further power reduction - Idle and Power-down.
The Idle mode freezes the CPU while allowing the RAM, timers, serial port and interrupt system to continue functioning.
The Power-down mode saves the RAM contents but freezes the oscillator causing all other chip functions to be inoperative.

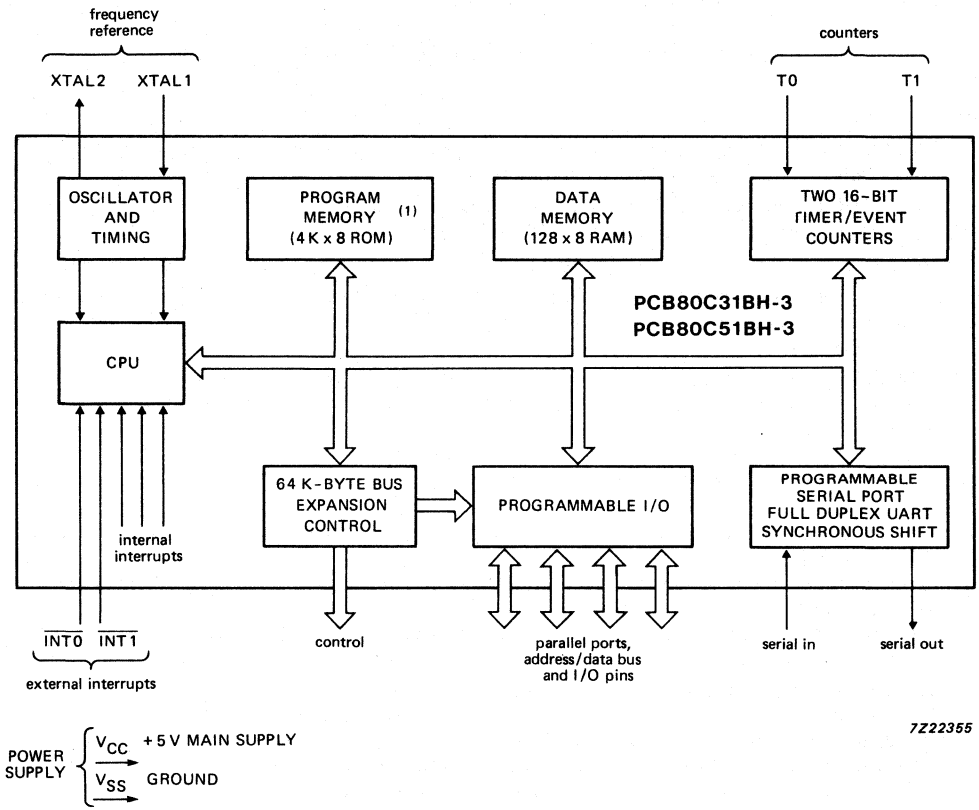
The device also functions as an arithmetic processor having facilities for both binary and BCD arithmetic plus bit-handling capabilities. The instruction set consists of over 100 instructions: 49 one-byte, 46 two-byte and 16 three-byte. With a 12 MHz crystal, 58% of the instructions are executed in 1 μs and 40% in 2 μs. Multiply and divide instructions require 4 μs. Multiply, divide, subtract and compare are among the many instructions included in the instruction set.

Features

- 4 K x 8 ROM (80C51BH-3 only), 128 x 8 RAM
- Four 8-bit ports, 32 I/O lines
- Two 16-bit timer/event counters
- Full-duplex serial port
- External memory expandable to 128 K, external ROM up to 64 K and/or external RAM up to 64 K
- Boolean processing
- 218 bit-addressable locations
- On-chip oscillator
- Five-source interrupt structure with two priority levels
- With a 12 MHz clock, 58% of the instructions execute in 1 μs; multiply and divide instructions execute in 4 μs; all other instructions execute in 2 μs
- Enhanced architecture with:
  non-page-oriented-instructions
  direct addressing
  four 8-byte + 1-byte register blanks
  stack depth up to 128-bytes
  multiply, divide, subtract and compare instructions
- PCB80C51/C31BH-3
  XTAL frequency range: 1,2 to 16 MHz
  temperature range: 0 °C to + 70 °C
  PCF80C51/C31BH-3
  XTAL frequency range: 1,2 to 12 MHz
  temperature range: –40 °C to + 85 °C
  PCA80C51/C31BH-3
  XTAL frequency range: 1,2 to 12 MHz
  temperature range: –40 °C to +125 °C

PACKAGE OUTLINES

PCB/PCF80C51/C31BH-3P, PCA80C51/C31BH-3P: 40 lead DIL; plastic (SOT-129).
PCB/PCF80C51/C31BH-3WP, PCA80C51/C31BH-3WP: 44-lead PLCC; plastic leaded chip-carrier (SOT187 pedestal or SOT187AA pocket version depending on source, versions are interchangeable.



(1) PCB80C51BH-3 only.

Fig. 1 Block diagram.

DEVELOPMENT DATA

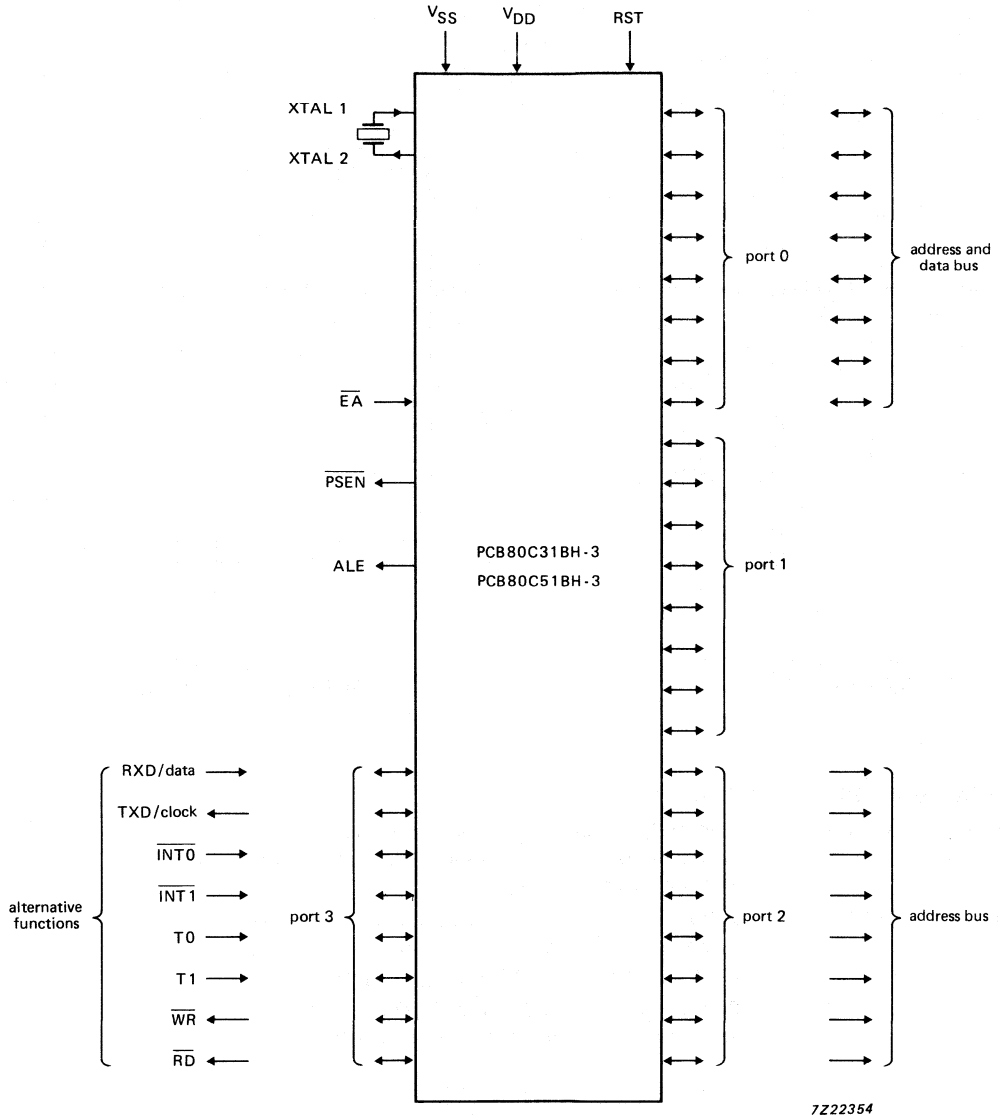


Fig. 2 Functional diagram.

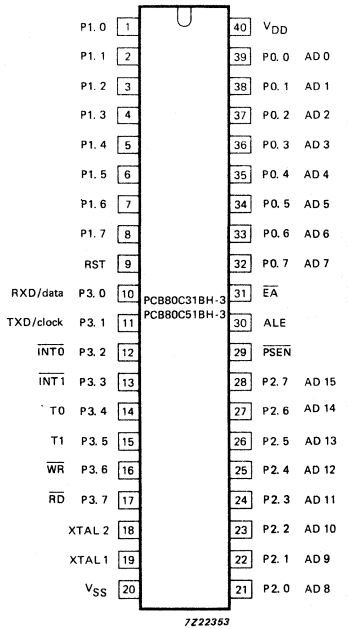


Fig. 3a Pinning diagram for PCB80C51/C31BH-3P.

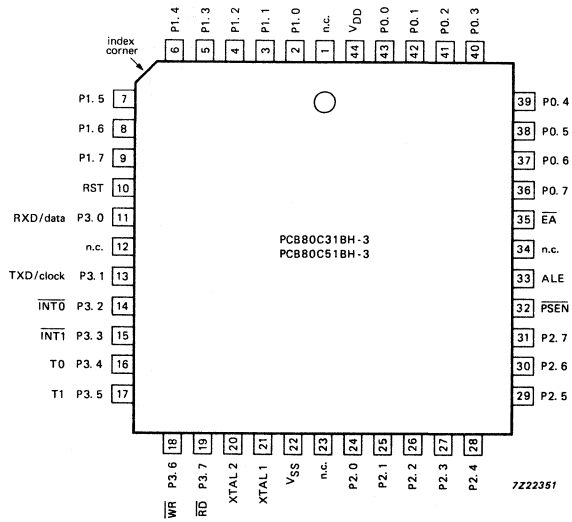


Fig. 3b Pinning diagram for PCB80C51/C31BH-3WP.

PINNING (PCB80C51/C31BH-3P)

1–8	P1.0–P1.7	**Port 1:** 8-bit quasi-bidirectional I/O port. Port 1 can drive one TTL (= 4 LSTTL) input. It can drive CMOS inputs without external pull-ups.
9	RST	**RESET:** a high level on this pin for two machine cycles with the oscillator is running resets the device. An internal pull-down permits Power-On reset using only a capacitor connected to V_{DD}.
10–17	P3.0–P3.7	**Port 3:** 8-bit quasi-bidirectional I/O port with internal pull-ups. It also serves the following alternative functions:

Port pin	*Alternative function*
P3.0	**RXD/data:** serial port receiver data input (asynchronous) or data input/output (synchronous)
P3.1	**TXD/clock:** serial port transmitter data output (asynchronous) or clock output (synchronous)
P3.2	**$\overline{\text{INT0}}$:** external interrupt 0 or gate control input for timer/event counter 0
P3.3	**$\overline{\text{INT1}}$:** external interrupt 1 or gate control input for timer/event counter 1

	P3.4	**T0:** external input for timer/event counter 0
	P3.5	**T1:** external input for timer/event counter 1
	P3.6	**$\overline{WR}$:** external data memory write strobe
	P3.7	**$\overline{RD}$:** external data memory read strobe
		An alternative function can only be activated if the corresponding bit latch in the port SFR contains a logic 1. Port 3 can drive one TTL input. It can drive CMOS inputs without external pull-ups.
18	XTAL 2	**Crystal input 2:** output of the inverting amplifier that forms the oscillator. Left open-circuit when an external oscillator is used (see figures 9 and 10).
19	XTAL 1	**Crystal input 1:** input to the inverting amplifier that forms the oscillator, and input to the internal clock generator. Receives the external oscillator signal when an external oscillator is used (see figures 9 and 10).
20	V_{SS}	**Ground:** circuit ground potential.
21–28	P2.0–P2.7	**Port 2:** 8-bit quasi-bidirectional I/O port with internal pull-ups. Port 2 can drive one TTL input. It can drive CMOS inputs without external pull-ups. During accesses to external memories (RAM/ROM) that use 16-bit addresses (MOVX@DPTR), Port 2 emits the high-order address byte. When used for an external RAM with an 8-bit address (MOVX@Ri), Port 2 emits the contents of the P2 special function register.
29	$\overline{PSEN}$	**Program Store Enable output:** read strobe to the external program memory. It is activated twice each machine cycle during fetches from external program memory. When executing out of external program memory, two activations of $\overline{PSEN}$ are skipped during each access to external data memory. $\overline{PSEN}$ is not activated (remains HIGH) during fetches from internal program memory. $\overline{PSEN}$ can drive 8 LSTTL inputs. It can drive CMOS inputs without an external pull-up.
30	ALE	**Address Latch Enable output:** latches the low byte of the addres during accesses to external memory in normal operation. It is activated every six oscillator periods except during an external data memory access. ALE can drive up to 8 LSTTL inputs. It can drive CMOS inputs without an external pull-up.
31	$\overline{EA}$	**External Access input:** When $\overline{EA}$ is held at a TTL high level the CPU executes out of the internal program memory (ROM), provided the program counter is less than 4096. When $\overline{EA}$ is held at a TTL low level, the CPU executes out of external program memory. $\overline{EA}$ is not allowed to float.
32–39	P0.7–P0.0	**Port 0:** 8-bit open drain bidirectional I/O port. It is also the multiplexed low-order address and data bus during accesses to external memory (during these accesses it activates internal pull-ups). Port 0 can drive up to eight TTL inputs.
40	V_{DD}	**Power supply:** + 5 V power supply pin during normal operation, Idle mode and Power-down mode.

DEVELOPMENT DATA

FUNCTIONAL DESCRIPTION

General

The PCB80C51BH-3 is a stand-alone high-performance microcontroller designed for use in real-time applications such as instrumentation, industrial control and intelligent computer peripherals.

The device provides hardware features, architectural enhancements and new instructions to function as a controller for applications requiring up to 64 K bytes of program memory and/or up to 64 K bytes of data memory.

The PCB80C51BH-3 is a control-oriented CPU without on-chip program memory. It can address 64 K bytes of external program memory in addition to 64 K bytes of external data memory. The PCB80C51BH-3 is a PCB80C31BH-3 with 4 K bytes of on-chip mask programmable ROM. For systems requiring extra capability, the PCB80C51BH-3 can be expanded using standard memories and pheripherals.

The two pin-compatible versions of this component reduce development problems to a minimum and provide maximum flexibility. The PCB80C51BH-3 is designed for low-cost, high volume production and the PCB80C31BH-3 for applications requiring the flexibility of external program memory which can be easily modified and updated in the field.

The PCB80C51BH-3 contains a non-volatile 4 K x 8 read-only program memory; a volatile 128 x 8 read/write data memory; 32 I/O lines; two 16-bit timer/event counters; five-source, two-priority-level, nested interrupt structure; a serial I/O port for either multi-processor communications, I/O expansion, or full duplex UART; and on-chip oscillator and timing circuits.

Memory organization

The central processing unit (CPU) manipulates operands in three memory spaces; these are the 64 K byte external data memory, 256-byte internal data memory and the 64 K byte internal and external program memory. The internal data memory address space is sub-divided into the 256-byte internal data RAM and 128-byte special function register (SFR) address spaces, as shown in Fig. 4. Figure 5 shows the special function register memory map. Internal RAM locations 0-127 are directly and indirectly addressable. The special function register locations 128-255 are only directly addressable.

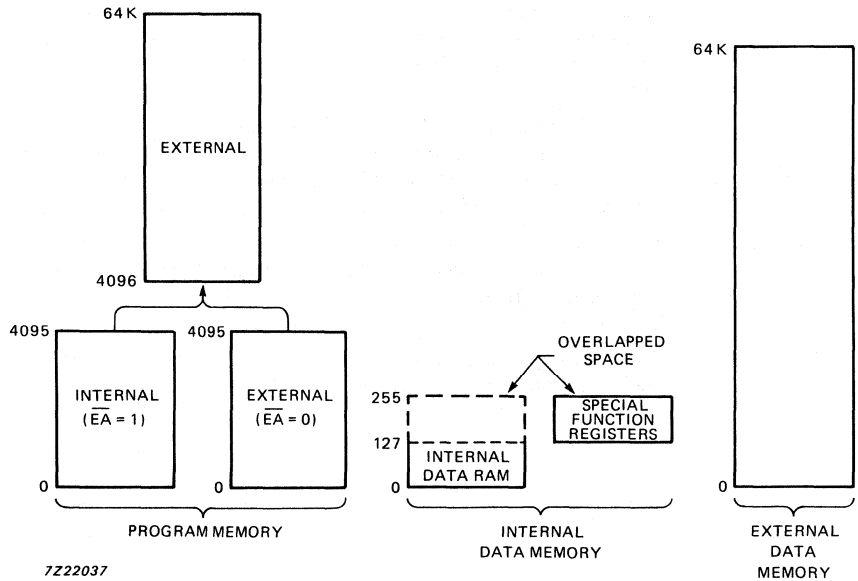
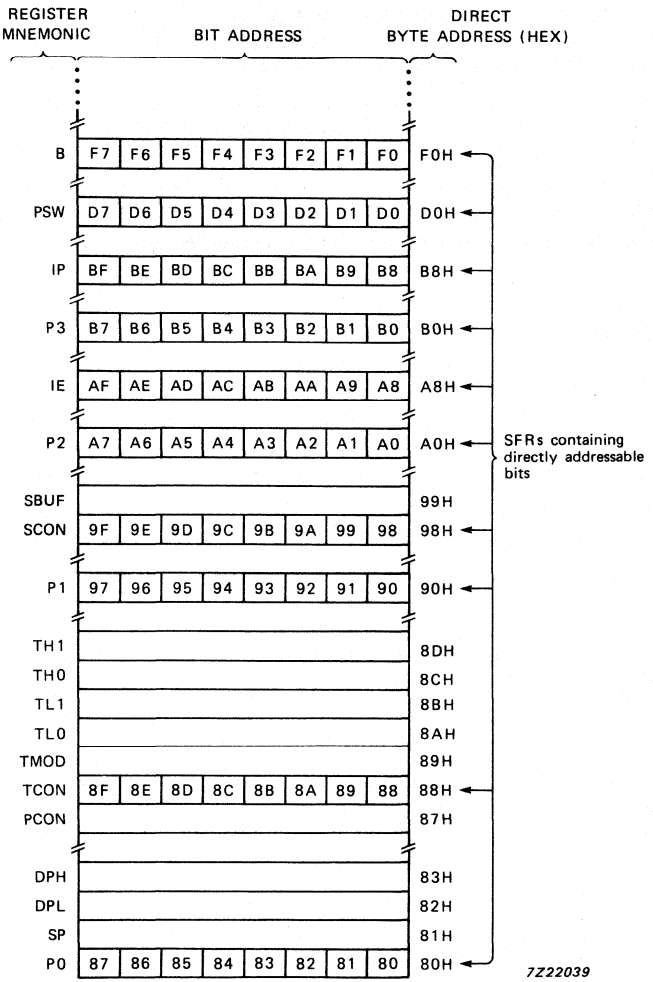


Fig. 4 Memory map.



# denotes read only registers

Fig. 5 Special function register memory map.

The internal data RAM contains four register banks (each with eight registers), 128 addressable bits, and the stack. The stack depth is limited by the available internal data RAM and its location is determined by the 8-bit stack pointer. All registers except the program counter and the four 8-register banks reside in the special function register address space. These memory mapped registers include arithmetic registers, pointers, I/O ports, interrupt system registers, timers and serial port. There are 128 addressable bite locations in the SFR address space.

The PCB80C51BH-3 contains 128 bytes of internal data RAM and 20 special function registers. It provides a non-paged program memory address space to accommodate relocatable code. Conditional branches are performed relative to the program counter. The register-indirect jump permits branching relative to a 16-bit base register with an offset provided by an 8-bit index register. 16-bit jumps and calls permit branching to any location in the contiguous 64 K program memory address space.

The PCB80C51BH-3 has five methods for addressing source operands:

- Register
- Direct
- Register-Indirect
- Immediate
- Base-Register-plus Index-Register-Indirect

The first three methods can be used for addressing destination operands. Most instructions have a "destination/source" field that specifies the data type, addressing methods and operands involved. For operations other than moves, the destination operand is also a source operand.

Access to memory addressing is as follows:

- Registers in the four 8-register banks through Register, Direct, or Register-Indirect
- 128 bytes of internal data RAM through Direct or Register-Indirect
- Special function registers through Direct
- External data memory through Register-Indirect
- Program memory look-up tables through Base-Register-plus Index-Register-Indirect

The PCB80C51BH-3 is classified as an 8-bit device since the internal ROM, RAM, special function registers (SFRs), Arithmetic Logic Unit (ALU) and external data bus are each 8-bits wide. It performs operations on bit, nibble, byte and double-byte data types.

Facilities are available for byte transfer, logic and integer arithmetic operations. Data transfer, logic and conditional branch operations can be performed directly on Boolean variables to provide excellent bit handling.

I/O facilities

The PCB80C51BH-3 has 32 individually addressable I/O lines organized as four parallel bit addressable ports. Ports 0, 1, 2 and 3 perform the following alternative functions:

- Port 0: provides the multiplexed low-order address and data bus used for expanding the PCB80C51BH-3 with standard memories and pheripherals.
- Port 2: provides the high-order address bus when expanding the PCB80C51BH-3 with external program memory and/or more than 256 bytes of external data memory.
- Port 3: pins can be configured individually to provide:
  - external interrupt request inputs
  - counter inputs
  - serial port receiver input and transmitter output
  - control signals to READ and WRITE to external data memory

The generation or use of a Port 3 pin as an alternative function is carried out automatically by the PCB80C51BH-3 provided the associated SFR bit is set HIGH.

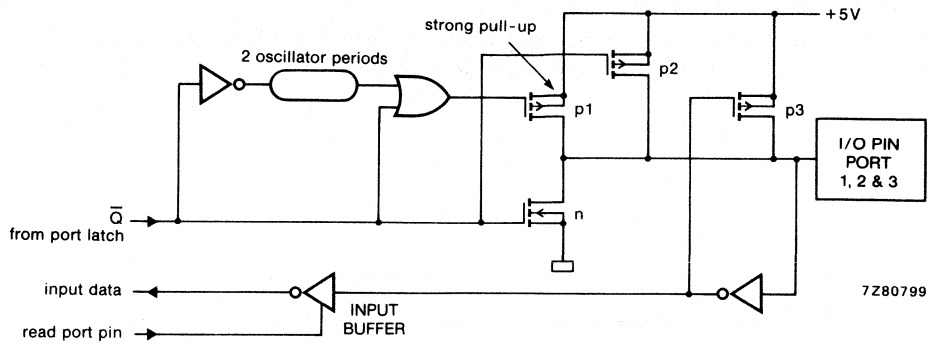


Fig. 6 I/O buffers in the PCB80C51BH-3 (Ports 1, 2 and 3).

Timer/event counters

The PCB80C51BH-3 contains two 16-bit timer/event counters: Timer 0 and Timer 1. Timer 0 and Timer 1 may be programmed to carry out the following functions:

- Measure time intervals and pulse durations
- Count events
- Generate interrupt requests

Each timer/event counter can be individually programmed to operate in three modes:

- Mode 0: 8-bit timer or 8-bit counter each with divide by 32 prescaler
- Mode 1: 16-bit time-interval or event counter
- Mode 2: 8-bit time-interval or event counter with automatic reload upon overflow

Counter 0 can be programmed to operate in an additional mode as follows:

- Mode 3: one 8-bit time-interval or event counter and one 8-bit time-interval counter

When Timer 0 is in Mode 3, Timer 1 can be programmed to operate in Modes 0, 1 or 2 but cannot set an interrupt request flag or generate an interrupt. However the overflow from Timer 1 can be used to pulse the serial port transmission-rate generator.

The frequency handling range of these counters with a 12 MHz crystal is as follows:

- In the timer function, the timer is incremented at a frequency of 1 MHz — a division by 12 of the oscillator frequency
- 0 Hz to an upper limit of 0,5 MHz when programmed for external inputs

Both internal and external inputs can be gated to the counter by a second external source for directly measuring pulse durations.

The counters are started and stopped under software control. Each one sets its interrupt request flag when it overflows from all logic 1s to all logic 0s (or automatic reload value) with the exception of mode 3 as described above.

On-chip peripheral functions

In addition to the CPU and memories, an interrupt system, extensive I/O facilities and several peripheral functions are integrated on-chip to relieve the CPU of repetitious, complicated or time critical tasks and to permit stringent real-time control of external system interfaces. The I/O facilities include the I/O pins, parallel ports, bidirectional address/data bus and the serial port for I/O expansion. The CPU peripheral functions integrated on-chip are the two 16-bit timer/event counters and the serial port.

Idle and Power-down operation (see Fig. 7)

The Power-down operation freezes the oscillator. Idle mode operation allows the interrupt, serial port and timer blocks to continue to function while the CPU is halted.

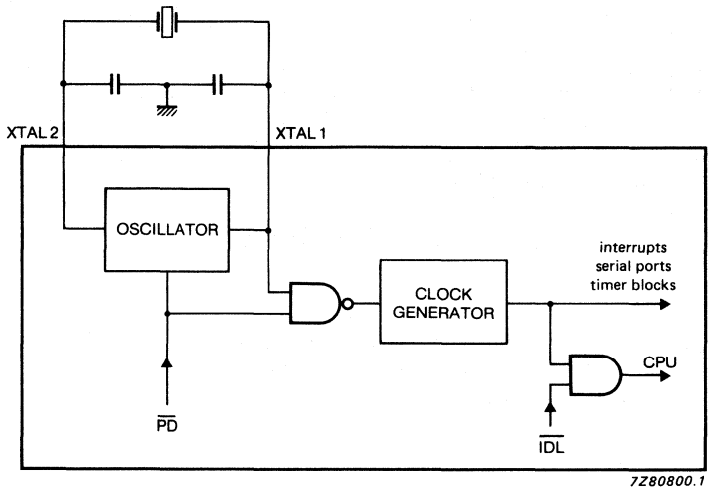


Fig. 7 Internal Idle and Power-down clock configuration.

Power control register (PCON)

These special modes are activated by software via the special function register PCON. Its hardware address is 87H. PCON is not bit addressable.

	7	6	5	4	3	2	1	0
PCON (87H)	SMOD	–	–	–	GF1	GF0	PD	IDL

Bit	Symbol	Function
PCON.7	SMOD	Double Baud rate bit. When set to logic 1 the baud rate is doubled when the serial port is being used in modes 1, 2 or 3
PCON.6	–	(reserved)
PCON.5	–	(reserved)
PCON.4	–	(reserved)
PCON.3	GF1	General-purpose flag bit
PCON.2	GF0	General-purpose flag bit
PCON.1	PD	Power-down bit. Setting this bit activates Power-down mode. It can only be set if input $\overline{EW}$ is high
PCON.0	IDL	Idle mode bit. Setting this bit activates the Idle mode

If logic 1s are written to PD and IDL at the same time, PD takes precedence. The reset value of PCON is (0XXX0000).

Idle mode

The instruction that sets PCON.0 is the last instruction executed in the normal operating mode before Idle mode is activated. Once in the Idle mode, the CPU status is preserved in its entirety: the Stack Pointer, Program Counter, Program Status Word, Accumulator, RAM and all other registers maintain their data during Idle mode. The status of the external pins during Idle mode is shown in Table 1.

There are two ways to terminate the Idle mode:

Activation of any enabled interrupt will cause PCON.0 to be cleared by hardware terminating Idle mode. The interrupt is serviced, and following the return from interrupt instruction (RETI), the next instruction to be executed will be the one which follows the instruction that wrote a logic 1 to PCON.0. The flag bits GF0 and GF1 may be used to determine whether the interrupt was received during normal execution or during the Idle mode. For example, the instruction that writes the PCON.0 can also set or clear one or both flag bits. When Idle mode is terminated by an interrupt, the service routine can examine the status of the flag bits.

The second way of terminating the Idle mode is with an external hardware reset. Since the oscillator is still running, the hardware reset is required to be active for two machine cycles (24 oscillator periods) to complete the reset operation.

Power-down mode

The instruction that sets PCON.1 is the last executed prior to going into the Power-down mode. Once in Power-down mode, the oscillator is stopped. Only the contents of the on-chip RAM are preserved. The special function registers are not saved. A hardware reset is the only way of exiting the Power-down mode.

In the Power-down mode, V_{DD} may be reduced to minimize circuit power consumption. The voltage must not be reduced until the Power-down mode is entered and must be restored before the hardware reset is applied which will free the oscillator. Reset should not be released until the oscillator has restarted and stabilized.

The status of the external pins during Power-down mode is shown in Table 1. If the Power-down mode is activated while in external program memory, the port data that is held in the special function register P2 is restored to Port 2. If the data is a logic 1, the port pin is held HIGH during the Power-down mode by the strong pull-up transistor p1 (see Fig. 6).

Table 1 Status of the external pins during Idle and Power-down modes

mode	memory	ALE	$\overline{PSEN}$	Port 0	Port 1	Port 2	Port 3
Idle	internal	1	1	port data	port data	port data	port data
Idle	external	1	1	floating	port data	address	port data
Power-down	internal	0	0	port data	port data	port data	port data
Power-down	external	0	0	floating	port data	port data	port data

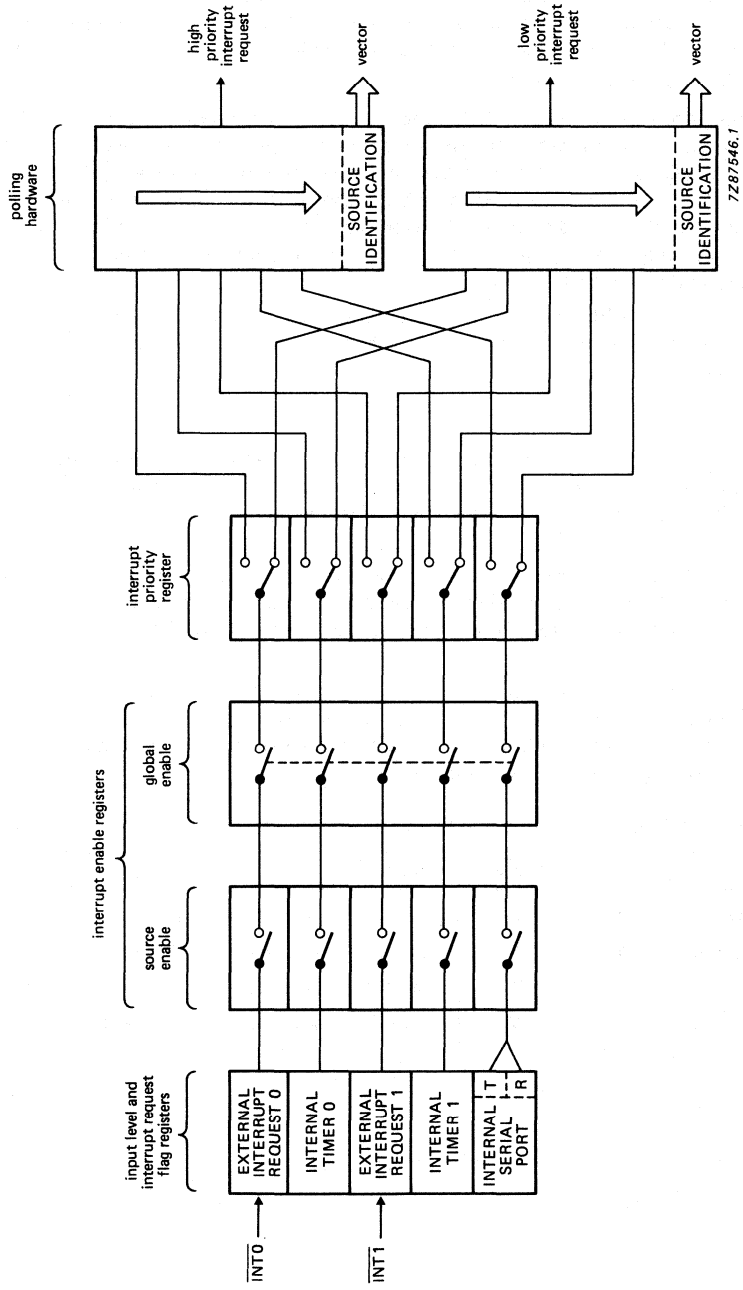


Fig. 8 Interrupt system.

Interrupt system (see Fig. 8)

External events and the real-time-driven on-chip peripherals require service by the CPU asynchronous to the execution of any particular section of code. To tie the asynchronous activities of these functions to normal program execution a multiple-source, two-priority-level, nested interrupt system is provided. Interrupt response latency is from 3 μs to 7 μs when using a 12 MHz crystal.

The PCB80C51BH-3 acknowledges interrupt requests from five sources as follows:

- $\overline{\text{INT0}}$ and $\overline{\text{INT1}}$: externally via pins 12 and 13 respectively
- Timer 0 and Timer 1: from the two internal counters
- Serial Port: from the internal serial I/O port

Each interrupt vectors to a separate location in program memory for its service program.

Each source can be individually enabled or disabled and can be programmed to a high or low priority level. Also all enabled sources can be globally disabled or enabled. Both external interrupts can be programmed to be level-activated or transition-activated; an active LOW level allows "wire-ORing" of several interrupt sources to the input pin.

Oscillator circuitry

The oscillator circuitry of the PCB80C51BH-3 is a single-stage inverting amplifier in a Pierce oscillator configuration. The circuitry between XTAL 1 and XTAL 2 is basically an inverter biased to the transfer point. Either a crystal or ceramic resonator can be used as the feedback element to complete the oscillator circuitry. Both are operated in parallel resonance. XTAL 1 is the high gain amplifier input and XTAL 2 is the output (see Fig. 9).

To drive the PCB80C51BH-3 externally, XTAL 1 is driven from an external source and XTAL 2 left open-circuit (see Fig. 10).

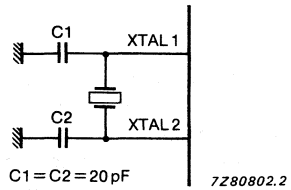


Fig. 9 PCB80C51BH-3 oscillator circuit.

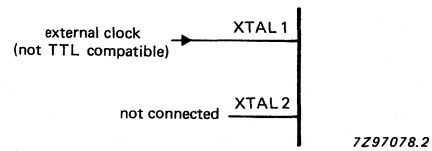


Fig. 10 Driving the PCB80C51BH-3 from an external source.

Reset circuitry

The reset circuitry for the PCB80C51BH-3 is connected to the reset pin, RST, as shown in Fig. 11. A Schmitt trigger is used at the input for noise rejection. The output of the Schmitt trigger is sampled by the reset circuitry every machine cycle.

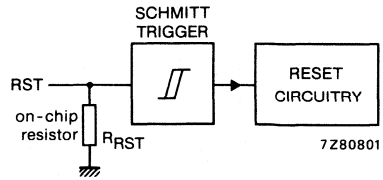


Fig. 11 Reset configuration at RST.

A reset is accomplished by holding the RST pin HIGH for at least two machine cycles (24 oscillator periods), while the oscillator is running. The CPU responds by executing an internal reset. During reset, ALE and $\overline{\text{PSEN}}$ output a HIGH level. In order to perform a correct reset, this level must not be affected by external elements.

The internal reset is executed during the second cycle in which RST is HIGH and is repeated every cycle until RST goes LOW. It leaves the internal registers as follows:

Register	Content
PC	000H
ACC	00H
B	00H
PSW	00H
SP	07H
DPTR	0000H
P0 - P3	0FFH
IP	(XX000000)
IE	(0X000000)
TMOD	00H
TCON	00H
TH0	00H
TL0	00H
TH1	00H
TL1	00H
SCON	00H
SBUF	Intermediate
PCON	(0XXX0000)

Where: x = indetermine.

The internal RAM is not affected by reset. When V_{DD} is turned on, the RAM content is indeterminate.

Power-on reset (see Fig. 12)

When V_{DD} is turned on and provided its rise-time does not exceed 10 ms, an automatic reset can be obtained by connecting the RST pin to V_{DD} via a 2,2 μF capacitor. When the power is switched on, the voltage on the RST pin is equal to V_{DD} minus the capacitor voltage, and decreases from V_{DD} as the capacitor charges through the internal resistor (R_{RST}) to ground. The larger the capacitor, the more slowly V_{RST} decreases. V_{RST} must remain above the lower threshold of the Schmitt trigger long enough to effect a complete reset. The time required is the oscillator start-up, plus 2 machine cycles.

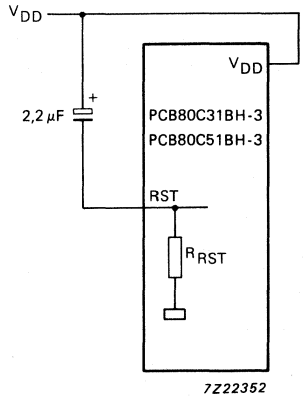


Fig. 12 Power-on reset.

INSTRUCTION SET

The PCB80C51BH-3 uses a powerful instruction set to allow expansion of on-chip CPU peripherals and to optimize byte efficiency and execution speed. Reassigned opcodes add new high-power operations and permit new addressing modes to make old operations more orthogonal when compared to the 8048 family.

The instruction set consists of 49 single-byte, 46 two-byte and 16 three-byte instructions. When using a 16 MHz oscillator, 64 instructions execute in 750 ns and 45 instructions execute in 1,5 μs. Multiply and divide instructions execute in 3 μs.

Table 2 Instruction set description.

mnemonic		description	bytes/ cycles	opcode (hex.)
Arithmetic operation				
ADD	A,Rr	Add register to A	1 1	2*
ADD	A,direct	Add direct byte to A	2 1	25
ADD	A,@Ri	Add indirect RAM to A	1 1	26, 27
ADD	A,#data	Add immediate data to A	2 1	24
ADDC	A,Rr	Add register to A with carry flag	1 1	3*
ADDC	A,direct	Add direct byte to A with carry flag	2 1	35
ADDC	A,@Ri	Add indirect RAM to A with carry flag	1 1	36, 37
ADDC	A,#data	Add immediate data to A with carry flag	2 1	34
SUBB	A,Rr	Subtract register from A with borrow	1 1	9*
SUBB	A,direct	Subtract direct byte from A with borrow	2 1	95
SUBB	A,@Ri	Subtract indirect RAM from A with borrow	1 1	96, 97
SUBB	A,#data	Subtract immediate data from A with borrow	2 1	94
INC	A	Increment A	1 1	04
INC	Rr	Increment register	1 1	0*
INC	direct	Increment direct byte	2 1	05
INC	@Ri	Increment indirect RAM	1 1	06, 07
DEC	A	Decrement A	1 1	14
DEC	Rr	Decrement register	1 1	1*
DEC	direct	Decrement direct byte	2 1	15
DEC	@Ri	Decrement indirect RAM	1 1	16, 17
INC	DPTR	Increment data pointer	1 2	A3
MUL	AB	Multiply A & B	1 4	A4
DIV	AB	Divide A by B	1 4	84
DA	A	Decimal adjust A	1 1	D4

INSTRUCTION SET (continued)

mnemonic		description	bytes/ cycles	opcode (hex.)
Logic operations				
ANL	A,Rr	AND register to A	1 1	5*
ANL	A,direct	AND direct byte to A	2 1	55
ANL	A,@Ri	AND indirect RAM to A	1 1	56, 57
ANL	A,#data	AND immediate data to A	2 1	54
ANL	direct,A	AND A to direct byte	2 1	52
ANL	direct,#data	AND immediate data to direct byte	3 2	53
ORL	A,Rr	OR register to A	1 1	4*
ORL	A,direct	OR direct byte to A	2 1	45
ORL	A,@Ri	OR indirect RAM to A	1 1	46, 47
ORL	A,#data	OR immediate data to A	2 1	44
ORL	direct,A	OR A to direct byte	2 1	42
ORL	direct,#data	OR immediate data to direct byte	3 2	43
XRL	A,Rr	Exclusive-OR register to A	1 1	6*
XRL	A,direct	Exclusive-OR direct byte to A	2 1	65
XRL	A,@Ri	Exclusive-OR indirect RAM to A	1 1	66, 67
XRL	A,#data	Exclusive-OR immediate data to A	2 1	64
XRL	direct, A	Exclusive-OR A to direct byte	2 1	62
XRL	direct,#data	Exclusive-OR immediate data to direct byte	3 2	63
CLR	A	Clear A	1 1	E4
CPL	A	Complement A	1 1	F4
RL	A	Rotate A left	1 1	23
RLC	A	Rotate A left through the carry flag	1 1	33
RR	A	Rotate A right	1 1	03
RRC	A	Rotate A right through the carry flag	1 1	13
SWAP	A	Swap nibbles within A	1 1	C4

mnemonic		description	bytes/ cycles	opcode (hex.)
Data transfer				
MOV	A,Rr	Move register to A	1 1	E*
MOV	A,direct **	Move direct byte to A	2 1	E5
MOV	A,@Ri	Move indirect RAM to A	1 1	E6, E7
MOV	A,#data	Move immediate data to A	2 1	74
MOV	Rr,A	Move A to register	1 1	F*
MOV	Rr,direct	Move direct byte to register	2 2	A*
MOV	Rr,#data	Move immediate data to register	2 1	7*
MOV	direct,A	Move A to direct byte	2 1	F5
MOV	direct,Rr	Move register to direct byte	2 2	8*
MOV	direct,direct	Move direct byte to direct byte	3 2	85
MOV	direct,@Ri	Move indirect RAM to direct byte	2 2	86, 87
MOV	direct,#data	Move immediate data to direct byte	3 2	75
MOV	@Ri,A	Move A to indirect RAM	1 1	F6, F7
MOV	@Ri,direct	Move direct byte to indirect RAM	2 2	A6, A7
MOV	@Ri,#data	Move immediate data to indirect RAM	2 1	76, 77
MOV	DPTR,#data16	Load data pointer with a 16-bit constant	3 2	90
MOVC	A,@A+DPTR	Move code byte relative to DPTR to A	1 2	93
MOVC	A,@A+PC	Move code byte relative to PC to A	1 2	83
MOVX	A,@Ri	Move external RAM (8-bit address) to A	1 2	E2, E3
MOVX	A,@DPTR	Move external RAM (16-bit address) to A	1 2	E0
MOVX	@Ri,A	Move A to external RAM (8-bit address)	1 2	F2, F3
MOVX	@DPTR,A	Move A to external RAM (16-bit address)	1 2	F0
PUSH	direct	Push direct byte onto stack	2 2	C0
POP	direct	Pop direct byte from stack	2 2	D0
XCH	A,Rr	Exchange register with A	1 1	C*
XCH	A,direct	Exchange direct byte with A	2 1	C5
XCH	A,@Ri	Exchange indirect RAM with A	1 1	C6, C7
XCHD	A,@Ri	Exchange LOW-order digit indirect RAM with A	1 1	D6, D7

** MOV A, ACC is not a valid instruction.

INSTRUCTION SET (continued)

mnemonic		description	bytes/ cycles	opcode (hex.)
Boolean variable manipulation				
CLR	C	Clear carry flag	1 1	C3
CLR	bit	Clear direct bit	2 1	C2
SETB	C	Set carry flag	1 1	D3
SETB	bit	Set direct bit	2 1	D2
CPL	C	Complement carry flag	1 1	B3
CPL	bit	Complement direct bit	2 1	B2
ANL	C,bit	AND direct bit to carry flag	2 2	82
ANL	C,/bit	AND complement of direct bit to carry flag	2 2	B0
ORL	C,bit	OR direct bit to carry flag	2 2	72
ORL	C,/bit	OR complement of direct bit to carry flag	2 2	A0
MOV	C,bit	Move direct bit to carry flag	2 1	A2
MOV	bit,C	Move carry flag to direct bit	2 2	92
Program and machine control				
ACALL	addr11	Absolute subroutine call	2 2	●1addr
LCALL	addr16	Long subroutine call	3 2	12
RET		Return from subroutine	1 2	22
RETI		Return from interrupt	1 2	32
AJMP	addr11	Absolute jump	2 2	▲1addr
LJMP	addr16	Long jump	3 2	02
SJMP	rel	Short jump (relative address)	2 2	80
JMP	@A+DPTR	Jump indirect relative to the DPTR	1 2	73
JZ	rel	Jump if A is zero	2 2	60
JNZ	rel	Jump if A is not zero	2 2	70
JC	rel	Jump if carry flag is set	2 2	40
JNC	rel	Jump if carry flag is not set	2 2	50
JB	bit,rel	Jump if direct bit is set	3 2	20
JNB	bit,rel	Jump if direct bit is not set	3 2	30
JBC	bit,rel	Jump if direct bit is set and clear bit	3 2	10
CJNE	A,direct,rel	Compare direct to A and jump if not equal	3 2	B5
CJNE	A,#data,rel	Compare immediate to A and jump if not equal	3 2	B4
CJNE	Rr,#data,rel	Compare immed. to reg. and jump if not equal	3 2	B*
CJNE	@Ri,#data,rel	Compare immed. to ind. and jump if not equal	3 2	B6, B7
DJNZ	Rr,rel	Decrement register and jump if not zero	2 2	D*
DJNZ	direct,rel	Decrement direct and jump if not zero	3 2	D5
NOP		No operation	1 1	00

Notes to Table 2

Data addressing modes

Rr	Working register R0-R7.
direct	128 internal RAM locations and any special function register (SFR).
@Ri	Indirect internal RAM location addressed by register R0 or R1 of the actual register bank.
#data	8-bit constant included in instruction.
#data16	16-bit constant included as bytes 2 and 3 of instruction.
bit	direct addressed bit in internal RAM or SFR.
addr16	16-bit destination address. Used by LCALL and LJMP. The branch will be anywhere within the 64 K-byte program memory address space.
addr11	11-bit destination address. Used by ACALL and AJMP. The branch will be within the same 2 K-byte page of program memory as the first byte of the following instruction.
rel	Signed (two's complement) 8-bit offset byte. Used by SJMP and all conditional jumps. Range is −128 to + 127 bytes relative to first byte of the following instruction.

Hexadecimal opcode cross-reference to Table 3

★ : 8, 9, A, B, C, D, E, F.

● : 11, 31, 51, 71, 91, B1, D1, F1.

▲ : 01, 21, 41, 61, 81, A1, C1, E1.

Table 3 Instruction map

first hexadecimal character of opcode ↓ — second hexadecimal character of opcode →

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	NOP	AJMP addr11	LJMP addr16	RR A	INC A	INC dir	INC @Ri		INC Rr							
							0	1	0	1	2	3	4	5	6	7
1	JBC bit,rel	ACALL addr11	LCALL addr16	RRC A	DEC A	DEC dir	DEC @Ri		DEC Rr							
							0	1	0	1	2	3	4	5	6	7
2	JB bit,rel	AJMP addr11	RET	RL A	ADD A,#data	ADD A,dir	ADD A,@Ri		ADD A,Rr							
							0	1	0	1	2	3	4	5	6	7
3	JNB bit,rel	ACALL addr11	RETI	RLC A	ADDC A,#data	ADDC A,dir	ADDC A,@Ri		ADDC A,Rr							
							0	1	0	1	2	3	4	5	6	7
4	JC rel	AJMP addr11	ORL dir,A	ORL dir,#data	ORL A,#data	ORL A,dir	ORL A,@Ri		ORL A,Rr							
							0	1	0	1	2	3	4	5	6	7
5	JNC rel	ACALL addr11	ANL dir,A	ANL dir,#data	ANL A,#data	ANL A,dir	ANL A,@Ri		ANL A,Rr							
							0	1	0	1	2	3	4	5	6	7
6	JZ rel	AJMP addr11	XRL dir,A	XRL dir,#data	XRL A,#data	XRL A,dir	XRL A,@Ri		XRL A,Rr							
							0	1	0	1	2	3	4	5	6	7
7	JNZ rel	ACALL addr11	ORL C,bit	JMP @A+DPTR	MOV A,#data	MOV dir,#data	MOV @Ri,#data		MOV Rr,#data							
							0	1	0	1	2	3	4	5	6	7
8	SJMP rel	AJMP addr11	ANL C,bit	MOVC A,@A+PC	DIV AB	MOV dir,dir	MOV dir,@Ri		MOV dir,Rr							
							0	1	0	1	2	3	4	5	6	7
9	MOV DPTR, #data	ACALL addr11	MOV bit,C	MOVC A,@A+DPTR	SUBB A,#data	SUBB A,dir	SUBB A,@Ri		SUBB A,Rr							
							0	1	0	1	2	3	4	5	6	7
A	ORL C,/bit	AJMP addr11	MOV C,bit	INC DPTR	MUL AB		MOV @Ri,dir		MOV Rr,dir							
							0	1	0	1	2	3	4	5	6	7
B	ANL C,/bit	ACALL addr11	CPL bit	CPL C	CJNE A, #data,rel	CJNE A,dir,rel	CJNE @Ri,#data,rel		CJNE Rr,#data,rel							
							0	1	0	1	2	3	4	5	6	7
C	PUSH dir	AJMP addr11	CLR bit	CLR C	SWAP A	XCH A,dir	XCH A,@Ri		XCH A,Rr							
							0	1	0	1	2	3	4	5	6	7
D	POP dir	ACALL addr11	SETB bit	SETB C	DA A	DJNZ dir,rel	XCHD A,@Ri		DJNZ Rr,rel							
							0	1	0	1	2	3	4	5	6	7
E	MOVX A,@DPTR	AJMP addr11	MOVX A,@Ri		CLR A	MOV * A,dir	MOV A,@Ri		MOV A,Rr							
			0	1			0	1	0	1	2	3	4	5	6	7
F	MOVX @DPTR,A	ACALL addr11	MOVX @Ri,A		CPL A	MOV dir,A	MOV @Ri,A		MOV Rr,A							
			0	1			0	1	0	1	2	3	4	5	6	7

* MOV A,ACC is not a valid instruction.

DEVELOPMENT DATA

RATINGS

Limiting values in accordance with the Absolute Maximum System (IEC 134)

Input voltage on any pin with respect to ground (V_{SS})	V_I	−0,5 to +6,5	V
Input, output current	$\pm I_I$, I_O	max. 5	mA
Total power dissipation	P_{tot}	max. 1	W
Storage temperature range	T_{stg}	−65 to +150	°C
Operating ambient temperature range	T_{amb} (PCB)	0 to + 70	°C
	(PCF)	−40 to + 85	°C
	(PCA80C)	−40 to +125	°C

DC CHARACTERISTICS (PCB80C51/C31BH-3, PCF80C51/C31BH-3)

V_{DD} = 5 V (±20%); V_{SS} = 0 V; T_{amb} = 0 to +70 °C (PCB80C51/C31BH-3) and −40 to +85 °C (PCF80C51/C31BH-3); all voltages with respect to V_{SS} unless otherwise specified

parameter	conditions	symbol	minimum	maximum	unit
Supply voltage		V_{DD}	4,0	6,0	V
Supply current operating (note 1)	V_{DD} = 5,5 V; f_{max} = 12 MHz	I_{DD}	–	18	mA
	V_{DD} = 5,5 V; f_{max} = 16 MHz	I_{DD}	–	23	mA
Idle mode (note 2)	f_{CLK} = 16 MHz, V_{DD} = 5 V ±10%; T_{amb} = 0 to +70 °C	I_{DDI}	–	5,8	mA
Idle mode (note 2)	f_{CLK} = 12 MHz V_{DD} = 5 V ± 10%; T_{amb} = −40 to +85 °C	I_{DDI}	–	4,4	mA
Power-down current (note 3)	2 V ⩽ V_{PD} ⩽ V_{DD} max.	I_{PD}	–	50	µA
Inputs					
LOW level input voltage (except EA)		V_{IL}	−0,5	0,2 V_{DD}−0,1	V
LOW level input voltage (EA)		V_{IL1}	−0,5	0,2 V_{DD}−0,3	V
HIGH level input voltage (except XTAL 1, RST)		V_{IH}	0,2V_{DD}+0,9	V_{DD}+ 0,5	V
HIGH level input voltage (XTAL 1, RST)		V_{IH1}	0,7V_{DD}	V_{DD}+ 0,5	V
Input current logic 0 (Ports 1, 2 and 3	V_I = 0,45 V	$-I_{IL}$	–	50	µA
Input current logic 1 to 0 transition (Ports 1, 2 and 3	V_I = 2 V	$\pm I_{TL}$	–	650	µA
Input leakage current (Port 0, EA)	0,45 V < V_I < V_{DD}	$\pm I_{LI}$	–	10	µA

DC CHARACTERISTICS (continued)

parameter	conditions	symbol	minimum	maximum	unit
Outputs					
LOW level output voltage (note 4) (Ports 1, 2 and 3)	I_{OL} = 1,6 mA	V_{OL}	–	0,45	V
LOW level output voltage (note 4) (Port 0, ALE, $\overline{PSEN}$)	I_{OL} = 3,2 mA	V_{OL1}	–	0,45	V
HIGH level output voltage (Ports 1, 2 and 3)	$-I_{OH}$ = 60 μA; V_{DD} = 5 V ± 10%	V_{OH}	2,4	–	V
	$-I_{OH}$ = 25 μA	V_{OH}	$0{,}75V_{DD}$	–	V
	$-I_{OH}$ = 10 μA	V_{OH}	$0{,}9V_{DD}$	–	V
HIGH level output voltage (note 5) (Port 0 in external Bus mode, ALE, $\overline{PSEN}$)	$-I_{OH}$ = 400 μA; V_{DD} = 5 V ± 10%	V_{OH1}	2,4	–	V
	$-I_{OH}$ = 150 μA	V_{OH1}	$0{,}75V_{DD}$	–	V
	$-I_{OH}$ = 40 μA	V_{OH1}	$0{,}9V_{DD}$	–	V
RST pull-down resistor		R_{RST}	50	150	kΩ
I/O pin capacitance	test frequency = 1 MHz; T_{amb} = 25 °C	$C_{I/O}$	–	10	pF

DEVELOPMENT DATA

DC CHARACTERISTICS (PCA80C51/C31BH-3)

V_{DD} = 5 V (± 10%); V_{SS} = 0 V; T_{amb} = −40 to + 125 °C (PCA80C51/C31BH-3); all voltages with respect to V_{SS} unless otherwise specified.

parameter	conditions	symbol	minimum	maximum	unit
Supply voltage		V_{DD}	4,5	5,5	V
Supply current operating (note 1)	V_{DD} = 5,5 V, f_{max} = 12 MHz	I_{DD}	–	20	mA
Idle mode (note 2)	f_{CLK} = 12 MHz, V_{DD} = 5 V ± 10%; T_{amb} = −40 to + 125 °C	I_{DDI}	–	4,4	mA
Power-down current (note 3)	2 V ⩽ V_{PD} ⩽ V_{DD} max.	I_{PD}	–	75	μA
Inputs					
LOW level input voltage (except $\overline{EA}$)		V_{IL}	−0,5	$0{,}2V_{DD}-0{,}25$	V
Low level input voltage ($\overline{EA}$)		V_{IL1}	−0,5	$0{,}2V_{DD}-0{,}45$	V
HIGH level input voltage (except XTAL 1, RST)		V_{IH}	$0{,}2V_{DD}+1{,}0$	$V_{DD}+0{,}5$	V
HIGH level input voltage (XTAL 1, RST)		V_{IH1}	$0{,}7V_{DD}+0{,}1$	$V_{DD}+0{,}5$	V
Input current logic 0 (Ports 1, 2 and 3)	V_I = 0,45 V	$-I_{IL}$	–	75	μA
Input current logic 1 to 0 transition (Ports 1, 2 and 3)	V_I = 2 V	$\pm I_{TL}$	–	750	μA
Input leakage current (Port 0, $\overline{EA}$)	0,45 V < V_I < V_{DD}	$\pm I_{LI}$	–	10	μA

DC CHARACTERISTICS (continued)

parameter	conditions	symbol	minimum	maximum	unit
Outputs					
LOW level output voltage (note 4) (Ports 1, 2 and 3)	I_{OL} = 1,6 mA	V_{OL}	–	0,45	V
LOW level output voltage (note 4) (Port 0, ALE, $\overline{PSEN}$)	I_{OL} = 3,2 mA	V_{OL1}	–	0,45	V
HIGH level output voltage (Ports 1, 2 and 3)	$-I_{OH}$ = 60 μA; V_{DD} = 5 V ± 10%	V_{OH}	2,4	–	V
	$-I_{OH}$ = 25 μA	V_{OH}	0,75V_{DD}	–	V
	$-I_{OH}$ = 10 μA	V_{OH}	0,9V_{DD}	–	V
HIGH level output voltage (note 5) (Port 0 in external Bus mode, ALE, $\overline{PSEN}$)	$-I_{OH}$ = 400 μA; V_{DD} = 5 V ± 10%	V_{OH1}	2,4	–	V
	$-I_{OH}$ = 150 μA	V_{OH1}	0,75V_{DD}	–	V
	$-I_{OH}$ = 40 μA	V_{OH1}	0,9V_{DD}	–	V
RST pull-down resistor		R_{RST}	50	150	kΩ
I/O pin capacitance	test frequency = 1 MHz; T_{amb} = 25 °C	$C_{I/O}$	–	10	pF

Notes to the DC characteristics

1. The operating supply current is measured with all output pins disconnected; XTAL 1 driven with $t_r = t_f$ = 10 ns; $V_{IL} = V_{SS}$ + 0,5 V; $V_{IH} = V_{DD}$–0,5 V; XTAL 2 not connected; $\overline{EA}$ = RST = Port 0 = V_{DD}.
2. The Idle mode supply current is measured with all output pins disconnected; XTAL 1 driven with $t_r = t_f$ = 10 ns; $V_{IL} = V_{SS}$ + 0,5 V; $V_{IH} = V_{DD}$–0,5 V; XTAL 2 not connected; $\overline{EA}$ = Port 0 = V_{DD}; RST = V_{SS}.
3. The power-down current is measured with all output pins disconnected; XTAL 2 not connected; $\overline{EA}$ = Port 0 = V_{DD}; RST = V_{SS}.
4. Capacitive loading on Port 0 and Port 2 may cause spurious noise pulses to be superimposed on the LOW level output voltage of ALE, Port 1 and Port 3. The noise is due to external Bus capacitance discharging into the Port 0 and Port 2 pins when these pins make a 1-to-0 transition during Bus operations. In the most adverse condition (capacitive loading > 100 pF) the noise pulse on ALE line may exceed 0,8 V. In this event it may be required to qualify ALE with a Schmitt trigger, or use an address latch with a Schmitt trigger STROBE input.
5. Capacitive loading on Port 0 and Port 2 may cause the HIGH level output voltage on ALE and $\overline{PSEN}$ to momentarily fall below the 0,9V_{DD} specification when the address bits are stabilizing.

Table 4 shows the supply current as a function of oscillator frequency and supply voltage in the operational and Idle modes for all three versions of the device.

Table 4 I_{DD} as a function of f_{osc} and V_{DD}

	operational mode I_{DD} maximum (mA)			Idle mode I_{DD} maximum (mA)		
frequency/V_{DD}	4 V	5 V	6 V	4 V	5 V	6 V
1,2 MHz	2,2	3,0	4,1	0,7	1,1	1,5
3,5 MHz	4,3	5,7	7,5	1,1	1,6	2,2
8 MHz	8,3	11	14	1,8	2,7	3,7
12 MHz*	12	16	20	2,5	3,7	5
16 MHz	16	20,5	25	3,5	5	6,5

DEVELOPMENT DATA

* For the PCF80C51/C31BH-3 and the PCA80C51/C31BH-3, $f_{max.}$ is 12 MHz; $f_{min.}$ is 1,2 MHz for all three versions.

AC CHARACTERISTICS

V_{DD} = 5 V ± 20% (PCB80C51/C31BH-3, PCF80C51/C31BH-3); V_{SS} = 0 V; $t_{CKmin.} = 1/f_{max.}$;
V_{DD} = 5 V ± 10% (PCA80C51/C31BH-3); V_{SS} = 0 V; $t_{CKmin.} = 1/f_{max.}$;
T_{amb} = 0 to +70 °C; $t_{CKmin.}$ = 63 ns for PCB80C51/C31BH-3;
T_{amb} = −40 to +85 °C, $t_{CKmin.}$ = 83 ns for PCF80C51/C31BH-3;
T_{amb} = −40 to +125 °C, $t_{CKmin.}$ = 83 ns for PCA80C51/C31BH-3;
C_L = 100 pF (Port 0, ALE and $\overline{PSEN}$); C_L = 80 pF (all other outputs); unless otherwise specified.

parameter	symbol	16 MHz		12 MHz		variable clock		unit
		min.	max.	min.	max.	min.	max.	
Program memory								
ALE pulse duration	t_{LL}	85	–	127	–	$2t_{CK}-40$	–	ns
Address set-up time to ALE	t_{AL}	8	–	28	–	$t_{CK}-55$	–	ns
Address hold time after ALE	t_{LA}	28	–	48	–	$t_{CK}-35$	–	ns
Time from ALE to valid instruction input	t_{LIV}	–	150	–	233	–	$4t_{CK}-100$	ns
Time from ALE to control pulse $\overline{PSEN}$	t_{LC}	23	–	43	–	$t_{CK}-40$	–	ns
Control pulse duration $\overline{PSEN}$	t_{CC}	143	–	205	–	$3t_{CK}-45$	–	ns
Time from $\overline{PSEN}$ to valid instruction input	t_{CIV}	–	83	–	145	–	$3t_{CK}-105$	ns
Input instruction hold time after $\overline{PSEN}$	t_{CI}	0	–	0	–	0	–	ns
Input instruction float delay after $\overline{PSEN}$ (note 3)	t_{CIF}	–	38	–	59	–	$t_{CK}-25$	ns
Address to valid instruction input	t_{AIV}	–	208	–	312	–	$5t_{CK}-105$	ns
Address float time after $\overline{PSEN}$	t_{AFC}	–	10	–	10	–	10	ns

parameter	symbol	16 MHz		12 MHz		variable clock		unit
		min.	max.	min.	max.	min.	max.	
External data memory								
$\overline{RD}$ pulse duration	t_{RR}	275	–	400	–	$6t_{CK}-100$	–	ns
$\overline{WR}$ pulse duration	t_{WW}	275	–	400	–	$6t_{CK}-100$	–	ns
Address hold time after ALE	t_{LA}	28	–	48	–	$t_{CK}-35$	–	ns
$\overline{RD}$ to valid data input	t_{RD}	–	148	–	252	–	$5t_{CK}-165$	ns
Data hold time after $\overline{RD}$	t_{DR}	0	–	0	–	0	–	ns
Data float delay after $\overline{RD}$	t_{DFR}	–	55	–	97	–	$2t_{CK}-70$	ns
Time from ALE to valid data input	t_{LD}	–	350	–	517	–	$8t_{CK}-150$	ns
Address to valid data input	t_{AD}	–	398	–	585	–	$9t_{CK}-165$	ns
Time from ALE to $\overline{RD}$ or $\overline{WR}$	t_{LW}	138	238	200	300	$3t_{CK}-50$	$3t_{CK}+50$	ns
Time from address to $\overline{RD}$ or $\overline{WR}$	t_{AW}	120	–	203	–	$4t_{CK}-130$	–	ns
Time from $\overline{RD}$ or $\overline{WR}$ HIGH to ALE HIGH	t_{WHLH}	23	103	43	123	$t_{CK}-40$	$t_{CK}+40$	ns
Data valid to $\overline{WR}$ transition	t_{DWX}	3	–	23	–	$t_{CK}-60$	–	ns
Data set-up time before $\overline{WR}$	t_{DW}	288	–	433	–	$7t_{CK}-150$	–	ns
Data hold time after $\overline{WR}$	t_{WD}	13	–	33	–	$t_{CK}-50$	–	ns
Address float delay after $\overline{RD}$	t_{AFR}	–	0	–	0	–	0	ns

Where:

$1/t_{CK}$ = 1,2 to 16 MHz (see Fig. 14 and Table 5).

Notes to the AC characteristics

1. The maximum operating frequency is:
   16 MHz for the PCB80C51/C31BH-3 and
   12 MHz for the PCF80C51/C31BH-3 and PCA80C51/C31BH-3.
2. The minimum operating frequency is 1,2 MHz for all versions.
3. Interfacing the 80C51/C31BH-3 to devices with float times up to 75 ns is permitted.
   This limited bus contention will not cause damage to Port 0 drivers.

AC CHARACTERISTICS (continued)

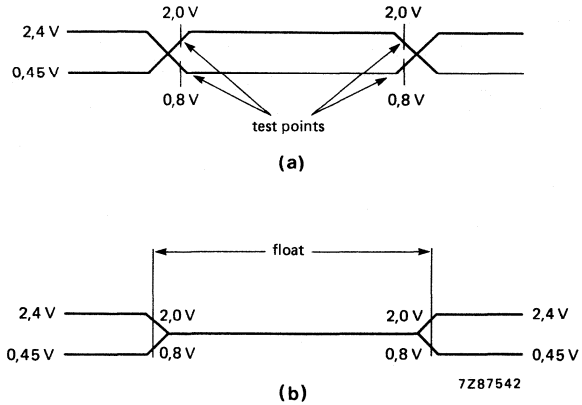


Fig. 13 AC testing input, output waveform (a) and float waveform (b).

AC testing inputs are driven at 2,4 V for a logic 1 and 0,45 V for a logic 0.
Timing measurements are taken at 2,0 V for a logic 1 and 0,8 V for a logic 0.
The float state is defined as the point at which a Port 0 pin sinks 3,2 mA or sources 400 μA at the voltage test levels.

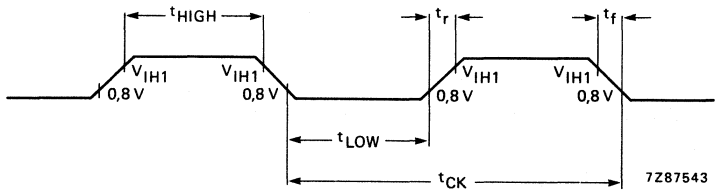


Fig. 14 External clock drive XTAL 1 (see Table 5).

Table 5 External clock drive XTAL 1 (see Fig. 14)

parameter	symbol	variable clock (f = 1,2 to 16 MHz)*		unit
		min.	max.	
oscillator clock period	t_{CK}	63	833,3	ns
HIGH time	t_{HIGH}	20	$t_{CK}-t_{LOW}$	ns
LOW time	t_{LOW}	20	$t_{CK}-t_{HIGH}$	ns
rise time	t_r	–	20	ns
fall time	t_f	–	20	ns

* The maximum operating frequency for the PCF80C51/C31BH-3 and PCA80C51/C31BH-3 is 12 MHz.

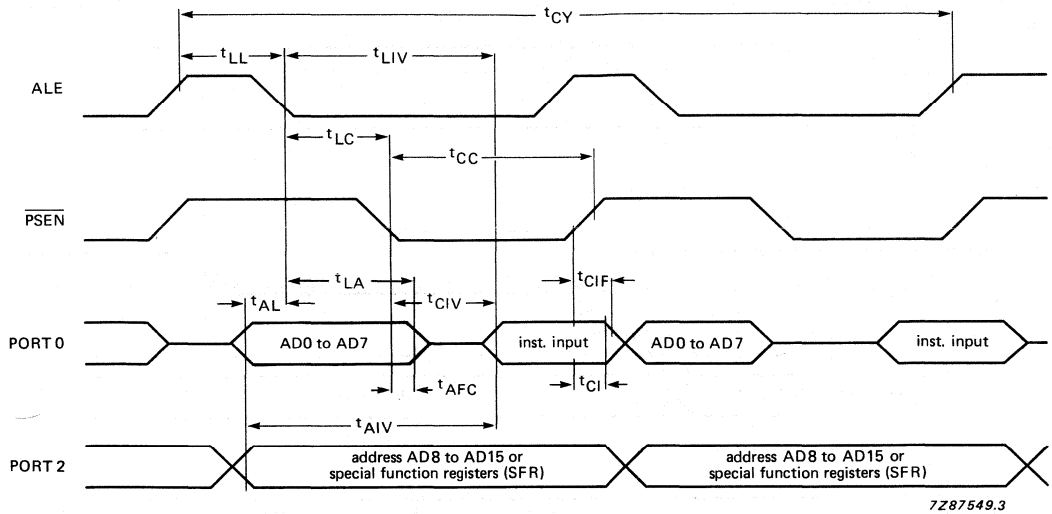


Fig. 15 Read from program memory.

DEVELOPMENT DATA

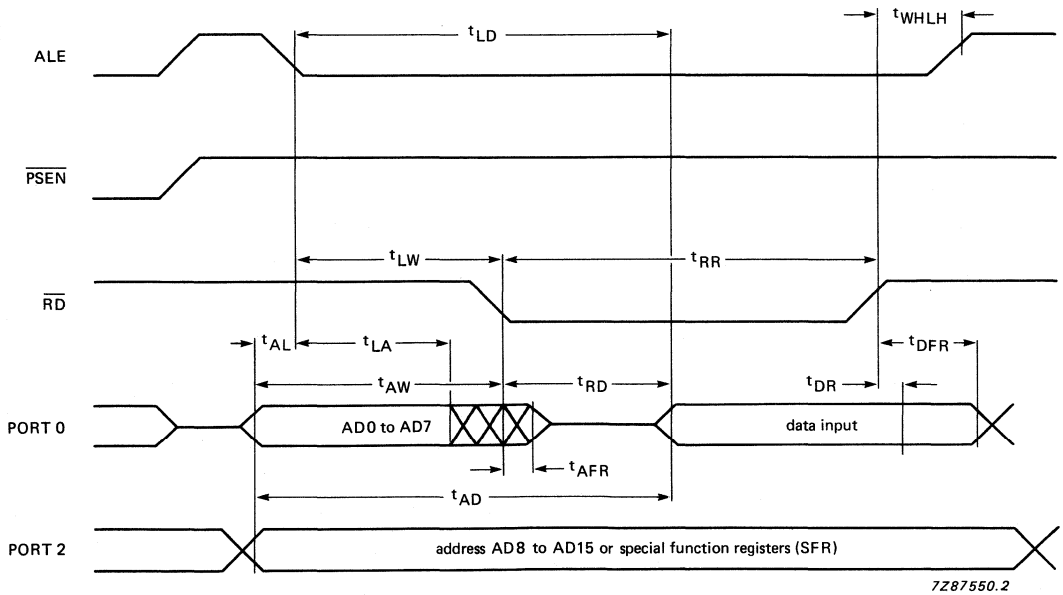


Fig. 16 Read from data memory.

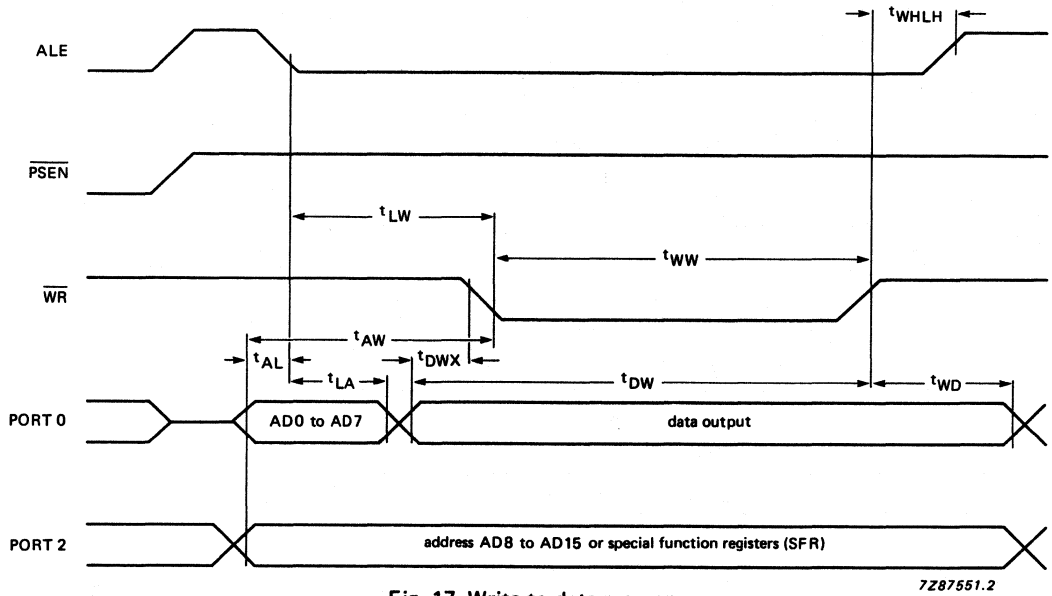


Fig. 17 Write to data memory.

DEVELOPMENT DATA

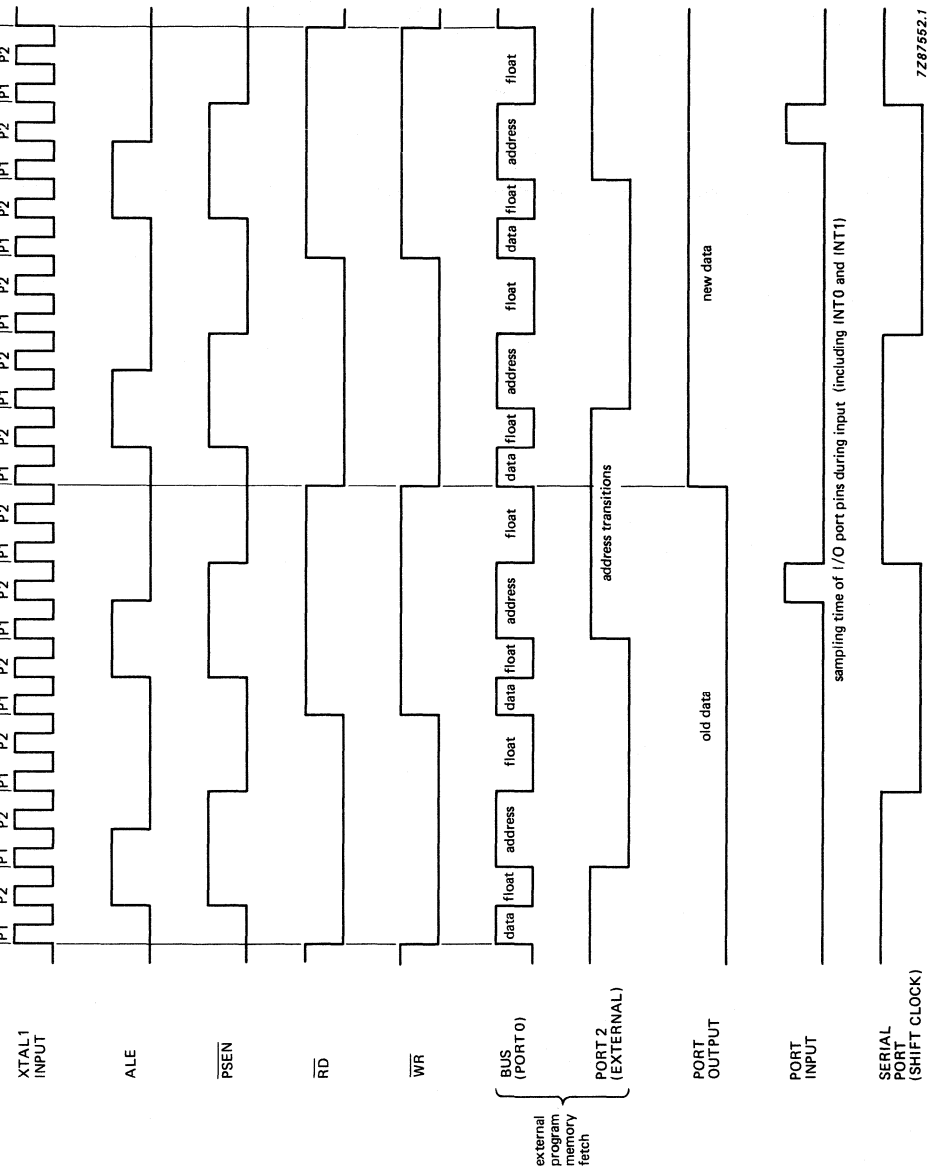


Fig. 18 Instruction cycle timing.

SUPERSEDES DATA OF SEPTEMBER 1986

SINGLE-CHIP 8-BIT CMOS MICROCONTROLLER

DESCRIPTION

The PC80CXX family of single-chip 8-bit CMOS microcontrollers consists of:

- The PCB80C49 with resident mask programmed 2 K x 8 ROM, 128 x 8 RAM.
- The PCB80C39 without resident program memory for use with external EPROM/ROM, 128 x 8 RAM.

All versions are pin and function compatible to their NMOS counter parts but with additional features and high performance.

The PC80CXX family are designed to be efficient control processors as well as arithmetic processors. Their instruction set allows the user to directly set and reset individual I/O, and to test individual individual bits within the accumulator. A large variety of branch and table look-up instructions enable efficient implementation of standard logic functions. Code efficiency is high; over 70% of the instructions are single byte; all others are two byte.
An on-chip 8-bit counter is provided, which can count either machine cycles (÷ 32) or external events. The counter can be programmed to cause an interrupt to the processor.
Program and data memories can be expanded using standard devices. Input/output capabilities can be expanded using standard devices.
The family has low power consumption and in addition a power down mode is provided.
For further detailed information see users manual 'single-chip 8-bit microcontrollers'.

Features

- 8-bit CPU, ROM, RAM, I/O in a single 40-pin package
- PCB80C49: 2K x 8 ROM, 128 x 8 RAM
- Internal counter/timer
- Internal oscillator, clock driver
- Single-level interrupts: external and counter/timer
- 17 internal registers: accumulator, 16 addressable registers
- Over 90 instructions: 70% single byte
- All instructions: 1 or 2 cycles
- Easily expandable memory and I/O
- TTL compatible inputs and outputs
- Single 5 V supply
- Wide frequency operating range
- Low current consumption
- Available with extended temperature ranges: (PCB version) 0 to + 70 °C
  (PCF version) −40 to + 85 °C
  (PCA version) −40 to + 110 °C
- Frequency range: 1 to 15 MHz for all temperature ranges

APPLICATIONS

- Peripheral interfaces and controllers
- Test and measurement instruments
- Sequencers
- Audio/video systems
- Environmental control systems
- Modems and data enciphering

PACKAGE OUTLINES

PCB/F/A80C39/C49: 40-lead DIL; plastic (SOT129).
PCB/F/A80C39/C49WP: 44-lead PLCC; plastic leaded chip carrier, 'pocket' version (SOT187AA); 'pedestal' version (SOT187). These versions are interchangeable.

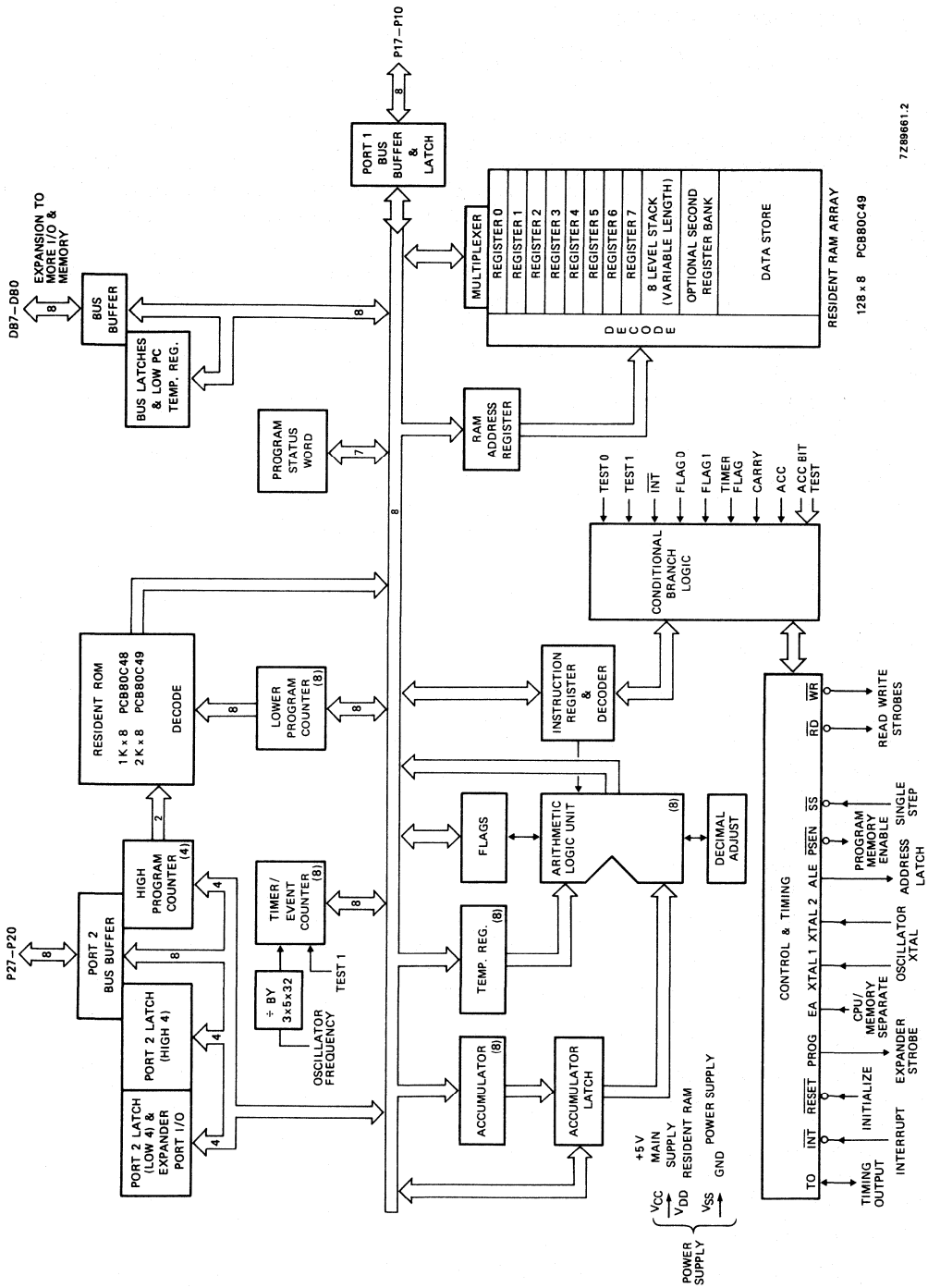


Fig. 1 Block diagram.

DEVELOPMENT DATA

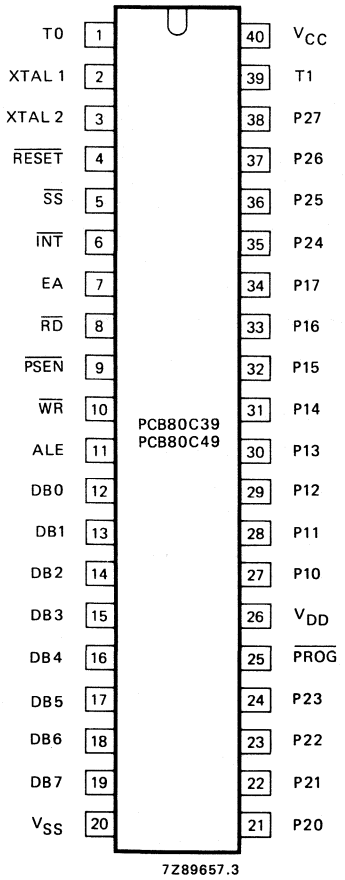


Fig. 2(a) Pinning diagram, DIL; for pin designation see next page.

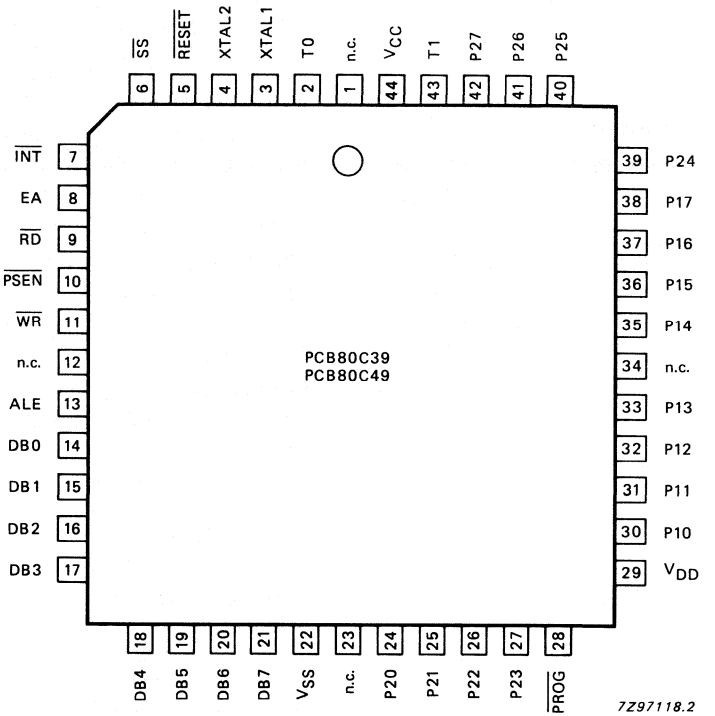


Fig. 2(b) Pinning diagram, PLCC.

Where: n.c. = not connected.

PIN DESIGNATION

designation	pin no.	function
DB0–DB7	12–19	**BUS.** Bidirectional I/O port that can be read or written using the $\overline{RD}$ and $\overline{WR}$ strobes. This port can also be statically latched. Contains the 8 lower order address bits during external memory access and receives the addressed instruction under control of $\overline{PSEN}$. $\overline{PSEN}$, ALE, $\overline{RD}$ and $\overline{WR}$ determine whether the access is an instruction fetch or a read/write access to external RAM.
P10–P17	27–34	**Port 1.** 8-bit quasi-bidirectional I/O port (note 1).
P20–P27	21–24, 35–38	**Port 2.** 8-bit quasi-bidirectional I/O port (note 1). P20–P23 contain the 4 higher order address bits during an access of external program memory.
$\overline{PROG}$	25	**Output strobe** (active LOW) for I/O expander.
T0	1	**Input pin** sensed using the JT0 and JNT0 instructions. **Clock output pin** when designated as such by the ENT0 CLK instruction.
T1	39	**Input pin** sensed using the JT1 and JNT1 instructions. Can be designated as the timer/counter input by the STRT CNT instruction.
$\overline{INT}$	6	**Interrupt input pin.** When LOW causes an interrupt in the current program if external interrupt is enabled. Can also be used as an input, testable using the JNI instruction. Interrupt is disabled during and after a RESET.
$\overline{RESET}$	4	**Reset input pin** used to initialize the microcontroller. Active LOW. During program verification the address is latched by a '0' to '1' transition on $\overline{RESET}$ and the data at the addressed location is output on BUS (note 2).
ALE	11	**Address latch enable.** Occurs once each machine cycle and is useful for timing and sampling. During external program or data memory access, ALE is used to strobe the address information multiplexed on the DB0 to DB7 outputs.
$\overline{RD}$	8	**Read BUS.** Active LOW strobe used to gate data onto BUS lines when reading from an external source.
$\overline{WR}$	10	**Write BUS.** Active LOW strobe used to write data from BUS lines to an external designation.
EA	7	**External access input.** When HIGH, forces instruction fetch from external memory.
$\overline{PSEN}$	9	**Program store enable.** Active LOW strobe that occurs only during a fetch from external program memory.
$\overline{SS}$	5	**Single step input.** Active LOW which is used with ALE to cause the microcontroller to execute a single instruction.
V_{DD}	26	**RAM power supply,** + 5 V during normal operation and power-down mode.

designation	pin no.	function
XTAL1	2	**One side of crystal** (or inductor) input for internal oscillator. Can also be used as an input for an external timing source (note 2).
XTAL2	3	**Other side of crystal.** XTAL2 must be driven with the inverted signal of XTAL1 when an external timing source of $11 < f_{osc} \leqslant 15$ MHz is used.
V_{SS}	20	**Ground.**
V_{CC}	40	**Mainly power supply**, + 5 V during normal operation.

Notes

1. Each port line can be designated as an input or an output. A line is designated as an input by first writing a logic 1 to the line. $\overline{\text{RESET}}$ sets all lines to logic 1.
2. Non-standard TTL V_{IH}.

FUNCTIONAL DESCRIPTION

Program memory (see Fig. 3)

The resident program memory is:
2048 byte ROM
The PCB80C39 has no resident program memory.
The total addressing capability is 4096 bytes.
The program memory address space is divided into two 2048-bytes banks MB0 and MB1.
The program memory is also divided into pages of 256 bytes for conditional branches.

There are three locations in program memory of special importance. These locations contain the first instruction to be executed after one of three events.
The three locations and their contents are:
location 0 — activation, then deactivation of the $\overline{\text{RESET}}$ line,
location 3 — activation of the $\overline{\text{INT}}$ line when the external interrupt is enabled,
location 7 — an overflow of the timer/counter if the T/C interrupt is enabled.

Data memory (see Fig. 4)

The resident data memory is: 128 byte RAM.

All locations are indirectly addressable by either of two RAM pointer registers at locations 0 and 1.
The first eight locations of RAM (0 to 7) are designated as working registers and are directly addressable by several instructions. By selecting register bank 1, RAM locations 24 to 31 become the working registers, replacing those in register bank 0 (0 to 7).
RAM locations 8 to 23 are designated as the stack. Two locations (bytes) are used per CALL, allowing up to eight levels of subroutine nesting.
If additional RAM is required, up to 256 bytes may be added and addressed directly using the MOVX instructions. If more RAM is required, an I/O port can be used to select one (256 byte) bank of external memory at a time.

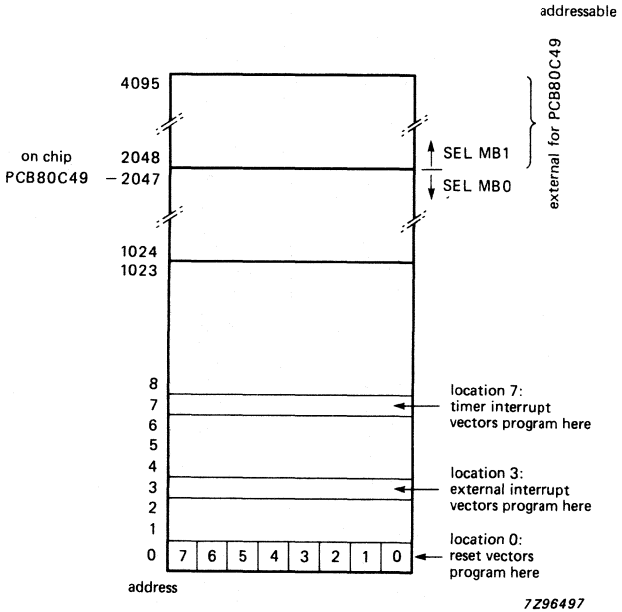


Fig. 3 Program memory map.

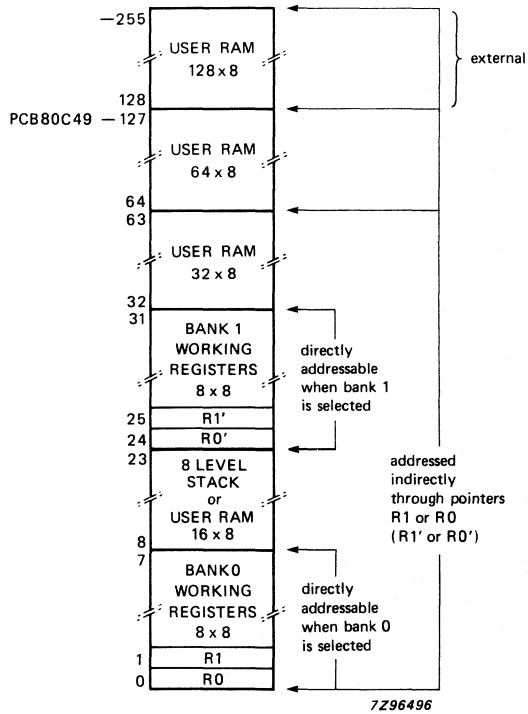


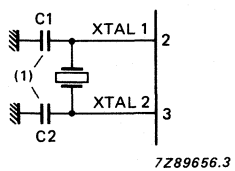
Fig. 4 Data memory map. In addition R0 or R1 (R0' or R1') may be used to address 256 words of an external RAM.

Program counter and stack

The program counter (PC) is a 12-bit counter/register that points to the location from which the next instruction is to be fetched. When EA is '0', the PC addresses an internal program memory. At the boundary of the internal program memory an automatic switch over to external memory is made. When EA is '1', all the program is fetched from external ROM/EPROM. The total address space is 4K bytes. An interrupt or CALL to a subroutine causes the contents of the program counter to be stored in one of the 8 register pairs of the program counter stack. The pair to be used is determined by a 3-bit stack pointer which is part of the program status word (PSW). Data RAM locations 8 to 23 are available as stack registers and are used to store the program counter and 4 bits of PSW. The stack pointer, when initialized to 000B, points to RAM locations 8 and 9. The first subroutine jump or interrupt results in the program counter contents being transferred to locations 8 and 9 of the RAM array. The stack pointer is then incremented by one to point to locations 10 and 11 in anticipation of another CALL. Nesting of subroutines within subroutines can continue up to 8 times without overflowing the stack. If overflow does occur the deepest address stored (location 8 and 9) will be overwritten and lost since the stack pointer overflows from 111 to 000. It also underflows from 000 to 111.
The end of a subroutine, which is signalled by a return instruction (RET or RETR), causes the stack pointer to be decremented and the contents of the resulting register pair to be transferred to the program counter.

Oscillator and clock

The PCB80C49 contains its own internal oscillator and clock driver.
A crystal, inductor or external pulse generator determines the oscillator frequency (see Figs 5, 6 and 7). The output of the oscillator is divided-by-three and is available at T0 (pin 1) by executing the ENT0 CLK instruction. This CLK signal is divided-by-five to define a machine (instruction) cycle. It is available at ALE (pin 11).



For quartz crystal
1 to 11 MHz: C1 = C2 = 15 to 25 pF
For ceramic resonators
1 to 11 MHz: C1 = C2 = 30^{+5}_{-10} pF

(1) Including crystal-socket stray capacitance.

Fig. 5 Crystal oscillator mode. Typical values are given. Crystal serial impedance should be $< 75\ \Omega$ at 6 MHz and $< 180\ \Omega$ at 3,6 MHz.

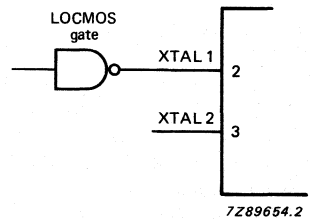
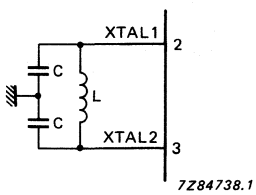


Fig. 6 Driving from an external source. Test conditions at XTAL1; Minimum HIGH (> 0,7 of V_{CC}) and LOW (< 0,13 of V_{CC}) times, should be at least 45% of a clock period. XTAL2 must be driven with the inverted signal of XTAL1 when an external timing source of $11 < f_{osc} \leqslant 15$ MHz is used.



L (μH)	C (pF)	f_{nom} (MHz)
45	20	5,2
120	20	3,2

Fig. 7 LC oscillator mode.

FUNCTIONAL DESCRIPTION (continued)

Timer/event counter

An internal counter is available which can count either external events or machine cycles (÷ 32). The machine cycles are divided-by-32 before they are applied to the input of the 8-bit counter. External events are applied directly to the input of the counter. The maximum frequency that can be counted is one third of the machine cycle frequency. The minimum positive duty cycle that can be detected is 0,2 times the cycle period. The counter is under program control and can be made to generate an interrupt to the processor when it overflows.

Interrupt

An interrupt may be generated by either an external input ($\overline{INT}$, pin 6) or the overflow of the internal timer/event counter, when enabled. In either case, the processor completes execution of the present instruction and then does a CALL to the interrupt service routine, After service, a RETR instruction restores the machine to the state it was prior to the interrupt. The external interrupt has priority over the internal interrupt.

Input/output

The PCB80CXX family has 27 I/O lines. These lines are arranged as three 8-line ports, which serve individually as either inputs, outputs or together as bidirectional ports, plus 3 'test' inputs which can alter program sequences when tested by conditional jump instructions.

Ports 1 and 2

Ports 1 and 2 are each 8-bits wide and have identical characteristics. Data written to these ports is statically latched and remains unchanged until rewritten. As input ports these lines are non-latching, e.g., inputs must be present until read by an input instruction. Inputs are fully TTL compatible and outputs will drive one standard TTL load.

The lines of ports 1 and 2 are called quasi-bidirectional because of a special output circuit structure which allows each line to serve as an input, an output, or both even though outputs are statically latched. The circuit configuration is shown in Fig. 8. Each line has a unique high-impedance pull-up transistor TR3, this is turned on when the line is pulled above 2 V by an external source or by writing a logic 1 to the port. This pull-up is sufficient to provide the source current for a TTL HIGH level, yet can be pulled LOW by a standard TTL gate, thus allowing the same pin to used for both input and output. When a logic 1 is written to a line, a second high impedance transistor TR2, pulls the line up to 5 V. To provide fast switching during a '0' to '1' transition, a relatively low-impedance transistor TR1 (approx. 750 Ω) is switched on for 1/5 of a machine cycle whenever a '1' is written to the line. Whenever a '0' is written to the line, a low-impedance (approx. 250 Ω) transistor TR4, overcomes the weak light pull-up and provides TTL current sinking capability.

Since the pull-down transistor TR4 is a low-impedance device, a '1' must first be written to any line which is to be used as an input. RESET initializes all lines to the high impedance '1' state. This structure allows input and output on the same pin and also allows a mixture of input lines and output lines on the same port. The quasi-bidirectional port in combination with the ANL and ORL logical instructions provide an efficient means for handling single line inputs and outputs within an 8-bit processor.

BUS

BUS is a true bidirectional 8-bit port with associated input and output strobes. The BUS port can operate in three different modes: as a latched I/O port, as a bidirectional bus port, or in an expanded system as a program memory address output port. If the bidirectional feature is not needed, BUS can serve as either a statically latched output port or non-latching input port. Input and output lines on this port cannot be mixed. The BUS port lines are either active HIGH, active LOW, or high impedance (floating).

As a static port, data is written and latched using the OUTL instruction and input using the INS instruction. The INS and OUTL instructions generate pulses on the corresponding RD and WR output strobe lines; however, in the static port mode they are generally not used. As a bidirectional port, the MOVX instructions are used to read and write the port. A write to the port generates a pulse on the WR output line and output data is valid at the trailing edge of WR. A read of the port generates a pulse on the RD output line and input data must be valid at the trailing edge of RD.

The latched mode (INS, OUTL) is intended for use in the single-chip configuration, where BUS is not being used as an expanded port. OUTL and MOVX instructions can be mixed if required. However, when using a MOVX instruction a previously latched output will be lost and BUS will be left in the high impedance state. INS does not put the BUS in a high impedance state. Therefore, in order to read an external byte (and not the previously latched value) using the INS instruction, it is necessary to follow an OUTL with a MOVX instruction to place BUS in a high impedance state.

OUTL should never be used in a system with external program memory, since latching BUS may cause the next instruction to be incorrectly fetched.

Test (T0, T1) and $\overline{INT}$ inputs

Three pins serve as inputs and are testable with the conditional jump instruction. These pins are T0, T1, and $\overline{INT}$ and they allow inputs to cause program branches without the necessity to load an input port into the accumulator. The T0, T1 and $\overline{INT}$ pins have other possible functions as well.

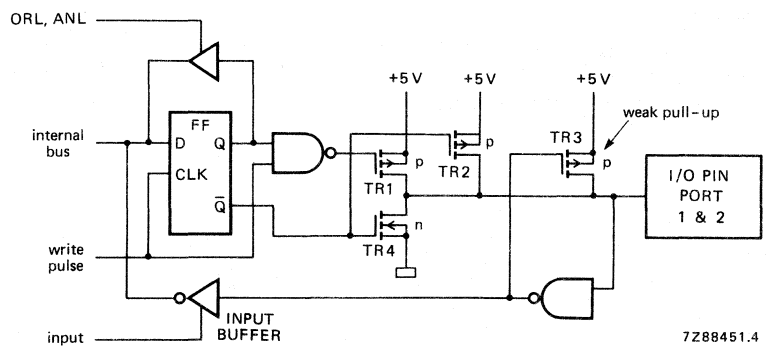


Fig. 8 Quasi-bidirectional port structure.

FUNCTIONAL DESCRIPTION (continued)

$\overline{\text{RESET}}$ input (see Fig. 10)

The $\overline{\text{RESET}}$ input provides a means to initialize the processor. This Schmitt-trigger input has an internal pull-up resistor. The combination of an external 47 kΩ resistor and a 1 μF capacitor provides a reset pulse of sufficient duration to guarantee that all circuitry is reset. If the reset pulse is generated otherwise, the $\overline{\text{RESET}}$ pin must be held at ground for at least 10 ms after the (0,13 V_{CC}) power supply is within tolerance. Only five machine cycles (2,5 μs at 6 MHz) are required if power is already on and the oscillator has stabilized.

Single step input ($\overline{\text{SS}}$)

Under control of the $\overline{\text{SS}}$ line, the processor can be forced to execute one instruction and then to wait until the single step switch is activated again.

IDLE mode

The PCB80CXX family is provided with a IDLE mode in which the internal oscillator, the internal timer and the external interrupt and counter are still functioning, while the status of following parts is maintained: RAM and register/Port 1 and 2/Bus. The IDLE mode is entered after execution of the IDLE instruction (opcode 1H). The IDLE mode is terminated by one of the two possible interrupts, if enabled, or a RESET signal. If an external interrupt terminates the IDLE mode, the next instruction that is executed is at location 3 of the program store. If a timer/counter interrupt terminates the IDLE mode, the next instruction is at location 7. The reset signal will terminate the IDLE mode, and also initialize the processor.

Power-down mode

In the PCB80CXX family, power can be removed from all but the 64 x 8 bit and 128 x 8 bit data RAM array, for low power standby operation. In the power-down mode the contents of the data RAM are maintained. V_{CC} serves as the + 5 V supply pin for most of the circuitry, while the V_{DD} pin supplies only the RAM array. In normal operation, both pins are at + 5 V. In standby, V_{CC} is at ground and V_{DD} is maintained at + 5 V.

Applying $\overline{\text{RESET}}$ to the processor through the $\overline{\text{RESET}}$ pin inhibits any access to the RAM by the processor and guarantees that the RAM cannot be inadvertently altered when power is removed from V_{CC}. Fig. 9 shows a typical power-down sequence.

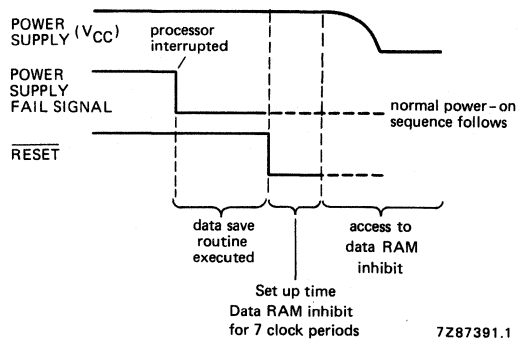


Fig. 9 Power-down sequence.

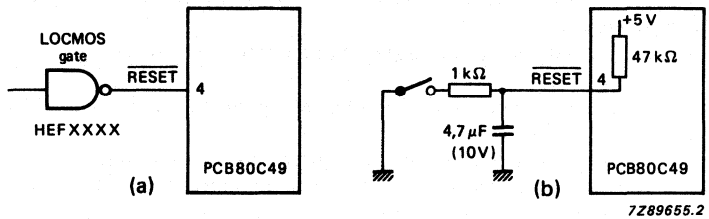


Fig. 10(a) External reset circuit; (b) power-on reset.

Instruction set

The PCB80CXX instruction set consists of over 90 one and two-byte instructions (see Table 2). Program code efficiency is high because:

- working registers and program variables are stored in the RAM, which require only a single byte to address,
- program memory is divided into pages of 256 bytes, which means that branch destination addresses require one byte.

In addition to performing logical and arithmetic operations, the instruction set manipulates and tests both bits and bytes. A set of MOVE instructions operates indirectly upon either RAM or ROM, which permits efficient access of pointers and data tables. The indirect jump instruction performs a multi-way branch (up to 256) upon the content of the accumulator to addresses stored in a look-up table. The 'decrement register and jump if not zero' DJNZ instruction saves a byte every time it is used as opposed to using separate increment and test instructions.

The on-chip counter enables either external events or time to be counted off-line from the main program. The PCB80CXX can either test the counter (under program control) or cause its overflow to generate an interrupt. These features are required for real-time applications. Instruction timing is shown in Table 3.

Table 1

Symbol definitions used in Table 2.

symbol	description
A	the accumulator
AC	the auxiliary carry flag
addr	program memory address (11-bits)
Bb	bit designation (b = 0–7)
BS	the bank switch
C	carry flag
CLK	clock signal
CNT	event counter
D	nibble designation (4-bits)
DBF	program memory bank flip-flop
data	number or expression (8-bits)
F0, F1	flags 0 and 1
$\overline{\text{I}}$	interrupt
$\overline{\text{INT}}$	external interrupt
P	'in-page' operation designation
Pp	port designation (p = 1, 2 or 4–7)
PSW	program status word
Rr	register designation (r = 0, 1 or 0–7)
SP	stack pointer
T	timer
TF	timer flag
T0, T1	test 0 and 1 inputs
#	immediate data prefix
@	indirect address prefix
$	current value of program counter
←	is replaced by
↔	is exchanged with

Notes to Table 2.

1. Instruction code designations r and p form the binary representation of the registers and ports involved.
2. The dot under the appropriate flag bit indicates that its content is subject to change by the instruction it appears in.
3. Numerical subscripts appearing in the FUNCTION column reference the specific bits affected.

Table 2 Instruction set

	mnemonic	function	description	instruction code D_7 D_6 D_5 D_4 D_3 D_2 D_1 D_0	cycles	bytes	flags C	AC	F0	F1	BS
ACCUMULATOR	ADD A,Rr	(A)←(A) + (Rr) for r = 0–7	Add contents of designated register to A	0 1 1 0 1 r r r	1	1	•	•			
	ADD A,@Rr	(A)←(A) + ((Rr)) for r = 0–1	Add indirect the contents of the data memory location to A	0 1 1 0 0 0 0 r	1	1	•	•			
	ADD A,#data	(A)←(A) + data	Add immediate data to A	0 0 0 0 0 0 1 1 d_7 d_6 d_5 d_4 d_3 d_2 d_1 d_0	2	2	•	•			
	ADDC A,Rr	(A)←(A) + (C) + (Rr) for r = 0–7	Add with carry the contents of designated register to A	0 1 1 1 1 r r r	1	1	•	•			
	ADDC A,@Rr	(A)←(A) + (C) + ((Rr)) for r = 0–1	Add indirect with carry the contents of the data memory location to A	0 1 1 1 0 0 0 r	1	1	•	•			
	ADDC A,#data	(A)←(A) + (C) + data	Add immediate data with carry to A	0 0 0 1 0 0 1 1 d_7 d_6 d_5 d_4 d_3 d_2 d_1 d_0	2	2	•	•			
	ANL A,Rr	(A)←(A) AND (Rr) for r = 0–7	Logical AND contents of designated register with A	0 1 0 1 1 r r r	1	1					
	ANL A,@Rr	(A)←(A) AND ((Rr)) for r = 0–1	Logical AND indirect the contents of data memory with A	0 1 0 1 0 0 0 r	1	1					
	ANL A,#data	(A)←(A) AND data	Logical AND immediate data with A	0 1 0 1 0 0 1 1 d_7 d_6 d_5 d_4 d_3 d_2 d_1 d_0	2	2					
	CLR A	(A)←0	Clear the contents of A	0 0 1 0 0 1 1 1	1	1					
	CPL A	(A)←NOT(A)	Complement the contents of A	0 0 1 1 0 1 1 1	1	1					
	DA A		Decimal adjust the contents of A	0 1 0 1 0 1 1 1	1	1	•	•			
	DEC A	(A)←(A) –1	Decrement the contents of A by 1	0 0 0 0 0 1 1 1	1	1					
	INC A	(A)←(A) + 1	Increment the contents of A by 1	0 0 0 1 0 1 1 1	1	1					
	ORL A,Rr	(A)←(A) OR (Rr) for r = 0–7	Logical OR contents of designated register with A	0 1 0 0 1 r r r	1	1					
	ORL A,@Rr	(A)←(A) OR ((Rr)) for r = 0–1	Logical OR indirect the contents of data memory location with A	0 1 0 0 0 0 0 r	1	1					

DEVELOPMENT DATA

ACCUMULATOR (continued)	ORL A,#data	(A)←(A) OR data	Logical OR immediate data with A	0	1	0	0	0	0	1	1	2	2										
				d_7	d_6	d_5	d_4	d_3	d_2	d_1	d_0												
	RL A	$(A_{n+1})\leftarrow(A_n)$ $(A_0)\leftarrow(A_7)$ for n = 0–6	Rotate A left by 1-bit without carry	1	1	1	0	0	1	1	1	1	1										
	RLC A	$(A_{n+1})\leftarrow(A_n)$ $(A_0)\leftarrow(C)$ $(C)\leftarrow(A_7)$ for n = 0–6	Rotate A left by 1-bit through carry	1	1	1	1	0	1	1	1	1	1	•									
	RR A	$(A_n)\leftarrow(A_{n+1})$ $(A_7)\leftarrow(A_0)$ for n = 0–6	Rotate A right by 1-bit without carry	0	1	1	1	0	1	1	1	1	1										
	RRC A	$(A_n)\leftarrow(A_{n+1})$ $(A_7)\leftarrow(C)$ $(C)\leftarrow(A_0)$ n = 0–6	Rotate A right by 1-bit through carry	0	1	1	0	0	1	1	1	1	1	•									
	SWAP A	$(A_{4-7})\leftrightarrow(A_{0-3})$	Swap the two 4-bit nibbles in A	0	1	0	0	0	1	1	1	1	1										
	XRL A,Rr	(A)←(A) XOR (Rr)	Logical XOR contents of designated register with A	1	1	0	1	1	r	r	r	1	1										
	XRL A,@Rr	(A)←(A) XOR ((Rr))	Logical XOR indirect the contents of data memory location with A	1	1	0	1	0	0	0	r	1	1										
	XRL A,#data	(A)←(A) XOR data	Logical XOR immediate data with A	1	1	0	1	0	0	1	1	2	2										
				d_7	d_6	d_5	d_4	d_3	d_2	d_1	d_0												

Table 2 (continued)

	mnemonic	function	description	instruction code D_7 D_6 D_5 D_4 D_3 D_2 D_1 D_0	cycles	bytes	flags C	AC	F0	F1	BS
BRANCH	DJNZ Rr, addr	(Rr)←(Rr) −1 if (Rr) not zero: (PC_{0-7})←addr	Decrement the specified register and test contents	1 1 1 0 1 r r r a_7 a_6 a_5 a_4 a_3 a_2 a_1 a_0	2	2					
	JBb addr	(PC_{0-7})←addr if Bb = 1; (PC)←(PC) + 2 if Bb = 0	Jump to specified address if accumulator bit is set	b_2 b_1 b_0 1 0 0 1 0 a_7 a_6 a_5 a_4 a_3 a_2 a_1 a_0	2	2					
	JC addr	(PC_{0-7})←addr if C = 1; (PC)←(PC) + 2 if C = 0	Jump to specified address if carry flag is set	1 1 1 1 0 1 1 0 a_7 a_6 a_5 a_4 a_3 a_2 a_1 a_0	2	2					
	JF0 addr	(PC_{0-7})←addr if (F0 = 1; (PC)←(PC) + 2 if F0 = 0	Jump to specified address if flag F0 is set	1 0 1 1 0 1 1 0 a_7 a_6 a_5 a_4 a_3 a_2 a_1 a_0	2	2					
	JF1 addr	(PC_{0-7})←addr if F1 = 1; (PC)←(PC) + 2 if F1 = 0	Jump to specified address if flag F1 is set	0 1 1 1 0 1 1 0 a_7 a_6 a_5 a_4 a_3 a_2 a_1 a_0	2	2					
	JMP addr	(PC_{8-10})←$addr_{8-10}$ (PC_{0-7})←$addr_{0-7}$ (PC_{11})←(DBF)	Direct jump to specified address within the 2K address block	a_{10} a_9 a_8 0 0 1 0 0 a_7 a_6 a_5 a_4 a_3 a_2 a_1 a_0	2	2					
	JMPP @A	(PC_{0-7})←((A))	Jump indirect to specified address within address page	1 0 1 1 0 0 1 1	2	1					
	JNC addr	(PC_{0-7})←addr if C = 0; (PC)←(PC) + 2 if C = 1	Jump to specified address if carry flag is LOW	1 1 1 0 0 1 1 0 a_7 a_6 a_5 a_4 a_3 a_2 a_1 a_0	2	2					
	JNI	(PC_{0-7})←addr if $\overline{INT}$ = 0; (PC)←(PC) + 2 if $\overline{INT}$ = 1	Jump to specified address if $\overline{INT}$ input is LOW	1 0 0 0 0 1 1 0 a_7 a_6 a_5 a_4 a_3 a_2 a_1 a_0	2	2					
	JNT0 addr	(PC_{0-7})←addr if T0 = 0; (PC)←(PC) + 2 if T0 = 1	Jump to specified address if T0 is LOW	0 0 1 0 0 1 1 0 a_7 a_6 a_5 a_4 a_3 a_2 a_1 a_0	2	2					
	JNT1 addr	(PC_{0-7})←addr if T1 = 0; (PC)←(PC) + 2 if T1 = 1	Jump to specified address if T1 is LOW	0 1 0 0 0 1 1 0 a_7 a_6 a_5 a_4 a_3 a_2 a_1 a_0	2	2					

BRANCH (continued)	JNZ addr	$(PC_{0-7}) \leftarrow$addr if A $\neq$ 0; (PC)$\leftarrow$(PC) + 2 if A = 0	Jump to specified address if A is non-zero	1 a_7	0 a_6	0 a_5	1 a_4	0 a_3	1 a_2	1 a_1	0 a_0	2	2					
	JTF addr	$(PC_{0-7}) \leftarrow$addr if TF = 1; (PC)$\leftarrow$(PC) + 2 if TF = 0	Jump to specified address if timer flag is set to 1	0 a_7	0 a_6	0 a_5	1 a_4	0 a_3	1 a_2	1 a_1	0 a_0	2	2					
	JT0 addr	$(PC_{0-7}) \leftarrow$addr if T0 = 1; (PC)$\leftarrow$(PC) + 2 if T0 = 0	Jump to specified address if T0 = 1	0 a_7	0 a_6	1 a_5	1 a_4	0 a_3	1 a_2	1 a_1	0 a_0	2	2					
	JT1 addr	$(PC_{0-7}) \leftarrow$addr if T1 = 1; (PC)$\leftarrow$(PC) + 2 if T1 = 0	Jump to specified address if T1 = 1	0 a_7	1 a_6	0 a_5	1 a_4	0 a_3	1 a_2	1 a_1	0 a_0	2	2					
	JZ addr	$(PC_{0-7}) \leftarrow$addr if A = 0; (PC)$\leftarrow$(PC) + 2 if A $\neq$ 0	Jump to specified address if A is zero	1 a_7	1 a_6	0 a_5	0 a_4	0 a_3	1 a_2	1 a_1	0 a_0	2	2					
CONTROL	EN I		Enable external ($\overline{INT}$) interrupt	0	0	0	0	0	1	0	1	1	1					
	DIS I		Disable external ($\overline{INT}$) interrupt	0	0	0	1	0	1	0	1	1	1					
	SEL RB0	(BS)$\leftarrow$0	Select bank 0 (locations 0–7) of data memory	1	1	0	0	0	1	0	1	1	1					•
	SEL RB1	(BS)$\leftarrow$1	Select bank 1 (locations 24–31) of data memory	1	1	0	1	0	1	0	1	1	1					•
	SEL MB0	(DBF)$\leftarrow$0	Select program memory bank 0; addresses 0–2047	1	1	1	0	0	1	0	1	1	1					
	SEL MB1	(DBF)$\leftarrow$1	Select program memory bank 1; addresses 2048–4095	1	1	1	1	0	1	0	1	1	1					
	ENT0 CLK		Enable clock output onto T0	0	1	1	1	0	1	0	1	1	1					

Table 2 (continued)

	mnemonic	function	description	instruction code D_7 D_6 D_5 D_4 D_3 D_2 D_1 D_0	cycles	bytes	flags C	AC	F0	F1	BS
DATA MOVES	MOV A,#data	(A)←data	Move immediate data into A	0 0 1 0 0 0 1 1 d_7 d_6 d_5 d_4 d_3 d_2 d_1 d_0	2	2					
	MOV A,Rr	(A)←(Rr) for r = 0–7	Move the contents of the designated register into A	1 1 1 1 1 r r r	1	1					
	MOV A,@Rr	(A)←((R)) for r = 0–1	Move indirect the contents of data memory into A	1 1 1 1 0 0 0 r	1	1					
	MOV A,PSW	(A)←(PSW)	Move contents of the program status word into A	1 1 0 0 0 1 1 1	1	1					
	MOV Rr,#data	(Rr)←data for r = 0–7	Move immediate data into the designated register	1 0 1 1 1 r r r d_7 d_6 d_5 d_4 d_3 d_2 d_1 d_0	2	2					
	MOV Rr,A	(Rr)←(A) for r = 0–7	Move A contents into the designated register	1 0 1 0 1 r r r	1	1					
	MOV @Rr,A	((Rr))←(A) for r = 0–1	Move indirect A contents into data memory location	1 0 1 0 0 0 0 r	1	1					
	MOV @Rr,#data	((Rr))←data for r = 0–1	Move indirect the specified data into data memory	1 0 1 1 0 0 0 r d_7 d_6 d_5 d_4 d_3 d_2 d_1 d_0	2	2					
	MOV PSW,A	(PSW)←A	Move contents of A into the program status word	1 1 0 1 0 1 1 1	1	1	•	•	•		•
	MOVP A,@A	(A)←((A))	Move data in the current page into A	1 0 1 0 0 0 1 1	2	1					
	MOVP3 A,@A	(A)←((A)) in page 3	Move data in page 3 of memory bank Ø into A	1 1 1 0 0 0 1 1	2	1					
	MOVX A,@Rr	(A)←((R)) for r = 0–1	Move indirect the contents of external memory location into A	1 0 0 0 0 0 0 r	2	1					
	MOVX @Rr,A	((Rr))←(A) for r = 0–1	Move indirect the contents of A into external memory	1 0 0 1 0 0 0 r	2	1					
	XCH A,Rr	(A)↔(Rr) for r = 0–7	Exchange A with designated register contents	0 0 1 0 1 r r r	1	1					
	XCH A,@Rr	(A)↔((R)) for r = 0–1	Exchange indirect A contents with location in data memory	0 0 1 0 0 0 0 r	1	1					

D.M.	XCHD A,@Rr	$(A_{0-3}) \leftrightarrow (Rr_{0-3})$ for r = 0–1	Exchange indirect 4-bit contents of A with data memory	0	0	1	1	0	0	0	r	1	1					
FLAGS	CPL C	(C)←NOT (C)	Complement content of carry bit	1	0	1	0	0	1	1	1	1	1	•				
	CPL F0	(F0)←NOT (F0)	Complement content of flag F0	1	0	0	1	0	1	0	1	1	1			•		
	CPL F1	(F1)←NOT (F1)	Complement content of flag F1	1	0	1	1	0	1	0	1	1	1				•	
	CLR C	(C)←0	Clear content of carry bit to 0	1	0	0	1	0	1	1	1	1	1	•				
	CLR F0	(F0)←0	Clear content of flag F0 to 0	1	0	0	0	0	1	0	1	1	1			•		
	CLR F1	(F1)←0	Clear content of flag F1 to 0	1	0	1	0	0	1	0	1	1	1				•	
INPUT/OUTPUT	ANL BUS,#data	(BUS)←(BUS) AND data	Logical AND immediate data with BUS	1 d_7	0 d_6	0 d_5	1 d_4	1 d_3	0 d_2	0 d_1	0 d_0	2	2					
	ANL Pp,#data	(Pp)←(Pp) AND data; p = 1–2	Logical AND immediate data with designated port (1 or 2)	1 d_7	0 d_6	0 d_5	1 d_4	1 d_3	0 d_2	p d_1	p d_0	2	2					
	ANLD Pp,A	(Pp)←(Pp) AND (A_{0-3}); p = 4–7	Logical AND contents of A with designated port (4–7)	1	0	0	1	1	1	p	p	2	1					
	IN A,Pp	(A)←(Pp) p = 1–2	Input data from designated port (1–2) into A	0	0	0	0	1	0	p	p	2	1					
	INS A,BUS	(A)←(BUS)	Input strobed BUS data into A	0	0	0	0	1	0	0	0	2	1					
	MOVD A,Pp	(A_{0-3})←(Pp); p = 4–7 (A_{4-7})←0	Move contents of designated port (4–7) into A	0	0	0	0	1	1	p	p	2	1					
	MOVD Pp,A	(Pp)←(A_{0-3}) p = 4–7	Move contents of A to designated port (4–7)	0	0	1	1	1	1	p	p	1	1					
	ORLD Pp,A	(Pp)←(Pp) OR (A_{0-3}); p = 4–7	Logical OR contents of A with designated port (4–7)	1	0	0	0	1	1	p	p	1	1					
	ORL BUS,#data	(BUS)←(BUS) OR data	Logical OR immediate data with BUS	1 d_7	0 d_6	0 d_5	0 d_4	1 d_3	0 d_2	0 d_1	0 d_0	2	2					
	ORL Pp,#data	(Pp)←(Pp) OR data p = 1–2	Logical OR immediate data with designated port (1–2)	1 d_7	0 d_6	0 d_5	0 d_4	1 d_3	0 d_2	p d_1	p d_0	2	2					
	OUTL BUS,A	(BUS)←(A)	Output contents A onto BUS	0	0	0	0	0	0	1	0	2	1					
	OUTL Pp,A	(Pp)←(A) p = 1–2	Output contents A to designated port (1–2)	0	0	1	1	1	0	p	p	2	1					

Table 2 (continued)

	mnemonic	function	description	instruction code D_7 D_6 D_5 D_4 D_3 D_2 D_1 D_0	cycles	bytes	flags C	AC	F0	F1	BS
REGISTER	DEC Rr	(Rr)←(Rr) −1 for r = 0–7	Decrement contents of designated register by 1	1 1 0 0 1 r r r	1	1					
	INC Rr	(Rr)←(Rr) + 1 for r = 0–7	Increment contents of designated register by 1	0 0 0 1 1 r r r	1	1					
	INC @Rr	((R))←((Rr)) + 1 for r = 0–1	Increment indirect the contents of data memory location by 1	0 0 0 1 0 0 0 r	1	1					
SUBROUTINE	CALL addr	((SP))←(PC), (PSW_{4-7}) (SP)←(SP) + 1 (PC_{8-10})←$addr_{8-10}$ (PC_{0-7})←$addr_{0-7}$	Call designated subroutine	a_{10} a_9 a_8 1 0 1 0 0 / a_7 a_6 a_5 a_4 a_3 a_2 a_1 a_0	2	2					
	RET	(SP)←(SP) −1 (PC)←((SP))	Return from subroutine without restoring program status word	1 0 0 0 0 0 1 1	2	1					
	RETR	(SP)←(SP) −1 (PC)←((SP)) (PSW_{4-7})←((SP))	Return from subroutine restoring program status word	1 0 0 1 0 0 1 1	2	1					
TIMER/COUNTER	EN TCNTI		Enable timer/counter interrupt	0 0 1 0 0 1 0 1	1	1					
	DIS TCNTI		Disable timer/counter interrupt	0 0 1 1 0 1 0 1	1	1					
	MOV A,T	(A)←(T)	Move contents of timer/counter into A	0 1 0 0 0 0 1 0	1	1					
	MOV T,A	(T)←(A)	Move contents of A into timer/counter	0 1 1 0 0 0 1 0	1	1					
	STOP TCNT		Stop count for event counter or timer	0 1 1 0 0 1 0 1	1	1					
	STRT CNT		Start count for event counter	0 1 0 0 0 1 0 1	1	1					
	STRT T		Start count for timer	0 1 0 1 0 1 0 1	1	1					
	IDLE		Entering IDLE mode	0 0 0 0 0 0 0 1	1	1					
	NOP		No operation	0 0 0 0 0 0 0 0	1	1					

Table 3 Instruction timing (see also Figs 11 and 12)

	cycle 1					cycle 2				
instruction	S1	S2	S3	S4	S5	S1	S2	S3	S4	S5
IN A,P	fetch instruction	increment program counter	–	increment timer	–	–	read port	* –	–	–
OUTL P,A	|	|	–	|	output to port	–	–	* –	–	–
ANL P,#data	|	* |	–	|	read port	fetch immediate data	–	*increment program counter	output to port	–
ORL P,#data	|	* |	–	|	read port	fetch immediate data	–	*increment program counter	output to port	–
INS A,BUS	|	|	–	|	–	–	read port	* –	–	–
OUTL BUS,A	|	|	–	|	output to port	–	–	* –	–	–
ANL BUS,#data	|	* |	–	|	read port	fetch immediate data	–	*increment program counter	output to port	–
ORL BUS,#data	|	* |	–	|	read port	fetch immediate data	–	*increment program counter	output to port	–
MOVX @R,A	|	|	output RAM address	|	output data to RAM	–	–	* –	–	–
MOVX A,@R	|	|	output RAM address	|	–	–	read data	* –	–	–
MOVD A,P	| fetch instruction	increment program counter	output opcode/ address	| increment timer	–	–	read P2 lower	* –	–	–

Table 3 Instruction timing (continued)

	cycle 1					cycle 2				
instruction	S1	S2	S3	S4	S5	S1	S2	S3	S4	S5
MOVD P,A	fetch instruction	increment program counter	output opcode/ address	increment timer	output data to P2 lower	–	–	* –	–	–
ANLD P,A	|		output opcode/ address	|	output data	–	–	* –	–	–
ORLD P,A	|		output opcode/ address	|	output data	–	–	* –	–	–
J (conditional)	|	*	sample condition	increment timer	–	fetch immediate data	–	*update program counter	–	–
STRT CNT STRT T	|	*	–	–	start counter					
STOP TCNT	|	*	–	–	stop counter					
EN I	|	*	–	enable interrupt	–					
DIS I	|	*	–	disable interrupt	–					
ENT0 CLK	fetch instruction	*increment program counter	–	enable clock	–					

* Valid instruction addresses are output at this time if external program memory is being accessed.

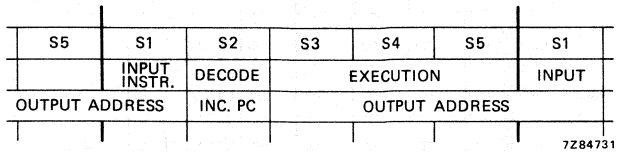


Fig. 11 Instruction cycle.

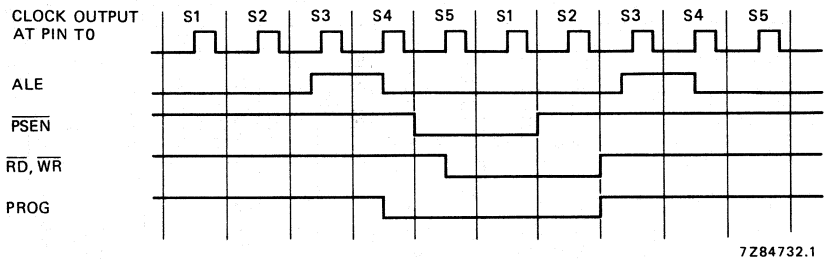


Fig. 12 Instruction cycle timing.

DEVELOPMENT DATA

Table 4 Instruction map.

first hexadicimal character of opcode	second hexadicimal character of opcode														
	0 1	2	3	4	5	6	7	8	9 1 2	A	B	C 4	D 5	E 6	F 7
0	NOP IDLE mode	OUTL BUS,A	ADD A, #data	JMP page 0	EN I		DEC A	INS A,BUS	IN A, Pp			MOVD A,Pp			
1	INC @Rr 0 1	JB0 addr	ADDC A, #data	CALL page 0	DIS I	JTF addr	INC A	INC Rr 0 1 2 3 4 5 6 7							
2	XCH A, @Rr 0 1		MOV A, #data	JMP page 1	EN TCNTI	JNT0 addr	CLR A	XCH A,Rr 0 1 2 3 4 5 6 7							
3	XCHD A, @Rr 0 1	JB1 addr		CALL page 1	DIS TCNTI	JT0 addr	CPL A		OUTL Pp,A 1 2			MOVD Pp,A 4 5 6 7			
4	ORL A, @Rr 0 1	MOV A, T	ORL A, #data	JMP page 2	STRT CNT	JNT1 addr	SWAP A	ORL A,Rr 0 1 2 3 4 5 6 7							
5	ANL A, @Rr 0 1	JB2 addr	ANL A, #data	CALL page 2	STRT T	JT1 addr	DA, A	ANL A,Rr 0 1 2 3 4 5 6 7							
6	ADD A, @Rr 0 1	MOV T,A		JMP page 3	STOP TCNT		RRC A	ADD A,Rr 0 1 2 3 4 5 6 7							
7	ADDC A, @Rr 0 1	JB3 addr		CALL page 3	ENT0 CLK	JF1 addr	RR A	ADDC A,Rr 0 1 2 3 4 5 6 7							
8	MOVX A, @Rr 0 1		RET	JMP page 4	CLR F0	JN1 addr		ORL BUS, #data	ORL Pp, #data 1 2			ORLD Pp,A 4 5 6 7			
9	MOVX @Rr,A 0 1	JB4 addr	RETR	CALL page 4	CPL F0	JNZ addr	CLR C	ANL BUS, #data	ANP Pp, #data 1 2			ANLD Pp,A 4 5 6 7			
A	MOV @Rr, A 0 1		MOVP A, @A	JMP page 5	CLR F1		CPL C	MOV Rr,A 0 1 2 3 4 5 6 7							
B	MOV @Rr, #data 0 1	JB5 addr	JMPP @A	CALL page 5	CPL F1	JF0 addr		MOV R, #data 0 1 2 3 4 5 6 7							
C				JMP page 6	SEL RB0	JZ addr	MOV A, PSW	DEC Rr 0 1 2 3 4 5 6 7							
D	XRL A, @Rr 0 1	JB6 addr	XRL A, #data	CALL page 6	SEL RB1		MOV PSW, A	XRL A,Rr 0 1 2 3 4 5 6 7							
E			MOVP3 A @A	JMP page 7	SEL MB0	JNC addr	RL A	DJNZ Rr, addr 0 1 2 3 4 5 6 7							
F	MOV A, @Rr 0 1	JB7 addr		CALL page 7	SEL MB1	JC addr	RLC A	MOV A,Rr 0 1 2 3 4 5 6 7							

RATINGS

Limiting values in accordance with the Absolute Maximum System (IEC 134)

parameter	symbol	min.	max.	unit
Input, output current on any single pin	I_I, I_O	–	± 10	mA
Total power dissipation	P_{tot}	–	0.5	W
Storage temperature range	T_{stg}	−65	+ 150	°C
Operating ambient temperature range				
PCB80C39/49 version	T_{amb}	0	+ 70	°C
PCF80C39/49 version	T_{amb}	−40	+ 85	°C
PCA80C39/49 version	T_{amb}	−40	+ 110	°C

DC CHARACTERISTICS

V_{DD} = 5 V (± 10%); V_{SS} = 0 V; T_{amb} = 0 to + 70 °C (PCB80C39/49); −40 to + 85 °C (PCF80C39/49); −40 to + 110 °C (PCA80C39/49). All voltages with respect to V_{SS} unless otherwise specified.

parameter	conditions	symbol	min.	max.	unit
Supply voltage		V_{DD}	4.5	5.5	V
Supply current					
operating	f_{CLK} = 15 MHz	I_{DD}	–	15	mA
	f_{CLK} = 11 MHz	I_{DD}	–	11.5	mA
	f_{CLK} = 6 MHz	I_{DD}	–	7.3	mA
	f_{CLK} = 1 MHz	I_{DD}	–	3	mA
IDLE mode	f_{CLK} = 15 MHz	I_{ID}	–	5	mA
	f_{CLK} = 11 MHz	I_{ID}	–	4.6	mA
	f_{CLK} = 6 MHz	I_{ID}	–	2.9	mA
	f_{CLK} = 1 MHz	I_{ID}	–	1.2	mA
power down mode	V_{DD} = 2 V; $\overline{RESET}$ = LOW	I_{PD}	–	2	μA
Inputs					
Input voltage LOW all inputs except $\overline{RESET}$; XTAL1; XTAL2		V_{IL}	−0.5	0.18 V_{DD}	V

DEVELOPMENT DATA

DC CHARACTERISTICS (continued)

parameter	conditions	symbol	min.	max.	unit
Input voltage LOW $\overline{RESET}$; XTAL1; XTAL2		V_{IL1}	−0.5	0.13 V_{DD}	V
Input voltage HIGH all inputs except $\overline{RESET}$; XTAL1; XTAL2	(PCB80C39/49 version)	V_{IH}	0.4 V_{DD}	V_{DD}	V
	(PCF/PCA80C39/49 version note 1)	V_{IH}	0.43 V_{DD}	V_{DD}	V
Input voltage HIGH $\overline{RESET}$; XTAL1; XTAL2		V_{IH1}	0.7 V_{DD}	V_{DD}	V
Outputs					
Output voltage LOW BUS	I_{OL} = 2 mA	V_{OL}	–	0.45	V
Output voltage LOW $\overline{RD}$; $\overline{WR}$; $\overline{PSEN}$; ALE	I_{OL} = 1.8 mA	V_{OL1}	–	0.45	V
Output voltage LOW $\overline{PROG}$	I_{OL} = 1 mA	V_{OL2}	–	0.45	V
Output voltage LOW all other outputs	I_{OL} = 1.6 mA	V_{OL3}	–	0.45	V
Output voltage HIGH BUS	$-I_{OH}$ = 400 μA	V_{OH}	0.75 V_{DD}	–	V
Output voltage HIGH $\overline{RD}$; $\overline{WR}$; $\overline{PSEN}$; ALE	$-I_{OH}$ = 100 μA	V_{OH1}	0.75 V_{DD}	–	V
Output voltage HIGH all other outputs	$-I_{OH}$ = 40 μA	V_{OH2}	0.75 V_{DD}	–	V
Input leakage current $\overline{INT}$; T1; EA	without internal pull-up $V_{SS} < V_I < V_{DD}$	$\pm I_{IL}$	–	10	μA
Input leakage current P10-P17; P20-P27; $\overline{SS}$	(PCB80C39/49 version) $V_{SS}+0.45 < V_I < V_{DD}$	$-I_{IL1}$	–	500	μA
	(PCF/PCA80C39/49 version note 1). With internal pull-up. $V_{SS}+0.45 < V_I < V_{DD}$	$-I_{IL1}$	–	600	μA

parameter	conditions	symbol	min.	max.	unit
Input leakage current $\overline{RESET}$	(PCB80C39/49 version) $V_{SS} < V_I < V_{IL1}$	$-I_{ILR}$	20	300	μA
	(PCF/PCA80C39/49 version note 1) $V_{SS} < V_I < V_{IL1}$	$-I_{ILR}$	20	350	μA
Output leakage current BUS; TO at high impedance state	$V_{SS}+0.45 < V_I < V_{CC}$	$\pm I_{OL}$	–	10	μA

Note to the DC characteristics

1. TTL levels are not fully compatible.

DEVELOPMENT DATA

AC CHARACTERISTICS

$V_{CC} = V_{DD} = 5\ V \pm 10\%$; $V_{SS} = 0\ V$; T_{amb} (PCB version) = 0 to + 70 °C; note 1;
T_{amb} (PCF version) = −40 to + 85 °C; note 1; T_{amb} (PCA version) = −40 to + 110 °C; note 1).

See waveforms Figs 13, 14, 15, 16 and 17.

parameter	f (t_{CL}) (note 2)	symbol	11 MHz		unit
			min.	max.	
Clock period (note 2)	$1/(f_{XTAL})$	t_{CL}	90.9	1000	ns
ALE pulse width	$7/30t_{CL}-170$	t_{LL}	150	–	ns
Address set-up time to ALE	$1/5t_{CL}-110$	t_{AL}	160	–	ns
Address hold time from ALE	$1/15t_{CL}-40$	t_{LA}	50	–	ns
Control pulse width $\overline{RD}$, $\overline{WR}$	$1/2t_{CL}-200$	t_{CC1}	480	–	ns
Control pulse width $\overline{PSEN}$	$2/5t_{CL}-200$	t_{CC2}	350	–	ns
Data set-up time before $\overline{WR}$	$13/30t_{CL}-200$	t_{DW}	390	–	ns
Data hold time after $\overline{WR}$ (note 3)	$1/15t_{CL}-50$	t_{WD}	40	–	ns
Data hold time $\overline{RD}$, $\overline{PSEN}$	$1/10t_{CL}-30$	t_{DR}	0	110	ns
$\overline{RD}$ to data input	$2/5t_{CL}-170$	t_{RD1}	–	375	ns
$\overline{PSEN}$ to data input	$3/10t_{CL}-170$	t_{RD2}	–	240	ns
Address set-up time to $\overline{WR}$	$1/3t_{CL}-150$	t_{AW}	300	–	ns
Address set-up time to data input ($\overline{RD}$)	$7/10t_{CL}-250$	t_{AD1}	–	730	ns
Address set-up time to data input ($\overline{PSEN}$)	$1/2t_{CL}-220$	t_{AD2}	–	460	ns
Address floating to $\overline{RD}$, $\overline{WR}$	$2/15t_{CL}-40$	t_{AFC1}	140	–	ns
Address floating to $\overline{PSEN}$	$1/30t_{CL}-40$	t_{AFC2}	10	–	ns
ALE to control pulse $\overline{RD}$, $\overline{WR}$	$1/5t_{CL}-75$	t_{LAFC1}	200	–	ns
ALE to control pulse $\overline{PSEN}$	$1/10t_{CL}-75$	t_{LAFC2}	60	–	ns
Control pulse to ALE $\overline{RD}$, $\overline{WR}$, $\overline{PROG}$	$1/15t_{CL}-40$	t_{CA1}	50	–	ns
Control pulse to ALE $\overline{PSEN}$	$4/15t_{CL}-40$	t_{CA2}	320	–	ns
Port control set-up to $\overline{PROG}$	$1/10t_{CL}-80$	t_{CP}	50	–	ns

DEVELOPMENT DATA

AC CHARACTERISTICS (continued)

parameter	f (t_{CL})	symbol	11 MHz		unit
			min.	max.	
Port control hold to $\overline{PROG}$	$4/15t_{CL}-260$	t_{PC}	100	–	ns
$\overline{PROG}$ to time port 2 input must be valid	$17/30t_{CL}-120$	t_{PR}	–	650	ns
Input data hold time from $\overline{PROG}$	$1/10t_{CL}$	t_{PF}	0	140	ns
Output data set-up time	$2/5t_{CL}-290$	t_{DP}	250	–	ns
Output data hold time	$1/10t_{CL}-90$	t_{PD}	40	–	ns
$\overline{PROG}$ pulse width	$7/10t_{CL}-250$	t_{PP}	700	–	ns
Port 2 I/O data set-up time to ALE	$4/15t_{CL}-200$	t_{PL}	160	–	ns
Port 2 I/O data hold time to ALE	$1/10t_{CL}-120$	t_{LP}	15	–	ns
Port output from ALE	$3/10t_{CL}+100$	t_{PV}	–	510	ns
Cycle time	$(1/f_{XTAL}) \times 15$	t_{CY}	1,36	15	μs
T0 repetition rate	$3/15t_{CL}$	t_{0PRR}	270	–	ns
Clock period (note 2)	$1/(f_{XTAL})$	t_{CL}	98,8	1000	ns

Notes to AC characteristics

1. Control outputs: C_L = 80 pF
   Bus outputs: C_L = 150 pF.
2. f(t_{CL}) assumes 50% duty cycle on XTAL1 and XTAL2; minimum frequency = 1 MHz; maximum frequency = 15 MHz for all versions.
3. Bus high-impedance load: 20 pF.

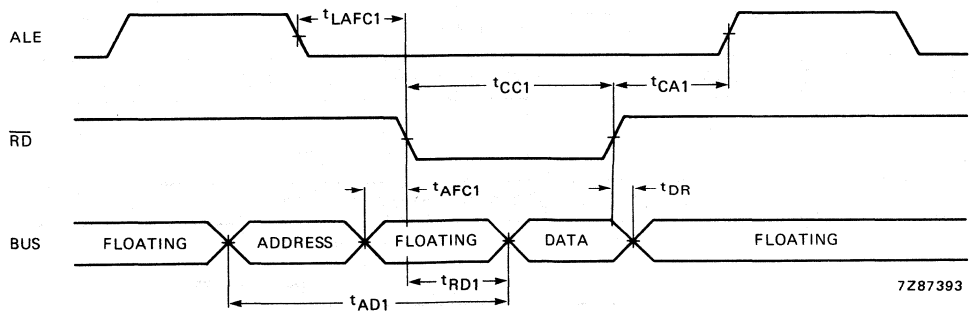


Fig. 13 Read from external data memory.

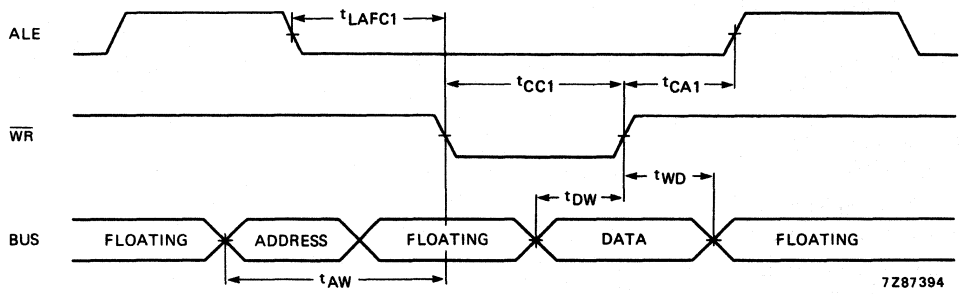


Fig. 14 Write to external data memory.

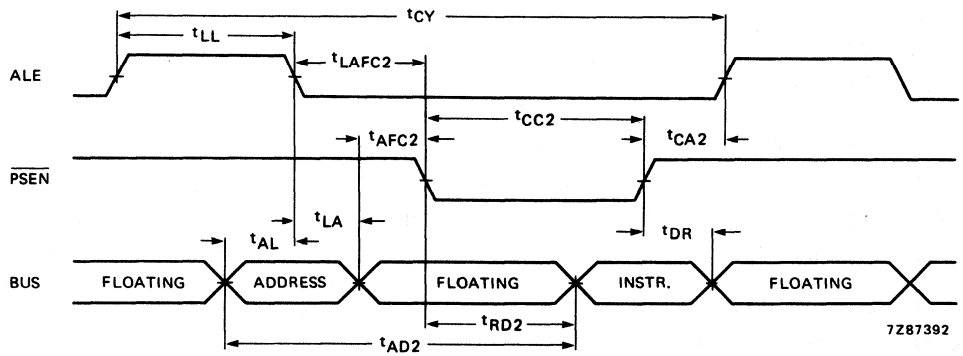


Fig. 15 Instruction fetch from external program memory.

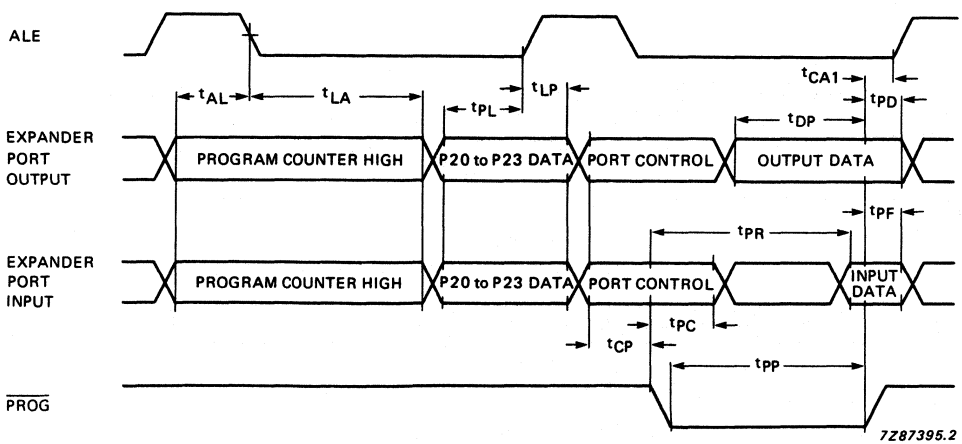


Fig. 16 Port 2 timing.

DEVELOPMENT DATA

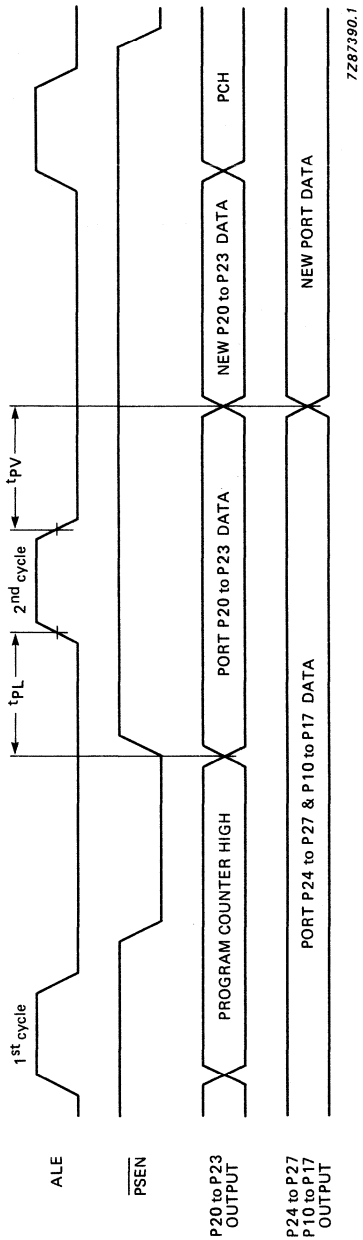


Fig. 17 I/O port timing.

SC80C451/SC83C451
I/O Expanded CMOS Microcontroller

Product Specification

Microprocessor Products

DESCRIPTION

The Signetics SC80C451/SC83C451 is an I/O expanded, single-chip microcontroller fabricated with Signetics' high-density epitaxial CMOS technology. It is offered in a 68-pin PLCC and a 64-pin DIP. The Signetics CMOS technology combines the high-speed and high-density characteristics of HMOS with the low power of CMOS. Signetics' epitaxial substrate minimizes latch-up sensitivity.

The SC80C451/SC83C451 is a functional extension of the SC80C31/SC80C51 microcontroller with three additional I/O ports and four I/O control lines. The PLCC version has a total of 68 pins. The DIP version has a total of 64 pins. The four added control lines associated with port 6 facilitate high-speed asynchronous I/O functions.

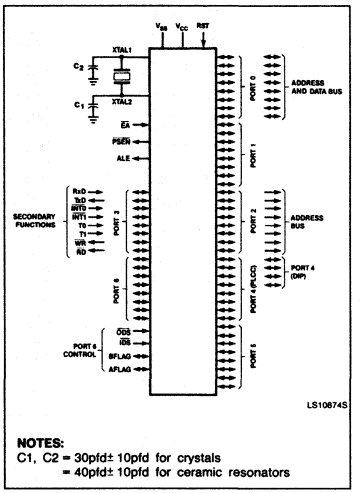
The SC80C451/SC83C451 includes a 4k × 8 ROM; a 128 × 8 RAM; 56 (PLCC) or 52 (DIP) I/O lines; two 16-bit timer/counters; a five source, two priority level, nested interrupt structure; a serial I/O port for either a full-duplex UART, I/O expansion, or multiprocessor communications; and on-chip oscillator and clock circuits.

The SC80C451/SC83C451 has two software selectable modes of reduced activity for further power reduction; idle mode and power-down mode. Idle mode freezes the CPU while allowing the RAM, timers, serial port, and interrupt system to continue functioning. Power-down mode freezes the oscillator, causing all other chip functions to be inoperative while maintaining the RAM contents.

FEATURES

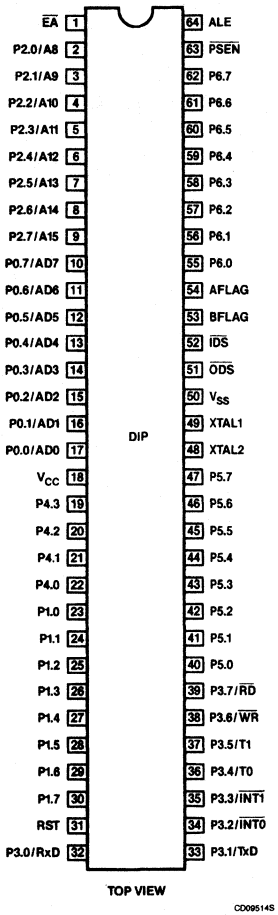
- **SC80C451 — CPU with RAM and I/O**
- **SC83C451 — 80C451 with mask-programmable ROM**
- **68-pin PLCC and 64-pin DIP packages**
  - **Seven 8-bit I/O ports (PLCC version)**
  - **Six 8-bit ports and one 4-bit port (DIP version)**
- **Port 6 features:**
  - **8 data pins**
  - **4 control pins**
  - **Direct MPU bus interface**
  - **Parallel printer interface**
- **Pin compatible with SC87C451 microcontroller**
- **On the microcontroller:**
  - **SC80C51 CPU**
  - **4k × 8 ROM (SC83C451 only)**
  - **128 × 8 RAM**
  - **Two 16-bit timer/counters**
  - **Full-duplex serial I/O**
  - **Two external interrupts**
- **External memory addressing capability:**
  - **64k ROM and 64k RAM**
- **Low power consumption:**
  - **Normal operation: 24mA typical @ 5V, 12MHz**
  - **Idle mode: 3mA typical @ 5V, 12MHz**
  - **Power-down mode: 50μA typical @ 2V**

LOGIC SYMBOL



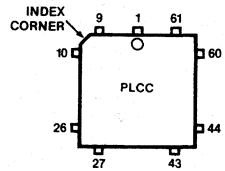
NOTES:
C1, C2 = 30pfd± 10pfd for crystals
= 40pfd± 10pfd for ceramic resonators

PIN CONFIGURATIONS



TOP VIEW

CD09514S



TOP VIEW

CD09520S

Pin	Function	Pin	Function
1	$\overline{EA}$	35	RST
2	P2.0/A8	36	P3.0/R × D
3	P2.1/A9	37	P3.1/T × D
4	P2.2/A10	38	P3.2/$\overline{INT0}$
5	P2.3/A11	39	P3.3/$\overline{INT1}$
6	P2.4/A12	40	P3.4/T0
7	P2.5/A13	41	P3.5/T1
8	P2.6/A14	42	P3.6/$\overline{WR}$
9	P2.7/A15	43	P3.7/$\overline{RD}$
10	P0.7/AD7	44	P5.0
11	P0.6/AD6	45	P5.1
12	P0.5/AD5	46	P5.2
13	P0.4/AD4	47	P5.3
14	P0.3/AD3	48	P5.4
15	P0.2/AD2	49	P5.5
16	P0.1/AD1	50	P5.6
17	P0.0/AD0	51	P5.7
18	V_{CC}	52	XTAL2
19	P4.7	53	XTAL1
20	P4.6	54	V_{SS}
21	P4.5	55	$\overline{ODS}$
22	P4.4	56	$\overline{IDS}$
23	P4.3	57	BFLAG
24	P4.2	58	AFLAG
25	P4.1	59	P6.0
26	P4.0	60	P6.1
27	P1.0	61	P6.2
28	P1.1	62	P6.3
29	P1.2	63	P6.4
30	P1.3	64	P6.5
31	P1.4	65	P6.6
32	P1.5	66	P6.7
33	P1.6	67	$\overline{PSEN}$
34	P1.7	68	ALE

ORDERING INFORMATION

SC□□ C451 A □ □ □ 6□ (CPxxxx)

ROM/RAM (bytes)
80 = EXT/128
83 = 4k/128

OPERATING TEMPERATURE RANGE
A = −40°C to +85°C
C = −0°C to +70°C

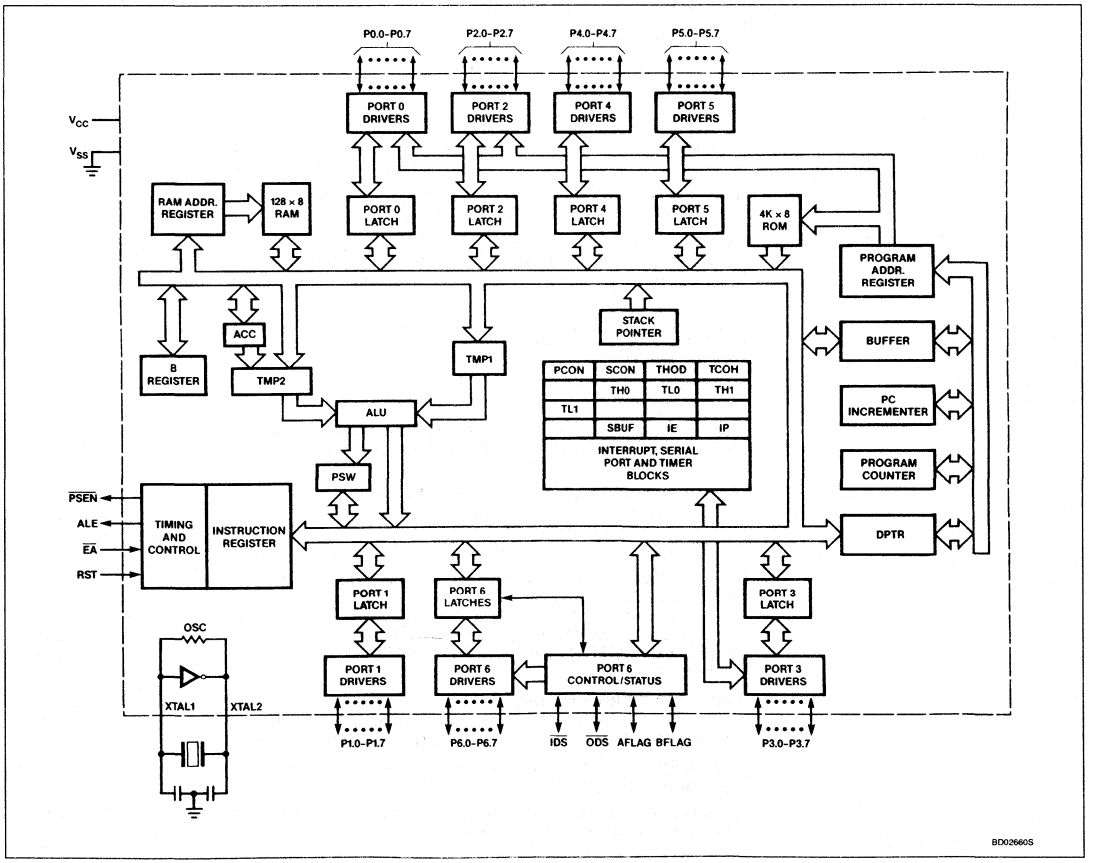
SPEED
C = 3.5 to 12MHz
G = 3.5 to 16MHz
2 = 0.5 to 12MHz

CUSTOM ROM PATTERN NUMBER
Applies to masked ROM versions only. Number will be assigned by Signetics. Contact Signetics sales office for ROM pattern submission requirements.

68-Pin LCC
64-Pin DIP

PACKAGE
N = Plastic DIP
I = Ceramic DIP
A = Plastic LCC

BLOCK DIAGRAM



PIN CONFIGURATION

MNEMONIC	PIN NO.		TYPE	NAME AND FUNCTION
	DIP	PLCC		
V_{SS}	50	54		**Ground:** 0V reference
V_{CC}	18	18		**Power Supply:** +5V
P0.0 – P0.7	17 – 10	17 – 10	I/O	**Port 0:** An 8-bit open drain, bidirectional I/O port. Port 0 is also the multiplexed low-order address and data bus during accesses to external memory, and outputs instruction bytes during program verification. External pull-ups are required during program verification. Port 0 can sink/source eight LS TTL inputs.
P1.0 – P1.7	23 – 30	27 – 34	I/O	**Port 1:** An 8-bit bidirectional I/O port with internal pull-ups. Port 1 receives the low-order address bytes during program verification. In the SC83C451, port 1 can sink/source three LS TTL inputs, and drive CMOS inputs without external pull-ups.
P2.0 – P2.7	2 – 9	2 – 9	I/O	**Port 2:** An 8-bit bidirectional I/O port with internal pull-ups. Port 2 emits the high-order address bytes during accesses to external memory and receives the high-order address bits and control signals during program verification in the SC83C451. Port 2 can sink/source three LS TTL inputs and drive CMOS inputs without external pull-ups.
P3.0 – P3.7	32 – 39	36 – 43	I/O	**Port 3:** An 8-bit bidirectional I/O port with internal pull-ups. Port 3 can sink/source three LS TTL inputs and drive CMOS inputs without external pull-ups. Port 3 also serves the SC83C451 special functions listed below:
	32	36	I	RXD (P3.0): Serial input port
	33	37	O	TXD (P3.1): Serial output port
	34	38	I	$\overline{INT0}$ (P3.2): External interrupt 0
	35	39	I	$\overline{INT1}$ (P3.3): External interrupt 1
	36	40	I	T0 (P3.4): Timer 0 external input
	37	41	I	T1 (P3.5): Timer 1 external input
	38	42	O	$\overline{WR}$ (P3.6): External data memory write strobe
	39	43	O	$\overline{RD}$ (P3.7): External data memory read strobe
P4.0 – P4.3	22 – 19		I/O	**Port 4:** A 4- or 8-bit bidirectional port with internal pull-ups. Port 4 can sink/source three LS TTL inputs and drive CMOS inputs without external pull-ups.
P4.0 – P4.7		26 – 19	I/O	
P5.0 – P5.7	40 – 47	44 – 51	I/O	**Port 5:** An 8-bit I/O port with internal pull-ups. Port 5 can sink/source three LS TTL inputs and drive CMOS inputs without external pull-ups.
P6.0 – P6.7	55 – 62	59 – 66	I/O	**Port 6:** A specialized 8-bit bidirectional I/O port with internal pull-ups. This special port can sink/source three LS TTL inputs and drive CMOS inputs without external pull-ups. Port 6 can be used in a strobed or non-strobed mode of operation, and in conjunction with four control pins that serve the functions listed below.
				Port 6 Control Lines:
$\overline{ODS}$	51	55	I	$\overline{ODS}$: Output data strobe
$\overline{IDS}$	52	56	I	$\overline{IDS}$: Input data strobe
BFLAG	53	57	I/O	BFLAG: A bidirectional I/O pin with internal pull-ups
AFLAG	54	58	I/O	AFLAG: A bidirectional I/O pin with internal pull-ups
RST	31	35	I	**RESET:** A high level on this pin for two machine cycles while the oscillator is running resets the device. An internal pull-down resistor permits power-on reset using only a capacitor connected to the V_{CC}.
ALE	64	68	O	**Address Latch Enable:** An output for latching the low byte of the address during accesses to external memory. ALE is activated at a constant rate of 1/6 the oscillator frequency except during an external data memory access, at which time one ALE is skipped. ALE can sink/source eight LS TTL inputs and drive CMOS inputs without an external pull-up.
$\overline{PSEN}$	63	67	O	**Program Store Enable:** This output is the read strobe to external program memory. $\overline{PSEN}$ is activated twice each machine cycle during fetches from external program memory; however, when executing out of external program memory, two activations of $\overline{PSEN}$ are skipped during each access to external data memory. $\overline{PSEN}$ is not activated during fetches from internal program memory. $\overline{PSEN}$ can sink/source eight LS TTL inputs and drive CMOS inputs without an external pull-up.
$\overline{EA}$	1	1	I	**Instruction Execution Control:** When $\overline{EA}$ is held high, the CPU executes out of internal program memory, unless the program counter exceeds OFFFH. When $\overline{EA}$ is held low, the CPU executes out of external program memory. $\overline{EA}$ must never be allowed to float.
XTAL1	49	53	I	**Crystal 1:** An input to the inverting amplifier that forms the oscillator. This input receives the external oscillator when an external oscillator is used.
XTAL2	48	52	O	**Crystal 2:** An output of the inverting amplifier that forms the oscillator. This pin should be floated when an external oscillator is used.

PORTS 4 and 5

Ports 4 and 5 are bidirectional I/O ports with internal pull-ups. Port 4 is an 8-bit port (PLCC version) or a 4-bit port (DIP Version). Port 4 and port 5 pins, with ones written to them, are pulled high by the internal pull-ups, and in that state can be used as inputs. Ports 4 and 5 are addressed at the special function register addresses shown in Table 1.

PORT 6

Port 6 is a special 8-bit bidirectional I/O port with internal pull-ups. (See Figure 1.) This port can be used as a standard I/O port, or in strobed modes of operation in conjunction with four special control lines: $\overline{ODS}$, $\overline{IDS}$, AFLAG, and BFLAG. Port 6 operating modes are controlled by the port 6 control status register (CSR). Port 6 and the CSR are addressed at the special function register addresses shown in Table 1. The following four control pins are used in conjunction with port 6:

$\overline{ODS}$: Output data strobe input for port 6. $\overline{ODS}$ can be programmed to control the port 6 output drivers and the output buffer full flag (OBF), or to clear only the OBF flag bit in the CSR (output-always mode). $\overline{ODS}$ is active low for output driver control. The OBF flag can be programmed to be cleared on the negative or positive edge of $\overline{ODS}$.

$\overline{IDS}$: Input data strobe input for port 6. $\overline{IDS}$ is used to control the port 6 input latch and input buffer full flag (IBF) bit in the CSR. The input data latch can be programmed to be transparent when the $\overline{IDS}$ is low and latched on the positive transition of $\overline{IDS}$, or to latch only on the positive transition of $\overline{IDS}$. Correspondingly, the IBF flag is set on the negative or positive transition of $\overline{IDS}$.

AFLAG: A bidirectional I/O pin. AFLAG can be programmed to be an output set high or low under program control, or to output the state of the output buffer full flag. AFLAG can also be programmed to be an input which selects whether the contents of the output buffer, or the contents of the port 6 control status register will be output on port 6. This feature grants complete port 6 status to external devices.

BFLAG: A bidirectional I/O pin. BFLAG can be programmed to be an output, set high or low under program control, or to output the state of the input buffer full flag. BFLAG can also be programmed to input an enable signal for port 6. When BFLAG is used as an enable input, port 6 output drivers are in the high impedance state, and the input latch does not respond to the $\overline{IDS}$ strobe when BFLAG is high. Both features are enabled when BFLAG is low. This feature facilitates the use of the SC83C451 in bused multiprocessor systems.

CONTROL STATUS REGISTER

The control status register (CSR) establishes the mode of operation for port 6 and indicates the current status of the port 6 I/O registers. All control status register bits can be read and written by the CPU, except bits 0 and 1, which are read only. Reset writes ones to bits 2 through 7, and writes zeros to bits 0 and 1.

CSR.0 — Input Buffer Full Flag (IBF) (Read only)
The IBF bit is set to a logic 1 when port 6 data is loaded into the input buffer under control of $\overline{IDS}$. This can occur on the negative or positive edge of $\overline{IDS}$, as determined by CSR.2. IBF is cleared when the CPU reads the input buffer register.

CSR.1 — Output Buffer Full Flag (OBF) (Read only)
The OBF flag is set to a logic 1 when the CPU writes to the port 6 output data buffer. OBF is cleared by the positive or negative edge of $\overline{ODS}$, as determined by CSR.3.

CSR.2 — IDS Mode Select (IDSM)
When CSR.2 = 0, a low-to-high transition on the $\overline{IDS}$ pin sets the IBF flag. The port 6 input buffer is loaded on the $\overline{IDS}$ positive edge. When CSR.2 = 1, a high-to-low transition on the $\overline{IDS}$ pin sets the IBF flag. The port 6 input buffer is transparent when $\overline{IDS}$ is low, and latched when $\overline{IDS}$ is high.

CSR.3 — Output Buffer Full Flag Clear Mode (OBFC)
When CSR.3 = 1, the positive edge of the $\overline{ODS}$ input clears the OBF flag. When CSR.3 = 0, the negative edge of the $\overline{ODS}$ input clears the OBF flag.

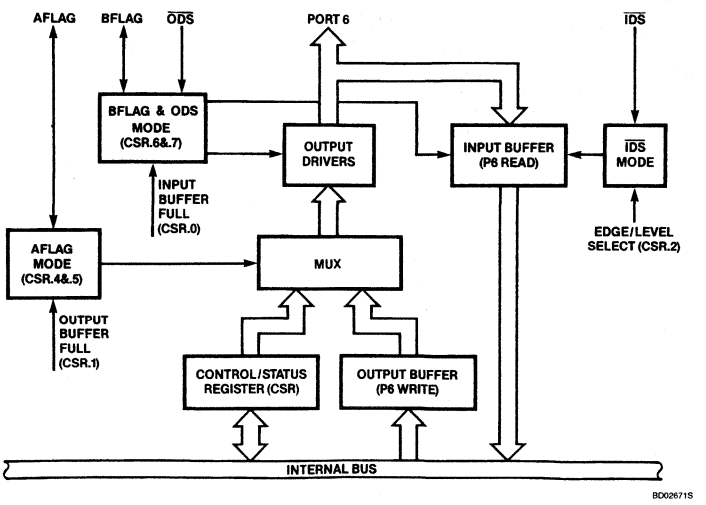


Figure 1. Port 6 Block Diagram

CSR.4, CSR.5 — AFLAG Mode Select (MA0, MA1)
Bits 4 and 5 select the mode of operation for the AFLAG pin, as follows:

MA1	MA0	AFLAG Function
0	0	Logic 0 output
0	1	Logic 1 output
1	0	OBF flag output (CSR.1)
1	1	Select (SEL) input mode

The select (SEL) input mode is used to determine whether the port 6 data register or the control status register is output on port 6. When the select feature is enabled, the AFLAG input controls the source of port 6 output data. A logic 0 on AFLAG input selects the port 6 data register, and a logic 1 on AFLAG input selects the control status register.

CSR.6, CSR.7 — BFLAG Mode Select (MB0, MB1)
Bits 6 and 7 select the mode of operation for the BFLAG pin, as follows:

MB1	MB0	BFLAG Function
0	0	Logic 0 output
0	1	Logic 1 output
1	0	IBF flag output (CSR.0)
1	1	Port enable ($\overline{PE}$)

In the port enable mode, $\overline{IDS}$ and $\overline{ODS}$ inputs are disabled when BFLAG input is high. When the BFLAG input is low, the port is enabled for I/O.

CONTROL STATUS REGISTER

(CSR)	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
	MB1	MB0	MA1	MA0	OBFC	IDSM	OBF	IBF
	BFLAG mode select		AFLAG mode select		Output buffer flag clear mode	Input data strobe mode	Output buffer full flag	Input buffer full flag
	0/0 = Logic 0 output* 0/1 = Logic 1 output* 1/0 = IBF output 1/1 = $\overline{PE}$ input (0 = Select) (1 = disable I/O)		0/0 = Logic 0 output 0/1 = Logic 1 output 1/0 = OBF output* 1/1 = SEL input (0 = Data) (1 = Control/status)		0 = Negative edge of $\overline{ODS}$ 1 = Positive edge of $\overline{ODS}$	0 = Positive edge of $\overline{IDS}$ 1 = Low level of $\overline{IDS}$	0 = Output data buffer empty 1 = Output data buffer full	0 = Input data buffer empty 1 = Input data buffer full

*Output-always mode: MB1 = 0, MA1 = 1, and MA0 = 0. In this mode port 6 is always enabled for output. $\overline{ODS}$ only clears the OBF flag.

SPECIAL FUNCTION REGISTER ADDRESSES

Special function register addresses for the SC83C451 are identical to those of the SC80C51, except for the additional registers listed in Table 1.

Table 1. Special Function Register Addresses

REGISTER ADDRESS			BIT ADDRESS							
Name	Symbol	Address	MSB							LSB
Port 4	P4	C0	C7	C6	C5	C4	C3	C2	C1	C0
Port 5	P5	C8	CF	CE	CD	CC	CB	CA	C9	C8
Port 6 data	P6	D8	DF	DE	DD	DC	DB	DA	D9	D8
Port 6 control status	CSR	E8	EF	EE	ED	EC	EB	EA	E9	E8

ABSOLUTE MAXIMUM RATINGS[1]

SYMBOL	PARAMETER	RATING	UNIT
T_A	Operating ambient temperature range	0 to +70	°C
T_{STG}	Storage temperature range	−65 to +150	°C
	Voltage from V_{CC} to V_{SS} [2]	−0.5 to +6.5	V
	Voltage from any pin to V_{SS} [2]	−0.5 to V_{CC} + 0.5	V
PD	Power dissipation	200	mW

DC ELECTRICAL CHARACTERISTICS

SYMBOL	PARAMETER	TEST CONDITIONS	LIMITS		UNIT
			Min	Max	
V_{IL}	Input low voltage, except $\overline{EA}$		−0.5	0.2 V_{CC} −.1	V
V_{IL1}	Input low voltage, $\overline{EA}$		−0.5	0.2 V_{CC} −.3	V
V_{IH}	Input high voltage, except XTAL1, RST		0.2V_{CC} +.9	V_{CC} +0.5	V
V_{IH1}	Input high voltage, XTAL1, RST		0.7V_{CC}	V_{CC} +0.5	V
V_{OL}	Output low voltage, ports 1,2,3,4,5,6, AFLAG, BFLAG	I_{OL} = 1.6mA[4]		0.45	V
V_{OL1}	Output low voltage, port 0, ALE, $\overline{PSEN}$	I_{OL} = 3.2mA[4]		0.45	V
V_{OH}	Output high voltage, ports 1,2,3,4,5,6, AFLAG, BFLAG	I_{OH} = −60μA, V_{CC} = 5V ±10%	2.4		V
		I_{OH} = −25μA	0.75V_{CC}		V
		I_{OH} = −10μA	0.9V_{CC}		V
V_{OH1}	Output high voltage, port 0 in external bus mode, ALE, $\overline{PSEN}$	I_{OH} = −400μA, V_{CC} = 5V ±10%	2.4		V
		I_{OH} = −150μA	0.75V_{CC}		V
		I_{OH} = −40μA[5]	0.9V_{CC}		V
I_{IL}	Logical 0 input current, ports 1,2,3,4,5,6, AFLAG, BFLAG	V_{IN} = 0.45V		−50	μA
I_{TL}	Logical 1 to 0 transition current, ports 1,2,3,4,5,6, AFLAG, BFLAG	V_{IN} = 2V		−650	μA
I_{LI}	Input leakage, current port 0, $\overline{EA}$, $\overline{IDS}$, $\overline{ODS}$	0.45 < V_{IN} < V_{CC}		+10	μA
R_{RST}	Reset pulldown resistor		50	150	kΩ
C_{IO}	Pin capacitance	Test freq = 1MHz, T_A = 25°C		10	pF
I_{PD}	Power-down current	V_{CC} = 2 to 6V[6]		50	μA

MAXIMUM I_{CC} (mA)

	OPERATING			IDLE		
Freq.	4V	5V	6V	4V	5V	6V
0.5MHz	1.5	2.5	3.0	1.0	2.0	2.5
3.5MHz	5.0	7.0	9.0	1.5	2.5	3.5
8.0MHz	11.0	14.5	18.0	2.0	3.5	4.5
12MHz	15.0	20.0	25.0	2.5	4.0	5.0
16MHz	20.0	25.0	30.0	3.5	5.5	7.0

AC ELECTRICAL CHARACTERISTICS

T_A = 0°C to +70°C, V_{CC} = 5V ±20%, V_{SS} = 0V, load capcitance for port 0, ALE, and $\overline{PSEN}$ = 100pF, load capacitance for all other outputs = 80 pF_3

SYMBOL	PARAMETER	12MHz CLOCK		VARIABLE CLOCK (f = 1/t_{CLCL})		
		Min	Max	Min	Max	Unit
	Speed range (see ordering code)					
1/t_{CLCL}	0			3.5	12	MHz
	1			3.5	16	
	2			0.5	12	
t_{LHLL}	ALE pulse width	127		$2t_{CLCL}$ – 40		ns
t_{AVLL}	Address valid to ALE low	28		t_{CLCL} – 55		ns
t_{LLAX}	Address hold after ALE low	48		t_{CLCL} – 35		ns
t_{LLIV}	ALE low to valid instr in		234		$4t_{CLCL}$ – 100	ns
t_{LLPL}	ALE low to $\overline{PSEN}$ low	43		t_{CLCL} – 40		ns
t_{PLPH}	$\overline{PSEN}$ pulse width	205		$3t_{CLCL}$ – 45		ns
t_{PLIV}	$\overline{PSEN}$ low to valid instr in		145		$3t_{CLCL}$ – 105	ns
t_{PXIX}	Input instr hold after $\overline{PSEN}$	0		0		ns
t_{PXIZ}	Input instr float after $\overline{PSEN}$		59		t_{CLCL} – 25	ns
t_{AVIV}	Address to valid instr in		312		$5t_{CLCL}$ – 105	ns
t_{PLAZ}	$\overline{PSEN}$ low to address float		20		20	ns
t_{RLRH}	$\overline{RD}$ pulse width	400		$6t_{CLCL}$ – 100		ns
t_{WLWH}	$\overline{WR}$ pulse width	400		$6t_{CLCL}$ – 100		ns
t_{RLDV}	$\overline{RD}$ low to valid data in		252		$5t_{CLCL}$ – 165	ns
t_{RHDX}	Data hold after $\overline{RD}$	0		0		ns
t_{RHDZ}	Data float after $\overline{RD}$		97		$2t_{CLCL}$ – 70	ns
t_{LLDV}	ALE low to valid data in		517		$8t_{CLCL}$ – 150	ns
t_{AVDV}	Address to valid data in		585		$9t_{CLCL}$ – 165	ns
t_{LLWL}	ALE low to $\overline{RD}$ or $\overline{WR}$ low	200	300	$3t_{CLCL}$ – 50	$3t_{CLCL}$ + 50	ns
t_{QVWX}	Data valid to $\overline{WR}$ transition	23		t_{CLCL} – 60		ns
t_{WHQX}	Data hold after $\overline{WR}$	33		t_{CLCL} – 50		ns
t_{RLAZ}	$\overline{RD}$ low to address float		0		0	ns
t_{WHLH}	$\overline{RD}$ or $\overline{WR}$ high to ALE high	43	123	t_{CLCL} – 40	t_{CLCL} + 40	ns
Port 6 input (input rise and fall times 5ns)						
t_{ILIH}	$\overline{IDS}$ width	270		$3t_{CLCL}$ + 20		ns
t_{DVIH}	Data setup to $\overline{IDS}$ high	10		10		ns
t_{IHDX}	Data hold after $\overline{IDS}$	30		30		ns
t_{FLIL}	$\overline{PE}$ to $\overline{IDS}$	25		25		ns
t_{IVFV}	$\overline{IDS}$ to BFLAG (IBF) delay		130		130	ns
Port 6 output						
t_{OLOH}	$\overline{ODS}$ width	270		$3t_{CLCL}$ + 20		ns
t_{FVDV}	SEL to data out delay		85		85	ns
t_{OLDV}	$\overline{ODS}$ to data out delay		80		80	ns
t_{OHDZ}	$\overline{ODS}$ to data float delay		35		35	ns
t_{OVFV}	$\overline{ODS}$ to AFLAG (OBF) delay		100		100	ns
t_{FLDV}	$\overline{PE}$ to data out delay		120		120	ns
t_{OHFH}	$\overline{ODS}$ high to AFLAG (SEL) delay	100		100		ns

NOTES:

1. Stress above those listed under Absolute Maximum Rating may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or at any conditions other than those described in the AC and DC Electrical Characteristics section of this specification is not implied.
2. This product includes circuitry specifically designed for the protection on its internal devices from damaging effects of excessive ststic charge. Nonetheless it is suggested that conventional precautions be taken to avoid applying voltages greater than the rated maximum.
3. Parameters are valid over operating temperature range unless otherwise specified.
4. Capacitive loading on ports 0 and 2 can cause spurious noise to be superimposed on the $V^{OLS}$of ALE and ports 1 and 3. The noise is caused by external bus capacitance discharging into the port 0 and pot 2 pins when these pins make 1-to-0 transitions during bus operations. In the worst cases it could be desirable to qualify ALE with a Schmitt trigger, or use an address latch with a Schmitt trigger STROBE input.
5. Capacitive loading on ports 0 and 2 can cause the V_{OH} on ALE and $\overline{PSEN}$ to momentarily fall before the 0.9 V_{CC} specification when the address bits are stabilizing.
6. Power-down I_{CC} is measured with all output pins disconnected; $\overline{EA}$ = port 0 = V_{CC}; XTAL2 N.C.; RST = V_{SS}.
7. I_{CC} is measured with all output pins disconnected; XTAL1 driven with t_{CLCH}, t_{CHCL} = 5ns, V_{IL} = V_{SS} +0.5V, V_{IH} = V_{CC} –0.5V; XTAL1 N.C.; $\overline{EA}$ = RST = Port 0 = V_{CC}. I_{CC} will be slightly higher if a crystal oscillator is used.
8. Idle I_{CC} is measured with all output pins disconnected; XTAL1 driven with t_{CLCH}, t_{CHCL} = 5ns, V_{IL} = V_{SS} +0.5V, V_{IH} = V_{CC} –0.5V; XTAL N.C.; port 0 = V_{CC}; $\overline{EA}$ = RST = V_{SS}.

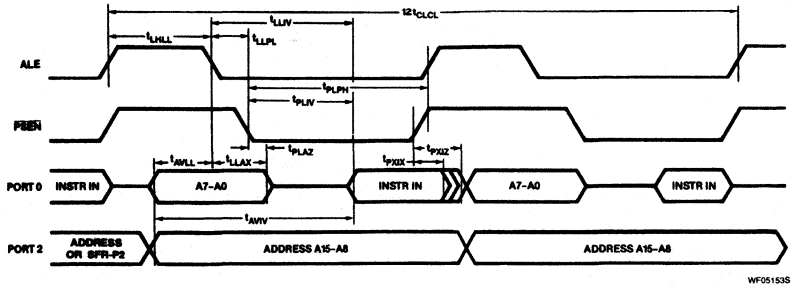


Figure 2. Program Memory Read Cycle

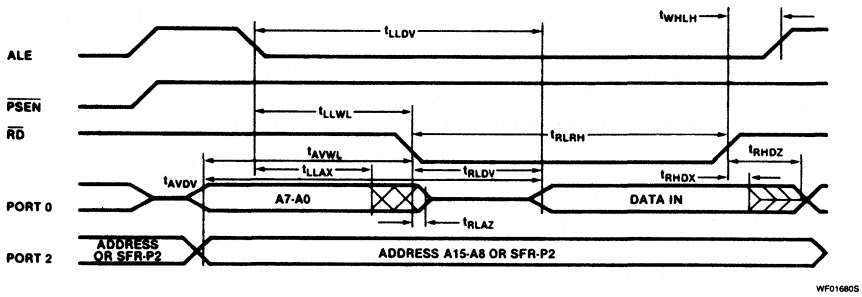


Figure 3. Data Memory Read Cycle

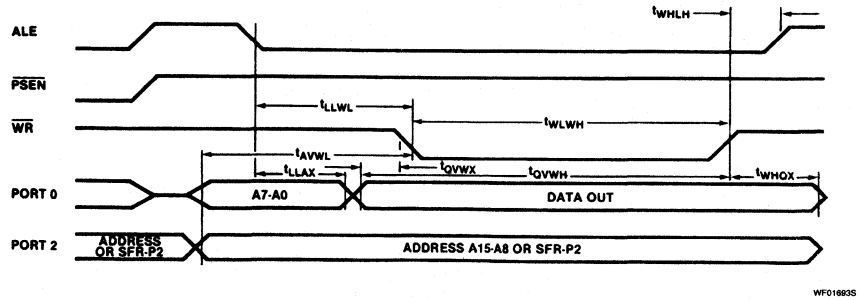


Figure 4. Data Memory Write Cycle

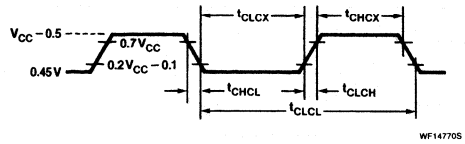


Figure 5. External Clock Drive Waveform

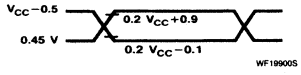


Figure 6. AC Testing Input, Output Waveforms[1]

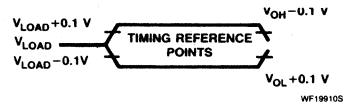


Figure 7. Float Waveforms[2]

NOTES:
1. AC inputs during testing are driven at $V_{CC}-0.5$ for a logic "1" and 0.45V for a logic "0". Timing measurements are made at V_{IH} min. for a logic "1" and V_{IL} max. for a logic "0".
2. For timing purposes a port pin is no longer floating when a 100mV change from load voltage occurs, and begins to flot when a 100mV change from the loaded V_{OH}/V_{OL} level occurs. $I_{OL}/I_{OH} > \pm 20$mA.

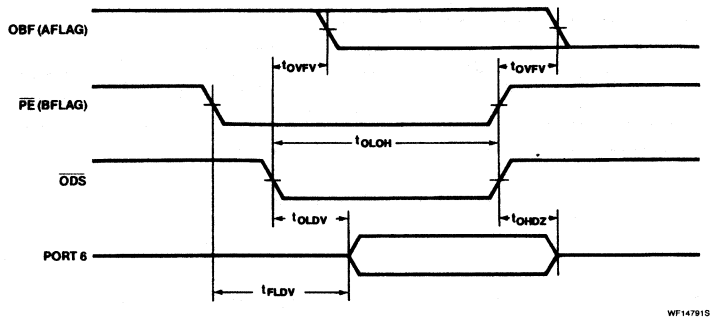


Figure 8. Port 6 Output Waveforms

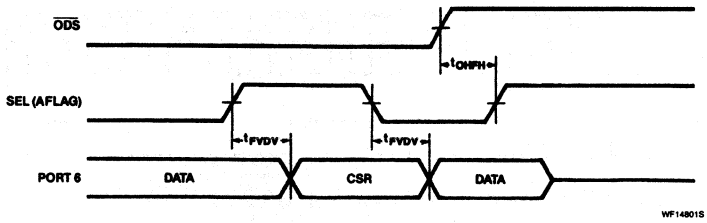


Figure 9. Port 6 Select Mode Waveforms

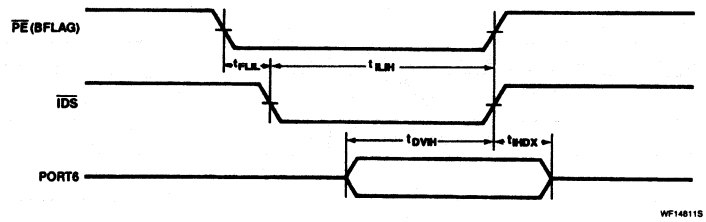


Figure 10. Port 6 Input Waveforms

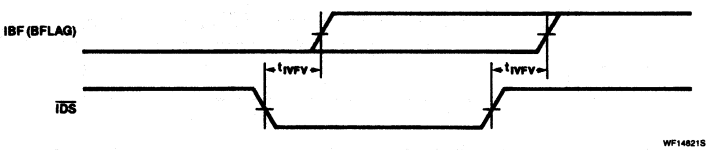
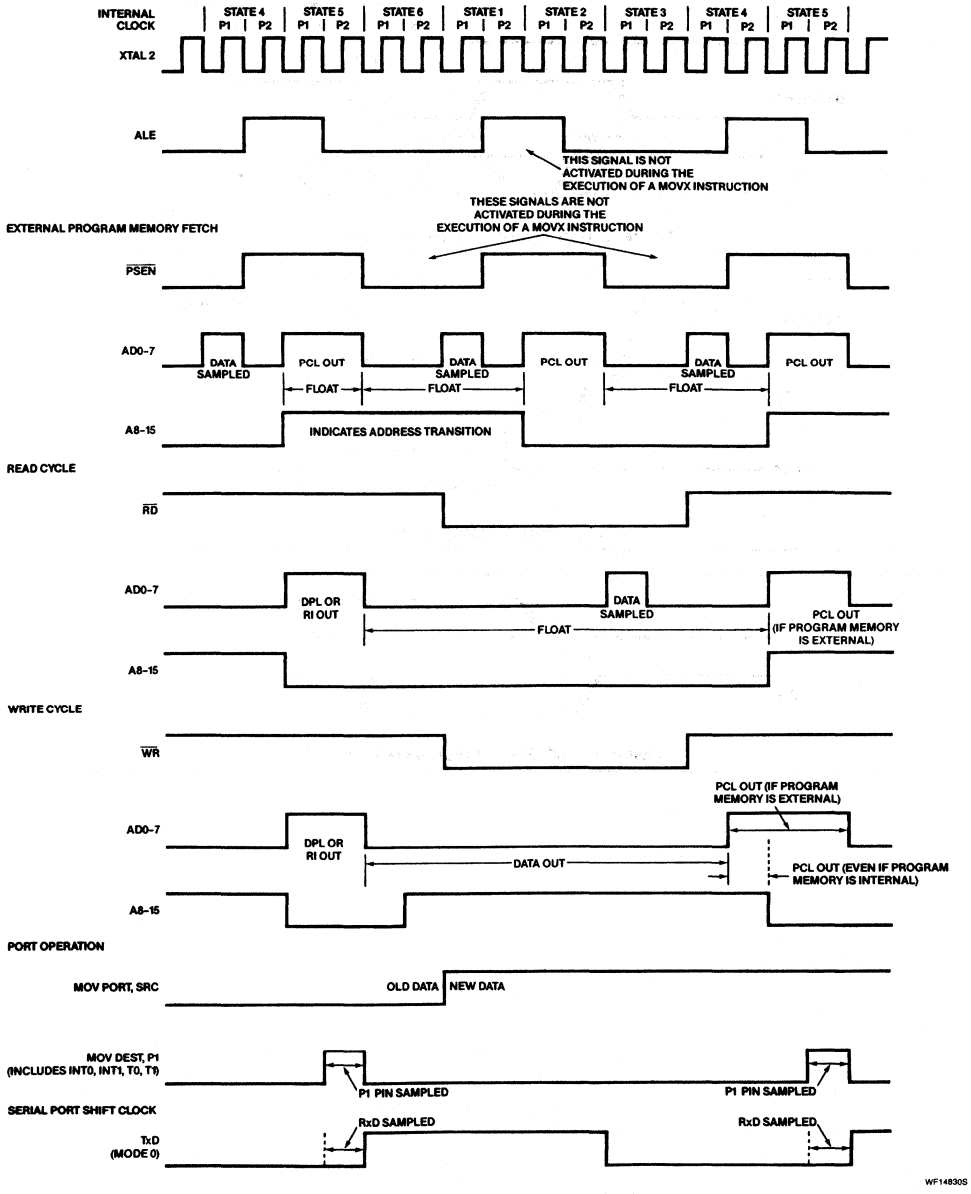


Figure 11. IBF Flag Output Waveforms



NOTE:
This diagram indicates when signals are clocked internally, however, the time it takes the signals to propagate to the pins ranges from 25ns to 125ns. This propagation delay is dependent on variables such as temperature and pin loading. Propagation also varies from output to output and component to component. Typically (T_A = 25°C, fully loaded), $\overline{RD}$ and $\overline{WR}$ propagation delays are approximately 50ns; propagation delays for other signals are approximately 85ns. These timing delays are incorporated in the AC specifications.

Figure 12. Timing Waveforms

Table 2. Instruction Set

MNEMONIC		DESCRIPTION	BYTE	CYCLES
Arithmetic operations				
ADD	A,Rn	Add register to accumulator	1	1
ADD	A,direct	Add direct byte to accumulator	2	1
ADD	A,@Ri	Add indirect RAM to accumulator	1	1
ADD	A,R#data	Add immediate data to accumulator	2	1
ADDC	A,Rn	Add register to accumulator with carry	1	1
ADDC	A,direct	Add direct byte to A with carry flag	2	1
ADDC	A,@Ri	Add indirect RAM to A with carry flag	1	1
ADDC	A,#data	Add immediate data to A with carry flag	2	1
SUBB	A,Rn	Subtract register from A with borrow	1	1
SUBB	A,direct	Subtract direct byte from A with borrow	2	1
SUBB	A,@Ri	Subtract indirect RAM from A w/borrow	1	1
SUBB	A,#data	Subtract immed. data from A w/borrow	2	1
INC	A	Increment accumulator	1	1
INC	Rn	Increment register	1	1
INC	direct	Increment direct byte	2	1
INC	@Ri	Increment indirect RAM	1	1
DEC	A	Decrement accumulator	1	1
DEC	Rn	Decrement register	1	1
DEC	direct	Decrement direct byte	2	1
DEC	@Ri	Decrement indirect RAM	1	1
INC	DPTR	Increment data pointer	1	2
MUL	AB	Multiply A & B	1	4
DIV	AB	Divide A by B	1	4
DA	A	Decimal adjust accumulator	1	1
Logical operations				
ANL	A,Rn	AND register to accumulator	1	1
ANL	A,direct	AND direct byte to accumulator	2	1
ANL	A,@Ri	AND indirect RAM to accumulator	1	1
ANL	A,#data	AND immediate data to accumulator	2	1
ANL	direct,A	AND accumulator to direct byte	2	1
ANL	direct,#data	AND immediate data to direct byte	3	2
ORL	A,Rn	OR register to accumulator	1	1
ORL	A,direct	OR direct byte to accumulator	2	1
ORL	A,@Ri	OR indirect RAM to accumulator	1	1
ORL	A,#data	OR immediate data to accumulator	2	1
ORL	direct,A	OR accumulator to direct byte	2	1
ORL	direct,#data	OR immediate data to direct byte	3	2
XRL	A,Rn	Exclusive-OR register to accumulator	1	1
XRL	A,direct	Exclusive-OR direct byte to accumulator	2	1
XRL	A,@Ri	Exclusive-OR indirect RAM to A	1	1
XRL	A,#data	Exclusive-OR immediate data to A	2	1
XRL	direct,A	Exclusive-OR accumulator to direct byte	2	1
XRL	direct,#data	Exclusive-OR immediate data to direct	3	2
CLR	A	Clear accumulator	1	1
CPL	A	Complement accumulator	1	1
RL	A	Rotate accumulator left	1	1
RLC	A	Rotate A left through the carry flag	1	1
RR	A	Rotate accumulator right	1	1
RRC	A	Rotate A right through carry flag	1	1
SWAP	A	Swap nibbles within the accumulator	1	1
Data transfer				
MOV	A,Rn	Move register to accumulator	1	1
MOV	A,direct	Move direct byte to accumulator	2	1
MOV	A,@Ri	Move indirect RAM to accumulator	1	1
MOV	A,#data	Move immediate data to accumulator	2	1
MOV	Rn,A	Move accumulator to register	1	1
MOV	Rn,direct	Move direct byte to register	2	2
MOV	Rn,#data	Move immediate data to register	2	1
MOV	direct,A	Move accumulator to direct byte	2	1
MOV	direct,Rn	Move register to direct byte	2	2
MOV	direct,direct	Move direct byte to direct	3	2
MOV	direct,@Ri	Move indirect RAM to direct byte	2	2
MOV	direct,#data	Move immediate data to direct byte	3	2
MOV	@Ri,A	Move accumulator to indirect RAM	1	1
MOV	@Ri,direct	Move direct byte to indirect RAM	2	2

Table 2. Instruction Set (Continued)

MNEMONIC		DESCRIPTION	BYTE	CYCLES
Data transfer (continued)				
MOV	@Ri,#data	Move immediate data to indirect RAM	2	1
MOV	DPTR,#data16	Load data pointer with a 16-bit constant	3	2
MOVC	A,@A + DPTR	Move code byte relative to DPTR to A	1	2
MOVC	A,@A + PC	Move code byte relative to PC to A	1	2
MOVX	A,@Ri	Move external RAM (8-bit addr) to A	1	2
MOVX	A,@DPTR	Move external RAM (16-bit addr) to A	1	2
MOVX	@Ri,A	Move A to external RAM (8-bit addr)	1	2
MOVX	@DPTR,A	Move A to external RAM (16-bit addr)	1	2
PUSH	direct	Push direct byte onto stack	2	2
POP	direct	Pop direct byte from stack	2	2
XCH	A,Rn	Exchange register with accumulator	1	1
XCH	A,direct	Exchange direct byte with accumulator	2	1
XCH	A,@Ri	Exchange indirect RAM with A	1	1
XCHD	A,@Ri	Exchange low-order digit ind. RAM w/A	1	1
Boolean variable manipulation				
CLR	C	Clear carry flag	1	1
CLR	bit	Clear direct bit	2	1
SETB	C	Set carry flag	1	1
SETB	bit	Set direct bit	2	1
CPL	C	Complement carry flag	1	1
CPL	bit	Complement direct bit	2	1
ANL	C,bit	AND direct bit to carry flag	2	2
ANL	C,/bit	AND complement of direct bit to carry	2	2
ORL	C,bit	OR direct bit to carry flag	2	2
ORL	C,/bit	OR complement of direct bit to carry	2	2
MOV	C,bit	Move direct bit to carry flag	2	1
MOV	bit,C	Move carry flag to direct bit	2	2
Program and machine control				
ACALL	addr11	Absolute subroutine call	2	2
LCALL	addr16	Long subroutine call	3	2
RET		Return from subroutine	1	2
RETI		Return from interrupt	1	2
AJMP	addr11	Absolute jump	2	2
LJMP	addr16	Long jump	3	2
SJMP	rel	Short jump (relative addr)	2	2
JMP	@A + DPTR	Jump indirect relative to the DPTR	1	2
JZ	rel	Jump if accumulator is zero	2	2
JNZ	rel	Jump if accumulator is not zero	2	2
JC	rel	Jump if carry flag is set	2	2
JNC	rel	Jump if no carry flag	2	2
JB	bit,rel	Jump if direct bit set	3	2
JNB	bit,rel	Jump if direct bit not set	3	2
JBC	bit,rel	Jump if direct bit is set & clear bit	3	2
CJNE	A,direct,rel	Compare direct to A & jump if not equal	3	2
CJNE	A,#data,rel	Comp. immed. to A & jump if not equal	3	2
CJNE	Rn,#data,rel	Comp. immed. to reg. & jump if not equal	3	2
CJNE	@Ri,#data,rel	Comp. immed. to ind. & jump if not equal	3	2
DJNZ	Rn,rel	Decrement register & jump if not zero	2	2
DJNZ	direct,rel	Decrement direct & jump if not zero	3	2
NOP		No operation	1	1

Notes on data addressing modes:

Rn – Working register R0-R7
direct – 128 internal RAM locations, any I/O port, control or status register
@Ri – Indirect Internal RAM location addressed by register R0 or R1
#data – 8-bit constant included in instruction
#data16 – 16-bit constant included as bytes 2 & 3 of instruction
bit – 128 software flags, any I/O pin, control or status bit

Notes on program addressing modes:

addr16 – Destination address for LCALL & LJMP may be anywhere within the 64-kilobyte program memory address space.
addr11 – Destination address for ACALL & AJMP will be within the same 2-kilobyte page of program memory as the first byte of the following instruction.
rel – SJMP and all conditional jumps include an 8-bit offset byte. Range is +127 – 128 bytes relative to first byte of the following instruction.

Table 3. Instruction Opcodes in Hexadecimal Order

HEX CODE	NUMBER OF BYTES	MNEMONIC	OPERANDS
00	1	NOP	
01	2	AJMP	code addr
02	3	LJMP	code addr
03	1	RR	A
04	1	INC	A
05	2	INC	data addr
06	1	INC	@R0
07	1	INC	@R1
08	1	INC	R0
09	1	INC	R1
0A	1	INC	R2
0B	1	INC	R3
0C	1	INC	R4
0D	1	INC	R5
0E	1	INC	R6
0F	1	INC	R7
10	3	JBC	bit addr, code addr
11	2	ACALL	code addr
12	3	LCALL	code addr
13	1	RRC	A
14	1	DEC	A
15	2	DEC	data addr
16	1	DEC	@R0
17	1	DEC	@R1
18	1	DEC	R0
19	1	DEC	R1
1A	1	DEC	R2
1B	1	DEC	R3
1C	1	DEC	R4
1D	1	DEC	R5
1E	1	DEC	R6
1F	1	DEC	R7
20	3	JB	bit addr, code addr
21	2	AJMP	code addr
22	1	RET	
23	1	RL	A
24	2	ADD	A,#data
25	2	ADD	A,data addr
26	1	ADD	A,@R0
27	1	ADD	A,@R1
28	1	ADD	A,R0
29	1	ADD	A,R1
2A	1	ADD	A,R2
2B	1	ADD	A,R3
2C	1	ADD	A,R4
2D	1	ADD	A,R5
2E	1	ADD	A,R6
2F	1	ADD	A,R7
30	3	JNB	bit addr,code addr
31	2	ACALL	code addr
32	1	RETI	
33	1	RLC	A
34	2	ADDC	A,#data
35	2	ADDC	A,data addr
36	1	ADDC	A,@R0
37	1	ADDC	A,@R1
38	1	ADDC	A,R0
39	1	ADDC	A,R1
3A	1	ADDC	A,R2
3B	1	ADDC	A,R3
3C	1	ADDC	A,R4
3D	1	ADDC	A,R5
3E	1	ADDC	A,R6
3F	1	ADDC	A,R7
40	2	JC	code addr
41	2	AJMP	code addr
42	2	ORL	data addr,A
43	3	ORL	data addr,#data
44	2	ORL	A,#data
45	2	ORL	A,data addr
46	1	ORL	A,@R0
47	1	ORL	A,@R1
48	1	ORL	A,R0
49	1	ORL	A,R1
4A	1	ORL	A,R2
4B	1	ORL	A,R3
4C	1	ORL	A,R4
4D	1	ORL	A,R5
4E	1	ORL	A,R6
4F	1	ORL	A,R7
50	2	JNC	code addr
51	2	ACALL	code addr
52	2	ANL	data addr,A
53	3	ANL	data addr,#data
54	2	ANL	A,#data
55	2	ANL	A,data addr
56	1	ANL	A,@R0
57	1	ANL	A,@R1
58	1	ANL	A,R0
59	1	ANL	A,R1
5A	1	ANL	A,R2
5B	1	ANL	A,R3
5C	1	ANL	A,R4
5D	1	ANL	A,R5
5E	1	ANL	A,R6
5F	1	ANL	A,R7
60	2	JZ	code addr
61	2	AJMP	code addr
62	2	XRL	data addr,A
63	3	XRL	data addr,#data
64	2	XRL	A,#data
65	2	XRL	A,data addr
66	1	XRL	A,@R0
67	1	XRL	A,@R1
68	1	XRL	A,R0
69	1	XRL	A,R1
6A	1	XRL	A,R2
6B	1	XRL	A,R3
6C	1	XRL	A,R4
6D	1	XRL	A,R5
6E	1	XRL	A,R6
6F	1	XRL	A,R7
70	2	JNZ	code addr
71	2	ACALL	code addr
72	2	ORL	C,bit addr
73	1	JMP	@A+DPTR
74	2	MOV	A,#data
75	3	MOV	data addr,#data
76	2	MOV	@R0,#data
77	2	MOV	@R1,#data
78	2	MOV	R0,#data
79	2	MOV	R1,#data
7A	2	MOV	R2,#data
7B	2	MOV	R3,#data
7C	2	MOV	R4,#data
7D	2	MOV	R5,#data
7E	2	MOV	R6,#data
7F	2	MOV	R7,#data

Table 3. Instruction Opcodes in Hexadecimal Order (Continued)

HEX CODE	NUMBER OF BYTES	MNEMONIC	OPERANDS
80	2	SJMP	code addr
81	2	AJMP	code addr
82	2	ANL	C,bit addr
83	1	MOVC	A,@A + PC
84	1	DIV	AB
85	3	MOV	data addr,data addr
86	2	MOV	data addr,@R0
87	2	MOV	data addr,@R1
88	2	MOV	data addr,R0
89	2	MOV	data addr,R1
8A	2	MOV	data addr,R2
8B	2	MOV	data addr,R3
8C	2	MOV	data addr,R4
8D	2	MOV	data addr,R5
8E	2	MOV	data addr,R6
8F	2	MOV	data addr,R7
90	3	MOV	DPTR,#data
91	2	ACALL	code addr
92	2	MOV	bit addr,C
93	1	MOVC	A,@A + DPTR
94	2	SUBB	A,#data
95	2	SUBB	A,data addr
96	1	SUBB	A,@R0
97	1	SUBB	A,@R1
98	1	SUBB	A,R0
99	1	SUBB	A,R1
9A	1	SUBB	A,R2
9B	1	SUBB	A,R3
9C	1	SUBB	A,R4
9D	1	SUBB	A,R5
9E	1	SUBB	A,R6
9F	1	SUBB	A,R7
A0	2	ORL	C,/bit addr
A1	2	AJMP	code addr
A2	2	MOV	C,bit addr
A3	1	INC	DPTR
A4	1	MUL	AB
A5		reserved	
A6	2	MOV	@R0,data addr
A7	2	MOV	@R1,data addr
A8	2	MOV	R0,data addr
A9	2	MOV	R1,data addr
AA	2	MOV	R2,data addr
AB	2	MOV	R3,data addr
AC	2	MOV	R4,data addr
AD	2	MOV	R5,data addr
AE	2	MOV	R6,data addr
AF	2	MOV	R7,data addr
B0	2	ANL	C,/bit addr
B1	2	ACALL	code addr
B2	2	CPL	bit addr
B3	1	CPL	C
B4	3	CJNE	A,#data,code addr
B5	3	CJNE	A,data addr,code addr
B6	3	CJNE	@R0,#data,code addr
B7	3	CJNE	@R1,#data,code addr
B8	3	CJNE	R0,#data,code addr
B9	3	CJNE	R1,#data,code addr
BA	3	CJNE	R2,#data,code addr
BB	3	CJNE	R3,#data,code addr
BC	3	CJNE	R4,#data,code addr
BD	3	CJNE	R5,#data,code addr
BE	3	CJNE	R6,#data,code addr
BF	3	CJNE	R7,#data,code addr

HEX CODE	NUMBER OF BYTES	MNEMONIC	OPERANDS
C0	2	PUSH	data addr
C1	2	AJMP	code addr
C2	2	CLR	bit addr
C3	1	CLR	C
C4	1	SWAP	A
C5	2	XCH	A,data addr
C6	1	XCH	A,@R0
C7	1	XCH	A,@R1
C8	1	XCH	A,R0
C9	1	XCH	A,R1
CA	1	XCH	A,R2
CB	1	XCH	A,R3
CC	1	XCH	A,R4
CD	1	XCH	A,R5
CE	1	XCH	A,R6
CF	1	XCH	A,R7
D0	2	POP	data addr
D1	2	ACALL	code addr
D2	2	SETB	bit addr
D3	1	SETB	C
D4	1	DA	A
D5	3	DJNZ	data addr,code addr
D6	1	XCHD	A,@R0
D7	1	XCHD	A,@R1
D8	2	DJNZ	R0,code addr
D9	2	DJNZ	R1,code addr
DA	2	DJNZ	R2,code addr
DB	2	DJNZ	R3,code addr
DC	2	DJNZ	R4,code addr
DD	2	DJNZ	R5,code addr
DE	2	DJNZ	R6,code addr
DF	2	DJNZ	R7,code addr
E0	1	MOVX	A,@DPTR
E1	2	AJMP	code addr
E2	1	MOVX	A,@R0
E3	1	MOVX	A,@R1
E4	1	CLR	A
E5	2	MOV	A,data addr
E6	1	MOV	A,@R0
E7	1	MOV	A,@R1
E8	1	MOV	A,R0
E9	1	MOV	A,R1
EA	1	MOV	A,R2
EB	1	MOV	A,R3
EC	1	MOV	A,R4
ED	1	MOV	A,R5
EE	1	MOV	A,R6
EF	1	MOV	A,R7
F0	1	MOVX	@DPTR,A
F1	2	ACALL	code addr
F2	1	MOVX	@R0,A
F3	1	MOVX	@R1,A
F4	1	CPL	A
F5	2	MOV	data addr,A
F6	1	MOV	@R0,A
F7	1	MOV	@R1,A
F8	1	MOV	R0,A
F9	1	MOV	R1,A
FA	1	MOV	R2,A
FB	1	MOV	R3,A
FC	1	MOV	R4,A
FD	1	MOV	R5,A
FE	1	MOV	R6,A
FF	1	MOV	R7,A

SUPERSEDES DATA OF JULY 1988

SINGLE-CHIP 8-BIT MICROCONTROLLER

GENERAL DESCRIPTION

The PCB83C552 single-chip 8-bit microcontroller is manufactured in an advanced CMOS process and is a derivative of the PCB80C51 microcontroller family. The PCB83C552 has the same instruction set as the PCB80C51. Two versions of the derivative exist although the generic term "PCB83C552" is used to refer to both family members:

- PCB83C552: 8 K bytes mask-programmable ROM, 256 bytes RAM
- PCB80C552: ROM-less version of the PCB83C552

This I/O intensive device provides architectural enhancements to function as a controller in the field of automotive electronics, specifically engine management and gear box control.

The PCB83C552 contains a non-volatile 8 K x 8 read-only program memory, a volatile 256 x 8 read/write data memory, six 8-bit I/O ports, two 16-bit timer/event counters (identical to the timers of the 80C51), an additional 16-bit timer coupled to capture and compare latches, a fifteen-source, two-priority-level, nested interrupt structure, an 8-input ADC, a dual DAC pulse width modulated interface, two serial interfaces (UART and I^2C-bus), a 'watchdog' timer and on-chip oscillator and timing circuits. For systems that require extra capability, the PCB83C552 can be expanded using standard TTL compatible memories and logic.

The device also functions as an arithmetic processor having facilities for both binary and BCD arithmetic plus bit-handling capabilities. The instruction set consists of over 100 instructions: 49 one-byte, 45 two-byte and 17 three-byte. With a 12 MHz crystal, 58% of the instructions are executed in 1 μs and 40% in 2 μs. Multiply and divide instructions require 4 μs.

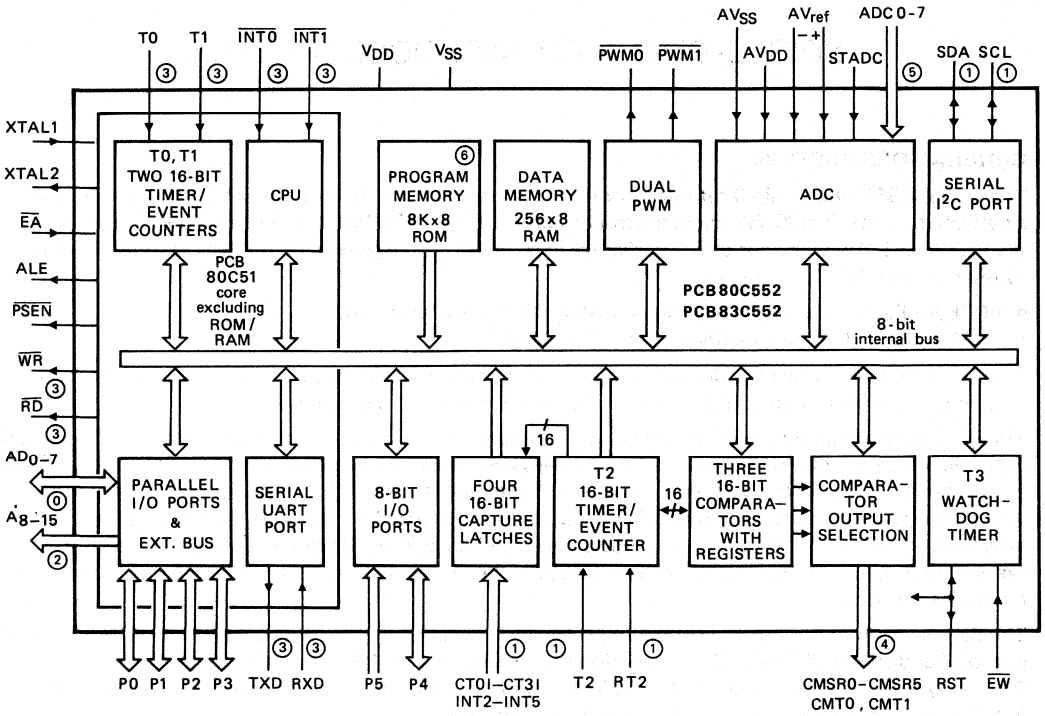
Features

- 80C51 central processing unit
- 8 K x 8 ROM, expandable externally to 64 K bytes
- 256 x 8 RAM, expandable externally to 64 K bytes
- Two standard 16-bit timer/counters
- An additional 16-bit timer/counter coupled to four capture registers and three compare registers
- A 10-bit ADC with 8 multiplexed analogue inputs
- Two 8-bit resolution, Pulse Width Modulated outputs
- Five 8-bit I/O ports plus one 8-bit input port shared with analogue inputs
- I^2C-bus serial I/O port with byte orientated master and slave functions
- Full-duplex UART compatible with the standard PCB80C51
- On-chip watchdog timer

- **A version for extended temperature range is in preparation**

PACKAGE OUTLINES

PCB83C552; PCB80C552: 68-lead plastic leaded-chip-carrier (PLCC) (SOT188 pedestal or SOT188AA pocket versions, these are interchangeable).



⓪ alternative function of port 0
① alternative functions of port 1
② alternative function of port 2
③ alternative function of port 3
④ alternative function of port 4
⑤ alternative function of port 5
⑥ not present in PCB80C552

7Z97647.5

POWER SUPPLY
V_{DD} + 5 V MAIN SUPPLY
V_{SS} GROUND

Fig. 1 Block diagram.

DEVELOPMENT DATA

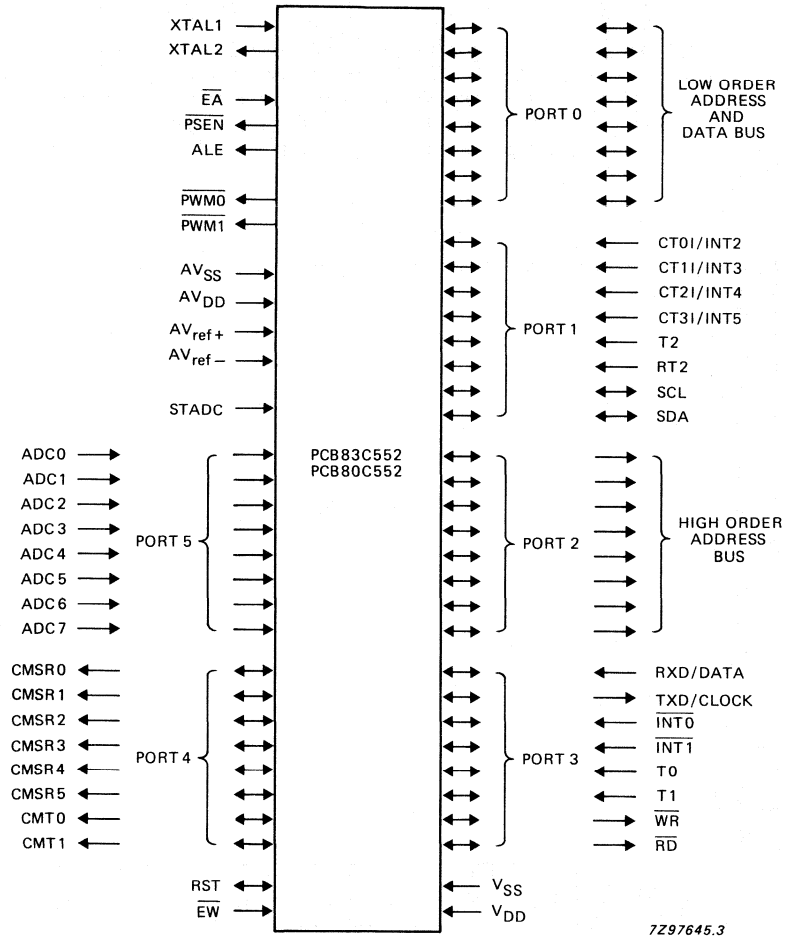
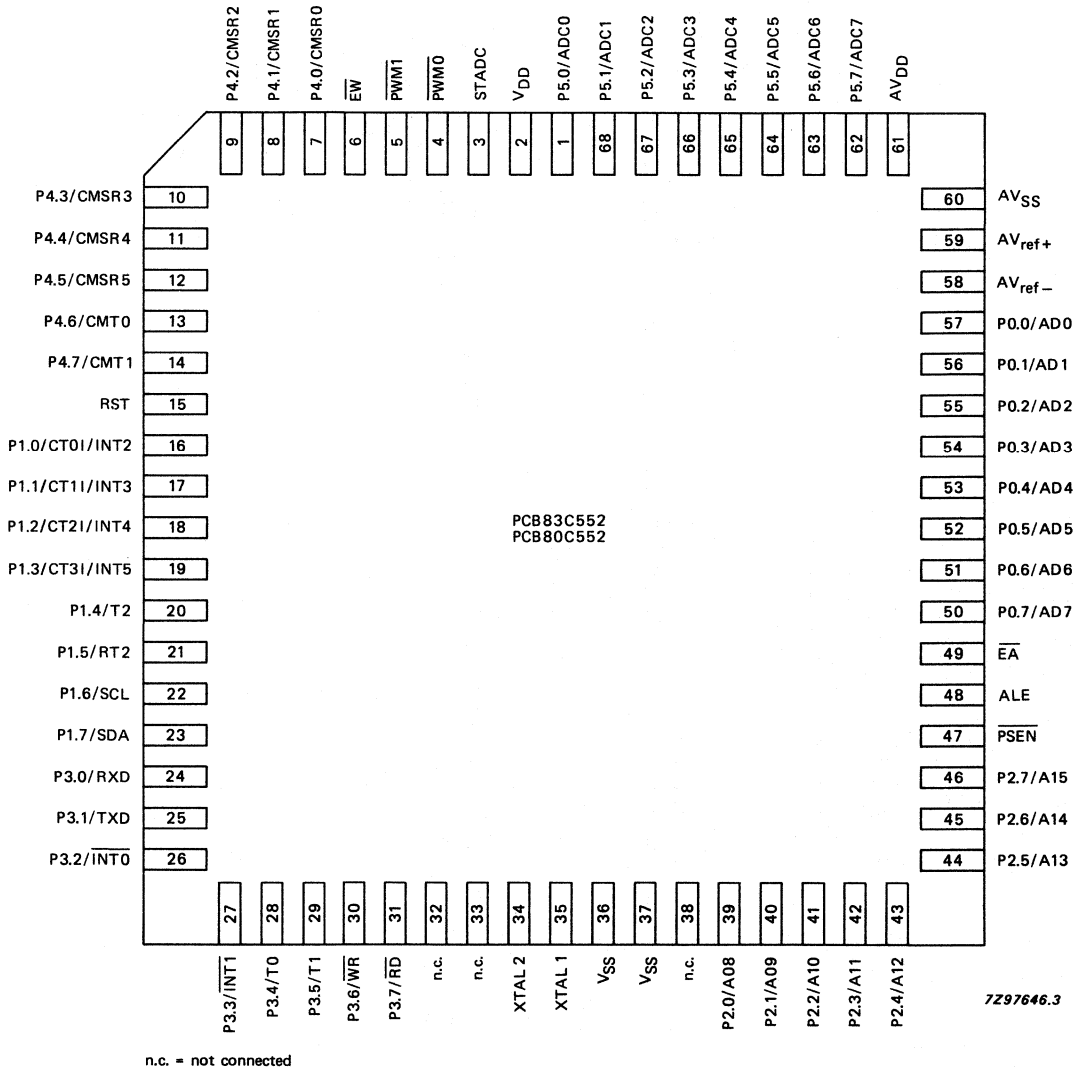


Fig. 2 Functional diagram.



n.c. = not connected

Fig. 3 Pinning diagram for PCB83C552.

DEVELOPMENT DATA

PINNING

2	V_{DD}	**Digital power supply**: + 5 V power supply pin during normal operation, Idle mode and Power-down mode
3	STADC	**Start ADC operation**: Input starting analogue to digital conversion (ADC operation can also be started by software)
4	$\overline{PWM0}$	**Pulse width** modulation output 0
5	$\overline{PWM1}$	**Pulse width** modulation output 1
6	$\overline{EW}$	**Enable watchdog timer**: Enable for T3 watchdog timer and disable Power-down mode
7-14	P4.0-P4.7	**Port 4** 8-bit quasi-bidirectional I/O port

Port pin	Alternative function	
P4.0	CMSR0	Timer T2: compare and set/reset outputs on a match with timer T2
P4.1	CMSR1	
P4.2	CMSR2	
P4.3	CMSR3	
P4.4	CMSR4	
P4.5	CMSR5	
P4.6	CMT0	Timer T2: compare and toggle outputs on a match with timer T2
P4.7	CMT1	

15	RST	**Reset**: Input to reset the PCB83C552. It also provides a reset pulse as output when timer T3 overflows.
16-23	P1.0-P1.7	**Port 1** 8-bit quasi-bidirectional I/O port

Port pin	Alternative function	
P1.0	CT0I/INT2	Capture timer input signals for timer T2 or Interrupts INT2 to INT5
P1.1	CT1I/INT3	
P1.2	CT2I/INT4	
P1.3	CT3I/INT5	
P1.4	T2	: T2 event input
P1.5	RT2	: T2 timer reset signal. Rising edge triggered
P1.6	SCL	: Serial port clock line I^2C-bus
P1.7	SDA	: Serial port data line I^2C-bus

24-31	P3.0-P3.7	**Port 3** 8-bit quasi-bidirectional I/O port

Port pin	Alternative function	
P3.0	RXD	: Serial input port
P3.1	TXD	: Serial output port
P3.2	$\overline{INT0}$	: External interrupt
P3.3	$\overline{INT1}$	: External interrupt
P3.4	T0	: Timer 0 external input
P3.5	T1	: Timer 1 external input
P3.6	$\overline{WR}$	: External data memory write strobe
P3.7	$\overline{RD}$	: External data memory read strobe

32, 33	Not connected

PINNING (continued)

34	XTAL2	**Crystal input 2:** output of the inverting amplifier that forms the oscillator. Left open-circuit when an external oscillator clock is used.
35	XTAL1	**Crystal input 1:** input to the inverting amplifier that forms the oscillator, and input to the internal clock generator. Receives the external oscillator clock signal when an external oscillator is used.
36, 37	V_{SS}	**Two digital** ground pins
38		Not connected
		Port 2
39-46	P2.0-P2.7	8-bit quasi-bidirectional I/O port Port pin — Alternative function P2.0-P2.7 — High-order address byte for external memory (A08-A15)
47	$\overline{PSEN}$	**Program store enable:** active LOW read strobe to external program memory
48	ALE	**Address latch enable:** latches the low byte of the address during accesses to external memory. It is activated every six oscillator periods. During an external data memory access one ALE pulse is skipped. ALE can drive up to 8 LSTTL inputs and handles CMOS inputs without an external pull-up.
49	$\overline{EA}$	**External access:** When $\overline{EA}$ is held at TTL level HIGH, the CPU executes out of the internal program ROM provided the program counter is less than 8192. When $\overline{EA}$ is held at TTL LOW level, the CPU executes out of external program memory. $\overline{EA}$ is not allowed to float.
		Port 0
50-57	P0.7-P0.0	8-bit binary I/O port Port pin — Alternative function P0.7-P0.0 — Multiplexed low-order address and data bus of external memory (AD7-AD0)
58	$AV_{ref}-$	**Low end** of analogue to digital conversion reference resistor
59	$AV_{ref}+$	**High end** of analogue to digital conversion reference resistor
60	AV_{SS}	**Analogue** ground
61	AV_{DD}	**Analogue** power supply
		Port 5
62-68,1	P5.7-P5.0	8-bit input port Port pin — Alternative function P5.0-P5.7 — Eight input channels to ADC (ADC0-ADC7)

To avoid a 'latch-up' effect at power-on, the voltage on any pin at any time must not be higher or lower than V_{DD} + 0,5 V or V_{SS} − 0,5 V respectively.

FUNCTIONAL DESCRIPTION

General

The PCB83C552 is a stand-alone high-performance microcontroller designed for use in real-time applications such as instrumentation, industrial control and specific automotive control applications.

The device provides in addition to the 80C51 standard functions, a number of dedicated hardware functions for these applications.

The PCB83C552 is a control-oriented CPU with on-chip program and data memory. It can be extended with external program memory up to 64 K bytes. It can also access up to 64 K bytes of external data memory. For systems requiring extra capability, the PCB83C552 can be expanded using standard memories and peripherals.

The PCB83C552 has two software selectable modes of reduced activity for further power reduction — Idle and Power-down. The Idle mode freezes the CPU while allowing the RAM, timers, serial ports and interrupt system to continue functioning. The Power-down mode saves the RAM contents but freezes the oscillator causing all other chip functions to be inoperative.

Memory organisation

The central processing unit (CPU) manupilates operands in three memory spaces; these are the 64 K-byte external data memory, 256-byte internal data memory and the 64 K-byte internal and external program memory. The internal data memory address space is sub-divided into the 256-byte internal data RAM and 128-byte Special Function Register (SFR) address spaces, as shown in Fig. 4. Figures 5(a) and (b) show the Special Function Register memory map. Internal RAM locations 0-127 are directly and indirectly addressable. Internal RAM locations 128-255 are only indirectly addressable. The special function register locations 128-255 are only directly addressable.

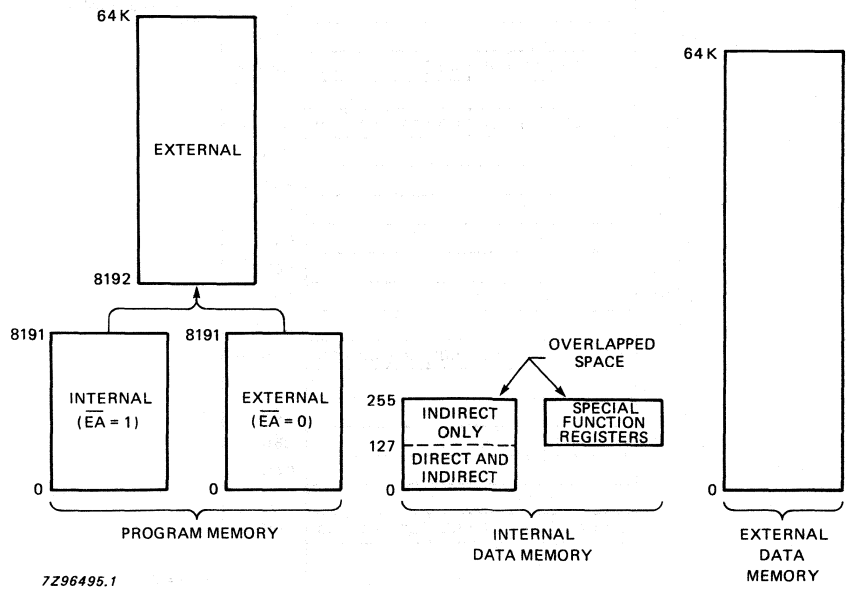
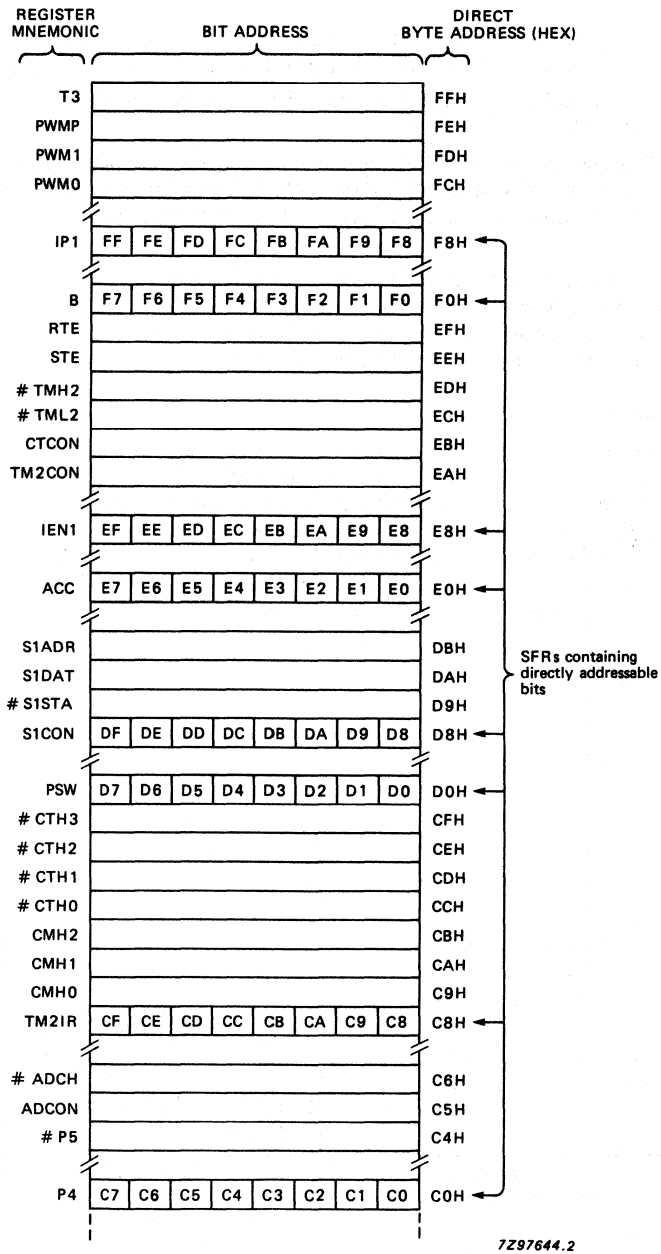


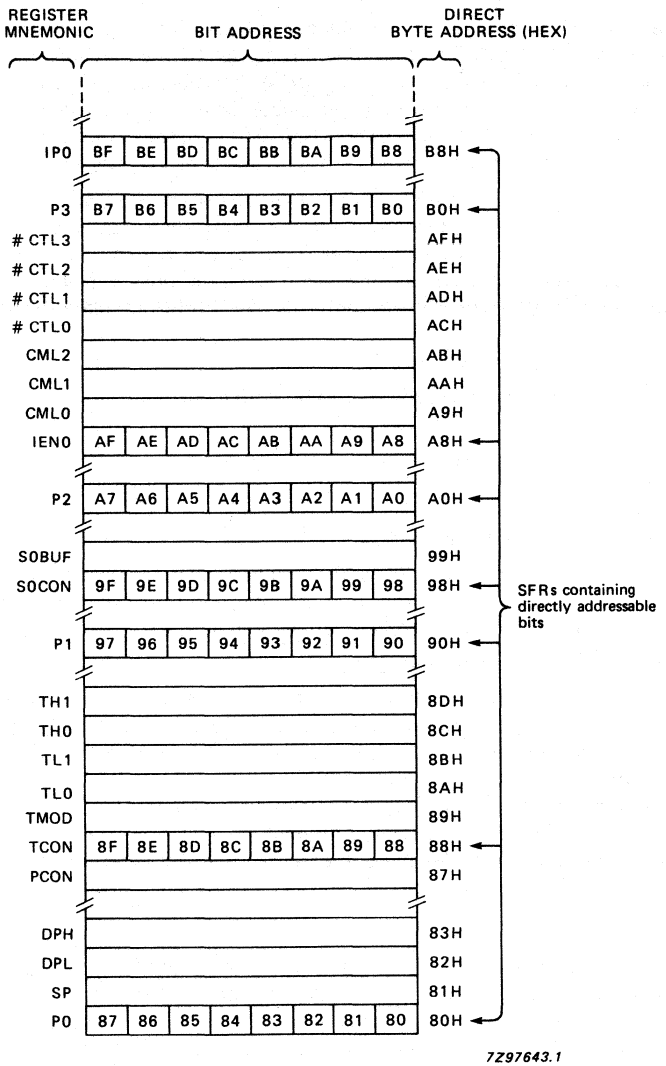
Fig. 4 Memory map.

FUNCTIONAL DESCRIPTION (continued)



# denotes read only registers

Fig. 5(a) Special Function Register memory map (continued in Fig. 5(b)).



# denotes read only registers

Fig. 5(b) Special Function Register memory map (continued from Fig. 5(a)).

FUNCTIONAL DESCRIPTION (continued)

The internal data RAM contains four register banks (each with eight registers), 128 addressable bits, a scratchpad area and the stack. The stack depth is limited by the available internal data RAM and its location is determined by the 8-bit stack pointer. All registers except the program counter and the four 8-register banks reside in the special function register address space. These memory mapped registers include arithmetic registers, pointers, I/O ports, interrupt system registers, ADC and PWM registers, timers and serial port registers. There are 128 addressable bit locations in the SFR address space.

The PCB83C552 contains 256 bytes of internal data RAM and 56 special function registers. It provides a non-paged program memory address space to accommodate relocatable code. Conditional branches are performed relative to the program counter. The register-indirect jump permits branching relative to a 16-bit base register with an offset provided by an 8-bit index register. 16-bit jumps and calls permit branching to any location in the contiguous 64 K program memory address space.

Addressing

The PCB83C552 has five methods for addressing source operands:

- Register
- Direct
- Register-Indirect
- Immediate
- Base-Register plus Index-Register-Indirect

The first three methods can be used for addressing destination operands. Most instructions have a "destination/source" field that specifies the data type, addressing methods and operands involved. For operations other than MOVs, the destination operand is also a source operand.

Access to memory addressing is as follows:

- Registers in one of the four 8-register banks through Register, Direct, or Register-Indirect
- 256 bytes of internal data RAM through Direct or Register-Indirect. Bytes 0-127 may be addressed directly/indirectly. Bytes 128-255 share their address locations with the SFR registers and so may only be addressed indirectly as data RAM
- Special function registers through Direct at address locations 128-255
- External data memory through Register-Indirect
- Program memory look-up tables through Base-Register plus Index-Register-Indirect

Instruction set

The PCB83C552 uses the powerful instruction set of the PCB80C51. Additional special function registers are incorporated to control the on-chip peripherals. The instruction set consists of 49 single-byte, 45 two-byte and 17 three-byte instructions. When using a 12 MHz oscillator, 64 instructions execute in 1 μs and 45 instructions execute in 2 μs. Multiply and divide instructions execute in 4 μs.

I/O facilities

The PCB83C552 has six 8-bit ports. Ports 0 - 3 are the same as in the 80C51, with the exception of the additional functions of Port 1. The parallel I/O function of Port 4 is equal to that of Ports 1, 2 and 3. Port 5 has a parallel input port function, but has no function as an output port. Port lines P1.7 and P1.6 may be selected as the SDA and SCL lines of serial port SIO1 (I^2C). Because the I^2C-bus may be active while the device is disconnected from V_{DD}, these pins are provided with open drain drivers. Pins P1.7 and P1.6 do not have pull-up devices when used as ports.

Ports 0, 1, 2, 3, 4 and 5 perform the following alternative functions:

- Port 0: provides the multiplexed low-order address and data bus used for expanding the PCB83C552 with standard memories and peripherals.
- Port 1: Port 1 is used for a number of special functions;
  - 4 capture inputs
  - external counter input
  - external counter reset input
  - SCL and SDA for the I^2C interface

Bits whose alternate function is not used may be used as normal bidirectional I/O pins.

- Port 2: provides the high-order address bus when expanding the PCB83C552 with external program memory and/or external data memory.
- Port 3: pins can be configured individually to provide:
  - external interrupt request inputs
  - counter inputs
  - serial port receiver input and transmitter output
  - control signals to READ and WRITE external data memory

The generation or use of a Port 3 pin as an alternative function is carried out automatically by the PCB83C552 provided the associated Special Function Register bit is set high.

- Port 4: can be configured to provide signals indicating a match between timer counter T2 and its compare registers.
- Port 5: may be used in conjunction with the ADC interface. Unused analogue inputs can be used as digital inputs. As Port 5 lines may be used as inputs to the ADC, these digital inputs have an inherent hysteresis to prevent the input logic from drawing too much current from the power lines when driven by analogue signals. Channel to channel crosstalk should be taken into consideration when both digital and analogue signals are simultaneously input to Port 5 (see DC characteristics).

All ports are bidirectional with the exception of Port 5 which is an input port.

DEVELOPMENT DATA

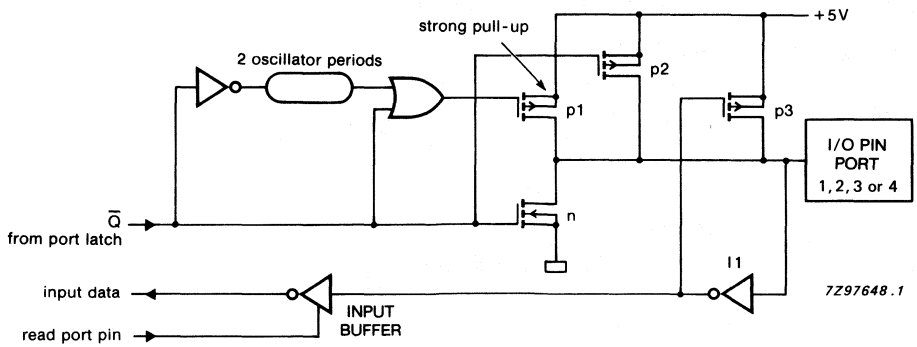


Fig. 6 I/O buffers in the PCB83C552 (Ports 2, 3, 4 and P1.0 to P1.5).

FUNCTIONAL DESCRIPTION (continued)

In addition to the standard 8-bit ports, the I/O facilities of the PCB83C552 also include a number of special I/O lines:

Pulse width modulated outputs

Two pulse width modulated output channels are provided with the PCB83C552. These channels output pulses of programmable length and interval. The repetition frequency is defined by an 8-bit prescaler PWMP which generates the clock for the counter. Both the prescaler and counter are common to both PWM channels. The 8-bit counter counts modulo 255 i.e. from 0 to 254 inclusive. The value of the 8-bit counter is compared to the contents of two registers: PWM0 and PWM1. Provided the contents of either of these registers is greater than the counter value, the output of $\overline{PWM0}$ or $\overline{PMW1}$ is set LOW. If the contents of these registers are equal to, or less than the counter value, the output will be HIGH. The pulse-width-ratio is therefore defined by the contents of the registers PWM0 and PWM1. The pulse-width-ratio is in the range of 0 to 255/255 and may be programmed in increments of 1/255.

The repetition frequency f_{PWM}, at the $\overline{PWMn}$ outputs is given by:

$$f_{PWM} = \frac{f_{osc}}{2 \times (1 + PWMP) \times 255}$$

This gives a repetition frequency range of 92 Hz to 23,5 kHz (f_{osc} = 12 MHz).

By loading the PWM registers with either 00H or FFH, the PWM outputs can be retained at a constant HIGH or LOW level respectively. When loading FFH to the PWM registers, the 8-bit counter will never actually reach this value.

Both $\overline{PWMn}$ output pins are driven by push-pull drivers, and are not shared with any other function.

Prescaler frequency control register PWMP

	7	6	5	4	3	2	1	0
PWMP (FEH)	MSB							LSB

Bit	Function
PWMP.0-7	Prescaler division factor = (PWMP) + 1

Pulse width registers PWM0 and PWM1

	7	6	5	4	3	2	1	0
PWM0 (FCH) PWM1 (FDH)	MSB							LSB

Bit	Function
PWM0.0-7 PWM1.0-7	Low/high ratio of $\overline{PWMn}$ signals $= \frac{(PWMn)}{255 - (PWMn)}$

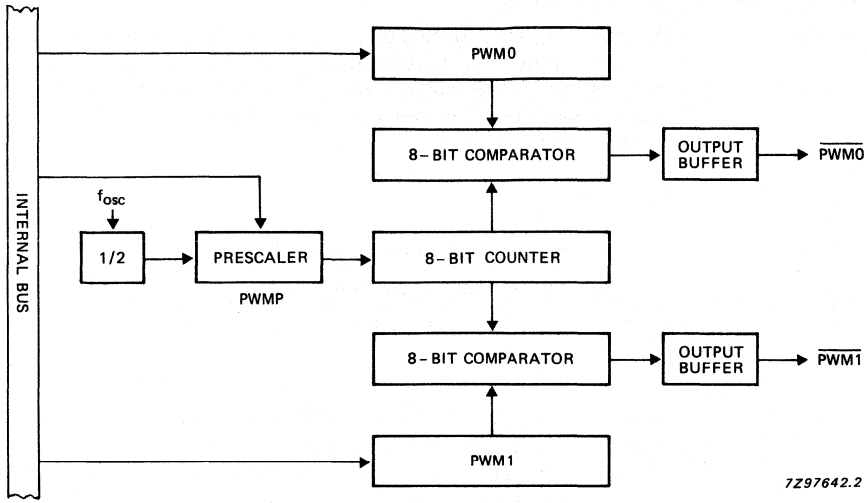


Fig. 7 Functional diagram of pulse width modulated outputs.

DEVELOPMENT DATA

Analogue input pins

The analogue input circuitry consists of an 8-input analogue multiplexer and an ADC with 10-bit resolution. The analogue reference voltage and analogue power supplies are connected via separate input pins. The conversion takes 50 machine cycles i.e. 50 μs at 12 MHz oscillator frequency.

The ADC is controlled using the ADCON control register. Input channels are selected by the analogue multiplexer, care of ADCON register bits 0-2.

ADC control register ADCON

	7	6	5	4	3	2	1	0
ADCON (C5H)	ADC.1	ADC.0	ADEX	ADCI	ADCS	AADR2	AADR1	AADR0

Bit	Symbol	Function
ADCON.7	ADC.1	Bit 1 of ADC converted value
ADCON.6	ADC.0	Bit 0 of ADC converted value
ADCON.5	ADEX	Enable external start of conversion by STADC 0 = Conversion cannot be started externally by STADC 1 = Conversion can be started externally by STADC
ADCON.4	ADCI	ADC interrupt flag: this flag is set when an ADC conversion result is ready to be read. An interrupt is invoked if this is enabled. The flag must be cleared by software. It cannot be set by software.

FUNCTIONAL DESCRIPTION (continued)

ADCON.3 ADCS — ADC start and status: setting this bit starts an ADC conversion. It may be set by software or by the external signal STADC. The ADC logic ensures that this signal is HIGH while the ADC is busy. On completion of the conversion, ADCS is reset at the same time the interrupt flag ADCI is set. ADCS can not be reset by software.

ADCI	ADCS	OPERATION
0	0	ADC not busy, a conversion can be started
0	1	ADC busy, start of a new conversion is blocked
1	0	Conversion completed, start of a new conversion is blocked
1	1	Not possible

ADCON.2 AADR2
ADCON.1 AADR1
ADCON.0 AADR0

Analogue input select: this binary coded address selects one of the eight analogue port bits of P5 to be input to the converter. It can only be changed when ADCI and ADCS are both LOW. AADR2 is the most significant bit (100 selects the ADC4 analogue input channel).

The completion of the 10-bit ADC conversion is flagged by ADCI in the ADCON register and the result is stored in special function register ADCH (upper 8 bits) and the 2 LSBs in register ADCON.

An ADC conversion in progress is unaffected by an external or software ADC start. The result of a completed conversion remains unaffected provided ADCI = logic 1. While ADCS = logic 1 or ADCI = logic 1, a new ADC START will be blocked and consequently lost. An ADC conversion already in progress is aborted when the Idle or Power-down mode is entered. The result of a completed conversion (ADCI = logic 1) remains unaffected when entering the Idle mode.

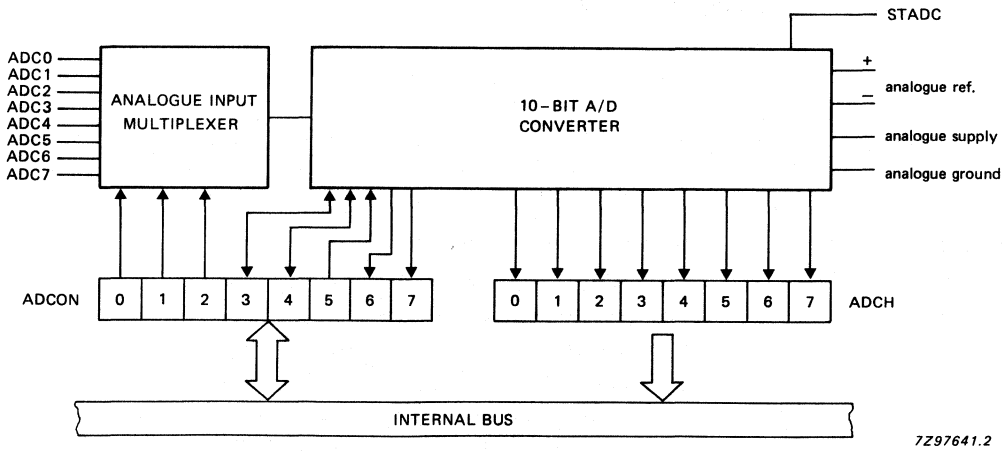


Fig. 8 Functional diagram of analogue input.

Timer/event counters

The PCB83C552 contains three 16-bit timer/event counters: Timer 0, Timer 1 and Timer T2 and one 8-bit timer, T3. Timer 0 and Timer 1 may be programmed to carry out the following functions:

- Measure time intervals and pulse durations
- Count events
- Generate interrupt requests

Timer 0 and Timer 1 can be programmed independently to operate in three modes:

- Mode 0: 8-bit timer or 8-bit counter each with divide-by-32 prescaler
- Mode 1: 16-bit time-interval or event counter
- Mode 2: 8-bit time-interval or event counter with automatic reload upon overflow

Timer 0 can be programmed to operate in an additional mode as follows:

- Mode 3: one 8-bit time-interval or event counter and one 8-bit time-interval counter

When Timer 0 is in Mode 3, Timer 1 can be programmed to operate in Modes 0, 1 or 2 but cannot set an interrupt request flag or generate an interrupt. However the overflow from Timer 1 can be used to pulse the serial port transmission-rate generator.

The frequency handling range of these counters with a 12 MHz crystal is as follows:

- In the timer function, the timer is incremented at a frequency of 1 MHz – a division by 12 of the oscillator frequency
- 0 Hz to an upper limit of 0,5 MHz when programmed for external inputs

Both internal and external inputs can be gated to the counter by a second external source for directly measuring pulse durations.

The counters are started and stopped under software control. Each one sets its interrupt request flag when it overflows from all logic 1s to all logic 0s (or automatic reload value), with the exception of mode 3 as previously described.

Timer T2

Timer T2 is a 16 bit timer/counter which has, coupled to it, capture and compare facilities. The operational diagram is shown in Fig. 9.

The 16 bit timer/counter is clocked via a prescaler with a programmable division factor of 1, 2, 4 or 8. The input of the prescaler is clocked with 1/12 of the oscillator frequency, or with positive edges on the T2 input, or it is switched to the off position. This prescaler is cleared if its division factor or its input source is changed, or if the timer/counter is reset. T2 is readable 'on the fly', but posseses no extra read latches; this means that software precautions have to be taken against misinterpretation in overflow from least to most significant byte during read. T2 is not loadable and is reset by the RST signal or at the positive edge of the input signal RT2, if enabled. In the Idle mode the timer/counter and prescaler are reset and halted.

T2 is connected to four 16-bit capture registers: CT0, CT1, CT2 and CT3. These registers are loaded with the contents of T2 and an interrupt is requested upon receipt of the input signals CT0I, CT1I, CT2I or CT3I. These input signals are shared with Port 1. Using the capture register CTCON, these inputs may capture on a positive edge, a negative edge or on either a positive or negative edge.

FUNCTIONAL DESCRIPTION (continued)

The contents of the compare registers CM0, CM1 and CM2 are continually compared with the counter value of Timer T2. When a match is found an interrupt may be invoked. Using the match signal of CM0, the controller sets bits 0-5 of Port 4, if the corresponding bits of the set enable register STE are logic 1.

Considering a match with CM1, if the corresponding bits of the reset/toggle enable register RTE are logic 1, then the controller will use the match signal to reset bits 0-5 of Port 4. Bits 6 and 7 of Port 4 may be 'toggled' by the signal that indicates a match between Timer T2 and CM2 if the corresponding bits of RTE are logic 1. CM0, CM1 and CM2 are reset by the RST signal.

Port 4 can be read and written by software without affecting the toggle, set and reset signals. At byte overflow of the least significant byte, or at a 16-bit overflow of the timer/counter, an interrupt sharing the same interrupt vector is requested. Either or both of these overflows can be programmed to request an interrupt.

All interupt flags must be reset by software.

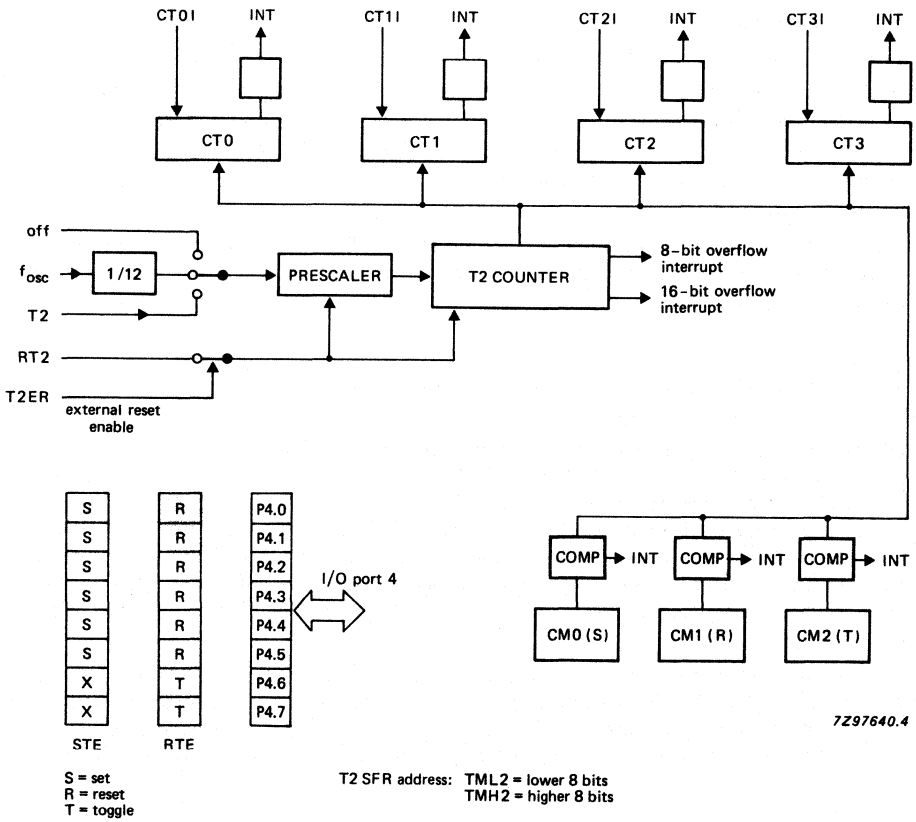


Fig. 9 Block diagram of Timer T2 configuration.

Counter control register TM2CON

	7	6	5	4	3	2	1	0
TM2CON (EAH)	T2IS1	T2IS0	T2ER	T2B0	T2P1	T2P0	T2MS1	T2MS0
	MSB							LSB

Bit	Symbol	Function
TM2CON.7	T2IS1	Timer 2 16 bit overflow interrupt select
TM2CON.6	T2IS0	Timer 2 byte overflow interrupt select
TM2CON.5	T2ER	Timer 2 external reset enable
TM2CON.4	T2B0	Timer 2 byte overflow interrupt flag
TM2CON.3	T2P1	Timer 2 prescaler select
TM2CON.2	T2P0	
TM2CON.1	T2MS1	Timer 2 mode select
TM2CON.0	T2MS0	

T2P1	T2P0	T2 clock
0	0	clock source
0	1	clock source/2
1	0	clock source/4
1	1	clock source/8

T2MS1	T2MS0	Mode Selected
0	0	Timer T2 is halted
0	1	T2 clock source = $f_{osc}/12$
1	0	Test mode; do not use
1	1	T2 clock source = pin T2

Capture control register CTCON

	7	6	5	4	3	2	1	0
CTCON (EBH)	CTN3	CTP3	CTN2	CTP2	CTN1	CTP1	CTN0	CTP0
	MSB							LSB

Bit	Symbol	Capture/Interrupt on:
CTCON.7	CTN3	CT3I negative edge
CTCON.6	CTP3	CT3I positive edge
CTCON.5	CTN2	CT2I negative edge
CTCON.4	CTP2	CT2I positive edge
CTCON.3	CTN1	CT1I negative edge
CTCON.2	CTP1	CT1I positive edge
CTCON.1	CTN0	CT0I negative edge
CTCON.0	CTP0	CT0I positive edge

DEVELOPMENT DATA

FUNCTIONAL DESCRIPTION (continued)

Timer interrupt flag register TM2IR

	7	6	5	4	3	2	1	0
TM2IR (C8H)	T20V	CMI2	CMI1	CMI0	CTI3	CTI2	CTI1	CTI0
	MSB							LSB

Bit	Symbol	Function
TM2IR.7	T20V	T2 16-bit overflow interrupt flag
TM2IR.6	CMI2	CM2 interrupt flag
TM2IR.5	CMI1	CM1 interrupt flag
TM2IR.4	CMI0	CM0 interrupt flag
TM2IR.3	CTI3/INT5	CT3 interrupt flag
TM2IR.2	CTI2/INT4	CT2 interrupt flag
TM2IR.1	CTI1/INT3	CT1 interrupt flag
TM2IR.0	CTI0/INT2	CT0 interrupt flag

Interrupt enable register IEN1 is used to enable/disable timer 2 interrupts.
Interrupt priority register IP1 is used to determine the timer interrupt priority.

The set enable register STE

	7	6	5	4	3	2	1	0
STE (EEH)	TG4.7	TG4.6	SP45	SP44	SP43	SP42	SP41	SP40
	MSB							LSB

Bit	Symbol	Function
STE.7	TG4.7	
STE.6	TG4.6	
STE.5	SP45	if "1" then P4.5 is set on a match between CM0 and T2
STE.4	SP44	if "1" then P4.4 is set on a match between CM0 and T2
STE.3	SP43	if "1" then P4.3 is set on a match between CM0 and T2
STE.2	SP42	if "1" then P4.2 is set on a match between CM0 and T2
STE.1	SP41	if "1" then P4.1 is set on a match between CM0 and T2
STE.0	SP40	if "1" then P4.0 is set on a match between CM0 and T2

If STE.n is "0" then P4.n is not affected by a match between CM0 and T2

The reset/toggle enable register RTE

	7	6	5	4	3	2	1	0
RTE (EFH)	TP47	TP46	RP45	RP44	RP43	RP42	RP41	RP40
	MSB							LSB

Bit	Symbol	Function
RTE.7	TP47	if "1" then P4.7 toggles on a match between CM2 and T2
RTE.6	TP46	if "1" then P4.6 toggles on a match between CM2 and T2
RTE.5	RP45	if "1" then P4.5 is reset on a match between CM1 and T2
RTE.4	RP44	if "1" then P4.4 is reset on a match between CM1 and T2
RTE.3	RP43	if "1" then P4.3 is reset on a match between CM1 and T2
RTE.2	RP42	if "1" then P4.2 is reset on a match between CM1 and T2
RTE.1	RP41	if "1" then P4.1 is reset on a match between CM1 and T2
RTE.0	RP40	if "1" then P4.0 is reset on a match between CM1 and T2

If RTE.n is "0" then P4.n is not affected by a match between CM1 and T2 or CM2 and T2

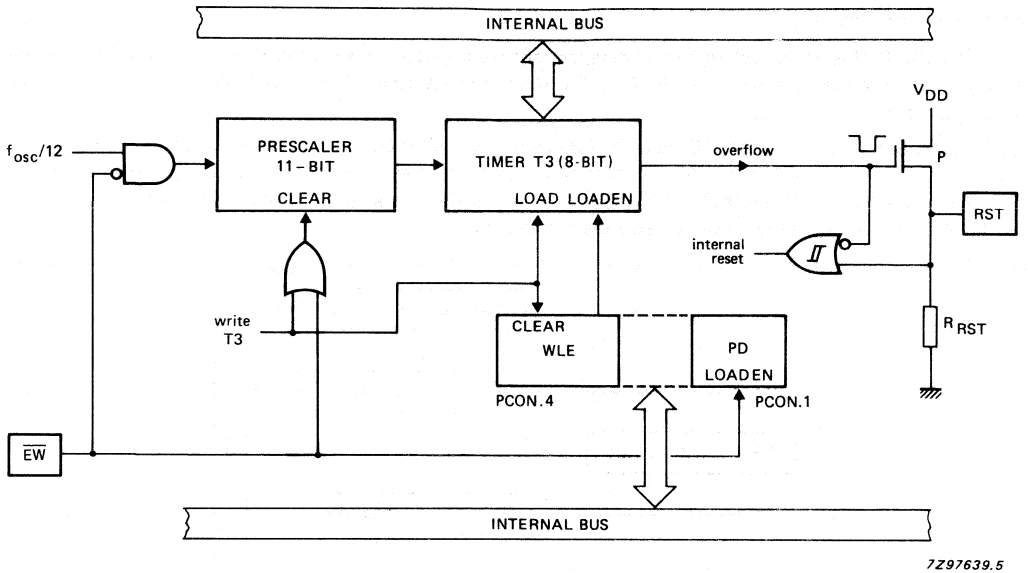


Fig. 10 Functional diagram of T3 watchdog timer.

T3 The watchdog timer (see Fig. 10)

In addition to Timer T2 and the standard timers, a watchdog timer consisting of an 11-bit prescaler and an 8-bit timer are also incorporated.

The timer is incremented every 2 ms, derived from the oscillator frequency of 12 MHz by the following:

$$f_{timer} = \frac{f_{osc}}{12 \times 2048}$$

When a timer overflow occurs, the microcontroller is reset and a reset output pulse is generated at pin RST. To prevent a system reset the timer must be reloaded in time by the application software. If the processor suffers a hardware/software malfunction, the software will fail to reload the timer. This failure will produce a reset upon overflow thus preventing the processor running out of control.

The watchdog timer can only be reloaded if the condition flag WLE = PCON.4 has been previously set by software.

At the moment the counter is loaded the condition flag is automatically cleared.

The time interval between the timer's reloading and occurance of a reset, is dependent upon the reloaded value. This may range from 2 ms to 0,5 s at an oscillator frequency of 12 MHz.

In the Idle state the watchdog timer and reset circuitry remain active.

The watchdog timer is controlled by the watchdog enable pin ($\overline{EW}$). A logic 0 enables the watchdog timer and disables the Power-down mode. A logic 1 disables and resets the watchdog timer and enables the Power-down mode.

DEVELOPMENT DATA

FUNCTIONAL DESCRIPTION (continued)

Serial I/O (see Fig. 11)

The PCB83C552 is equipped with two independent serial ports. SIO0 is the full duplex UART port and is identical to the serial port of the PCB80C51 (see "SINGLE-CHIP 8-BIT MICROCONTROLLERS" User Manual).

Serial port SIO1 supports the I^2C-bus, the function of which is controlled by the S1CON register. S1STA is the status register whose contents may also be used as a vector to various service routines. S1DAT is the data shift register and S1ADR the slave address register. The least significant bit of S1ADR enables/disables general call address recognition.

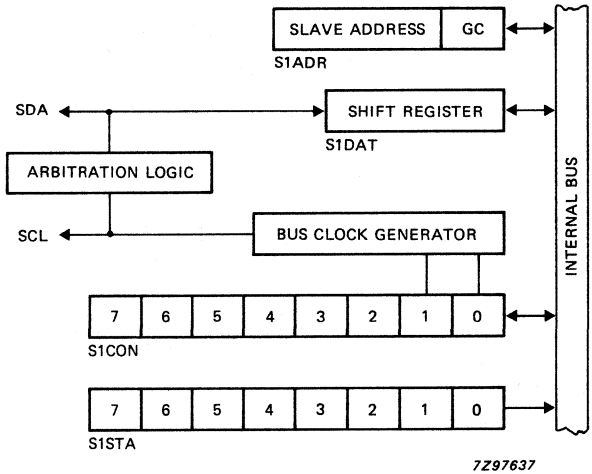


Fig. 11 Block diagram of I^2C serial I/O.

The I^2C serial I/O has complete autonomy in byte handling and operates in 4 modes:

1. Master transmitter
2. Master receiver
3. Slave transmitter
4. Slave receiver

Slave address recognition is performed by on-chip hardware.

The I^2C-bus consists of two lines: a data line (SDA) and a clock line (SCL). These lines also function as the I/O port lines on P1.7 and P1.6. The system is unique because data transport, clock generation, address recognition and bus control arbitration are all controlled by hardware.

Serial control register S1CON

	7	6	5	4	3	2	1	0
S1CON (D8H)	CR2	ENS1	STA	STO	SI	AA	CR1	CR0

Bits CR0, CR1 and CR2 determine the serial clock frequency that is generated in the master mode of operation.

Table 1 Serial clock rates

CR2 (1)	CR1	CR0	bit frequency (kHz) at f_{OSC}:			f_{OSC} divided by
			6 MHz	12 MHz	16 MHz (1)	
0	0	0	23	47	63	256 (1)
0	0	1	27	54	71	224 (1)
0	1	0	31	63	83	192 (1)
0	1	1	37	75	100	160 (1)
1	0	0	6.25	12.5	17	960
1	0	1	50	100	133 (2)	120
1	1	0	100	200 (2)	267 (2)	60
1	1	1	0.25 < 62.5	0.5 < 62.5	0.67 < 56	96 x (256 - reload value Timer 1) (reload value range: 0 - 254 in mode 2)

Notes to Table 1

(1) A maximum frequency of 16 MHz will be implemented in a later version. At present control bit CR2 is not used.

(2) These frequencies exceed the upper limit of 100 kHz of the I^2C-bus specification and cannot be used in an I^2C-bus application.

AA

Assert acknowledge bit. When this bit is set, an acknowledge is returned after any one of the following conditions:

- Own slave address is received
- General call address is received (S1ADR.0 = logic 1)
- A data byte is received, while the device is programmed to be a master receiver
- A data byte is received, while the device is a selected slave receiver

When this bit is reset, no acknowledge is returned. Consequently, no interrupt is requested when the own address or general call address is received.

SI

SIO1 interrupt flag. This flag is set, and an interrupt request is generated, after any of the following events occur:

- A START condition is generated in MST mode
- The own slave address has been received during AA = logic 1
- The general call address has been received while S1ADR.0 and AA = logic 1
- A data byte has been received or transmitted in MST mode (even if arbitration is lost)
- A data byte has been received or transmitted as selected slave
- A STOP or START condition is received as selected slave receiver or transmitter

FUNCTIONAL DESCRIPTION (continued)

STO

STOP flag. When in master mode, and this bit is set a STOP condition is generated. A STOP condition detected on the I^2C-bus clears this bit. This bit may also be set in slave mode in order to recover from an error condition. Then no STOP condition is generated to the I^2C-bus, but the hardware releases the SDA and SCL lines and switches to the not selected slave receiver mode. The STOP flag is cleared by the hardware.

STA

START flag. When this bit is set in slave mode, the hardware checks the I^2C-bus and generates a START condition if the bus is free or after the bus becomes free. If the device operates in master mode it will generate a repeated START condition.

ENS1

0 = Serial I/O Disabled and reset. P1.6 and P1.7 I/O port function with open drain

1 = Serial I/O Enabled. Output ports P1.6 and P1.7 must be set to logic 1

Serial status register S1STA (S1STA is a read-only register)

	7	6	5	4	3	2	1	0
S1STA (D9H)	SC4	SC3	SC2	SC1	SC0	0	0	0

S1STA.3 - S1STA.7 hold a status code. S1STA.0 - S1STA.2 are held LOW. The contents of the status register may be used as a vector to a service routine. This optimizes the response time of the software and consequently that of the I^2C-bus.

Abbreviations used:

SLA	:	7-bit slave address
R	:	Read bit
W	:	Write bit
ACK	:	Acknowledgement (acknowledge bit = logic 0)
$\overline{\text{ACK}}$	:	Not acknowledgement (acknowledge bit = logic 1)
DATA	:	8-bit data byte to or from I^2C-bus
MST	:	Master
SLV	:	Slave
TRX	:	Transmitter
REC	:	Receiver

The following is a list of the status codes:

MST/TRX mode

S1STA value

08H	– A START condition has been transmitted
10H	– A repeated START condition has been transmitted
18H	– SLA and W have been transmitted, ACK has been received
20H	– SLA and W have been transmitted, $\overline{\text{ACK}}$ received
28H	– DATA of S1DAT has been transmitted, ACK received
30H	– DATA of S1DAT has been transmitted, $\overline{\text{ACK}}$ received
38H	– Arbitration lost in SLA, R/W or DATA

MST/REC mode

S1STA value

38H – Arbitration lost while returning $\overline{\text{ACK}}$
40H – SLA and R have been transmitted, ACK received
48H – SLA and R have been transmitted, $\overline{\text{ACK}}$ received
50H – DATA has been received, ACK returned
58H – DATA has been received, $\overline{\text{ACK}}$ returned

SLV/REC mode

S1STA value

60H – Own SLA and W have been received, ACK returned
68H – Arbitration lost in SLA, R/W as MST. Own SLA and W have been received, ACK returned
70H – General CALL has been received, ACK returned
78H – Arbitration lost in SLA, R/W as MST. General call has been received
80H – Previously addressed with own SLA. DATA byte received, ACK returned
88H – Previously addressed with own SLA. DATA byte received, $\overline{\text{ACK}}$ returned
90H – Previously addressed with general call. DATA byte has been received, ACK has been returned
98H – Previously addressed with general call. DATA byte has been received, $\overline{\text{ACK}}$ has been returned
A0H – A STOP condition or repeated START condition has been received while still addressed as SLV/REC or SLV/TRX.

SLV/TRX mode

S1STA value

A8H – Own SLA and R have been received, ACK returned
B0H – Arbitration lost in SLA, R/W as MST. Own SLA and R have been received, ACK returned
B8H – DATA byte has been transmitted, ACK received
C0H – DATA byte has been transmitted, $\overline{\text{ACK}}$ received
C8H – Last DATA byte has been transmitted (AA = logic 0), ACK received

Miscellaneous

S1STA value

00H – Bus error during MST mode or selected SLV mode, due to an erroneous START or STOP condition

The data shift register S1DAT

S1DAT (DAH)	7	6	5	4	3	2	1	0

This register contains the serial data to be transmitted or data which has just been received. Bit 7 is transmitted or received first; i.e. data is shifted from right to left.

Address register S1ADR

S1ADR (DBH)	7	6	5	4	3	2	1	0

S1ADR.0, GC : 0 = general call address is not recognized
1 = general call address is recognized

S1ADR.7-1 : own slave address

This 8-bit register may be loaded with the 7-bit slave address to which the controller will respond when programmed as a slave receiver/transmitter. The LSB (GC) is used to determine whether the general call address is recognized.

DEVELOPMENT DATA

FUNCTIONAL DESCRIPTION (continued)

Idle and Power-down operation (see Fig. 12)

Idle mode operation permits the interrupt, serial ports and timer blocks to continue to function while the CPU is halted. The following functions are also switched off when the processor enters the Idle mode.

Timer T2	(stopped and reset)
PWM0, PWM1	(reset, output = HIGH)
ADC	(aborted if in progress)

The following functions remain active during Idle mode. These functions may generate an interrupt or reset and thus end the Idle mode.

Timer 0, Timer 1
Timer T3
SIO0, SIO1
External interrupt

The Power-down operation freezes the oscillator. The Power-down mode can only be activated by setting the PD bit in the PCON register. The PD bit can only be set if the EWN input is HIGH.

Idle mode

The instruction that sets PCON.0 is the last instruction executed in the normal operating mode before Idle mode is activated. Once in the Idle mode, the CPU status is preserved in its entirety: the Stack Pointer, Program Counter, Program Status Word, Accumulator, RAM and all other registers maintain their data during Idle mode. The status of the external pins during Idle mode is shown in Table 2.

There are two ways to terminate the Idle mode:

Activation of any enabled interrupt will cause PCON.0 to be cleared by hardware terminating Idle mode. The interrupt is serviced, and following return from interrupt instruction RETI, the next instruction to be executed will be the one which follows the instruction that wrote a logic 1 to PCON.0.

The flag bits GF0 and GF1 may be used to determine whether the interrupt was received during normal execution or during the Idle mode. For example, the instruction that writes to PCON.0 can also set or clear one or both flag bits. When Idle mode is terminated by an interrupt, the service routine can examine the status of the flag bits.

The second way of terminating the Idle mode is with an external hardware reset, or an internal reset caused by an overflow of Timer T3. Since the oscillator is still running, the hardware reset is required to be active for two machine cycles (24 oscillator periods) to complete the reset operation.

Power-down mode

The instruction that sets PCON.1 is the last executed prior to going into the Power-down mode. Once in Power-down mode, the oscillator is stopped. Only the contents of the on-chip RAM are preserved. The Special Function Registers are not saved. A hardware reset is the only way of exiting the Power-down mode.

In the Power-down mode, V_{DD} may be reduced to minimize circuit power consumption. The voltage must not be reduced until the Power-down mode is entered, but must be restored before the hardware reset is applied which will free the oscillator. Reset should not be released until the oscillator has restarted and stabilized.

The status of the external pins during Power-down mode is shown in Table 2. If the Power-down mode is activated while in external program memory, the port data that is held in the Special Function Register P2 is restored to Port 2. If the data is a logic 1, the port pin is held HIGH during the Power-down mode by the strong pull-up transistor p1 (see Fig. 6).

Table 2 Status of external pins during Idle and Power-down modes

mode	memory	ALE	$\overline{PSEN}$	Port 0	Port 1	Port 2	Port 3	Port 4	$\overline{PWM0}/\overline{PWM1}$
Idle (1)	internal	1	1	port data	port data	port data	port data	port data	HIGH
Idle (1)	external	1	1	floating	port data	address	port data	port data	HIGH
Power-down	internal	0	0	port data	port data	port data	port data	port data	HIGH
Power-down	external	0	0	floating	port data	port data	port data	port data	HIGH

Power control register (PCON)

These special modes are activated by software via the Special Function Register PCON. Its hardware address is 87H. PCON is not bit addressable.

	7	6	5	4	3	2	1	0
PCON (87H)	SMOD	–	–	WLE	GF1	GF0	PD	IDL

Bit	Symbol	Function
PCON.7	SMOD	Double Baud rate bit. When set to logic 1 the baud rate is doubled when the serial port SIO0 is being used in modes 1, 2 or 3
PCON.6	–	(reserved)
PCON.5	–	(reserved)
PCON.4	WLE	Watchdog load enable. This flag must be set by software prior to loading T3 (watchdog timer). It is cleared when T3 is loaded
PCON.3	GF1	General-purpose flag bit
PCON.2	GF0	General-purpose flag bit
PCON.1	PD	Power-down bit. Setting this bit activates Power-down mode. It can only be set if input $\overline{EW}$ is high.
PCON.0	IDL	Idle mode bit. Setting this bit activates the Idle mode

If logic 1s are written to PD and IDL at the same time, PD takes precedence. The reset value of PCON is (0XX00000).

DEVELOPMENT DATA

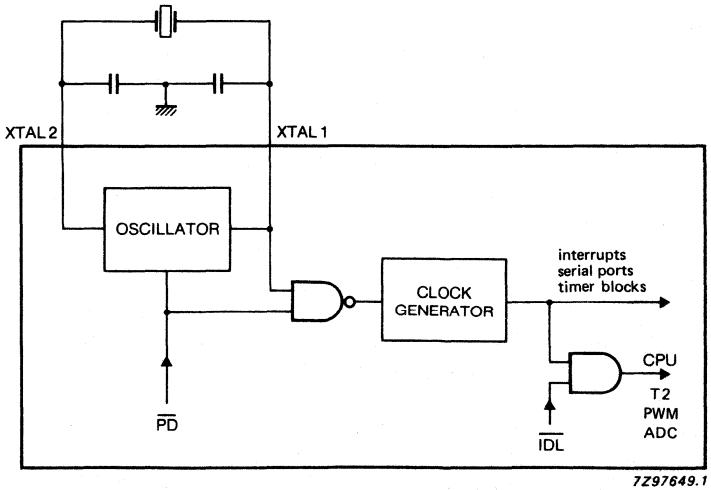


Fig. 12 Internal Idle and Power-down clock configuration.

FUNCTIONAL DESCRIPTION (continued)

Interrupt system (see Fig. 13)

External events and the real-time-driven on-chip peripherals require service by the CPU asynchronous to the execution of any particular section of code. To tie the asynchronous activities of these functions to normal program execution a multiple-source, two-priority-level, nested interrupt system is provided. Interrupt response latency is from 3 μs to 8 μs when using a 12 MHz crystal. The PCB83C552 acknowledges interrupt requests from fifteen sources as follows:

- $\overline{\text{INT0}}$ and $\overline{\text{INT1}}$: externally via pins 26 and 27 respectively
- INT2 to INT5: externally via pins 16 to 19 respectively. These pins are shared with the four capture inputs of Timer T2.
- Timer 0 and Timer 1: from the two internal counters
- Timer T2 (8 separate interrupts): 4 capture interrupts, 3 compare interrupts and an overflow interrupt
- ADC end-of-conversion interrupt
- I^2C serial I/O port interrupt
- UART serial I/O port interrupt.

Each interrupt vectors to a separate location in program memory for its service program. Each source can be individually enabled or disabled by a corresponding bit in the IE0 or IE1 register, moreover each interrupt may be programmed to a high or low priority level using a corresponding bit in the IP0 or IP1 register. Also all enabled sources can be globally disabled or enabled. Both external interrupts can be programmed to be level-activated or transition-activated, and an active LOW level allows "wire-ORing" of several interrupt sources to the input pin.

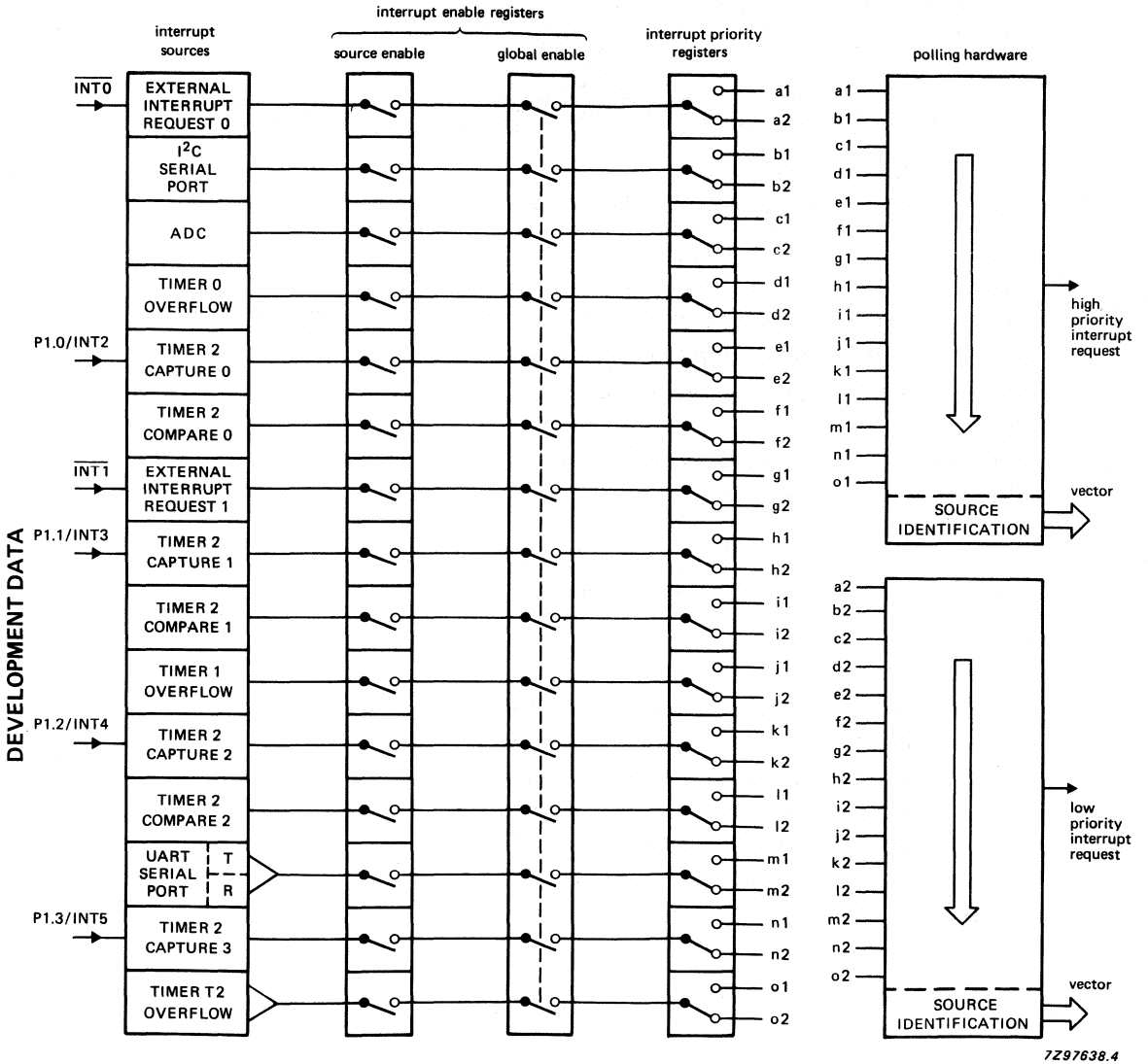


Fig. 13 Interrupt system.

FUNCTIONAL DESCRIPTION (continued)

Interrupt enable registers

	7	6	5	4	3	2	1	0
IEN0 (A8H)	EA	EAD	ES1	ES0	ET1	EX1	ET0	EX0

Bit	Symbol	Function
IEN0.7	EA	General enable/disable control 0 = No interrupt is enabled 1 = Any individually enabled interrupt will be accepted
IEN0.6	EAD	Enable ADC interrupt
IEN0.5	ES1	Enable SIO1 (I^2C) interrupt
IEN0.4	ES0	Enable SIO0 (UART) interrupt
IEN0.3	ET1	Enable Timer 1 interrupt
IEN0.2	EX1	Enable External 1 interrupt
IEN0.1	ET0	Enable Timer 0 interrupt
IEN0.0	EX0	Enable External 0 interrupt

	7	6	5	4	3	2	1	0
IEN1 (E8H)	ET2	ECM2	ECM1	ECM0	ECT3	ECT2	ECT1	ECT0

Bit	Symbol	Function
IEN1.7	ET2	Enable T2 overflow interrupt(s)
IEN1.6	ECM2	Enable T2 comparator 2 interrupt
IEN1.5	ECM1	Enable T2 comparator 1 interrupt
IEN1.4	ECM0	Enable T2 comparator 0 interrupt
IEN1.3	ECT3	Enable T2 capture register 3 interrupt
IEN1.2	ECT2	Enable T2 capture register 2 interrupt
IEN1.1	ECT1	Enable T2 capture register 1 interrupt
IEN1.0	ECT0	Enable T2 capture register 0 interrupt

where 0 = interrupt disabled
and 1 = interrupt enabled

Interrupt priority registers

	7	6	5	4	3	2	1	0
IP0 (B8H)	–	PAD	PS1	PS0	PT1	PX1	PT0	PX0

Bit	Symbol	Function
IP0.7	–	Unused
IP0.6	PAD	ADC interrupt priority level
IP0.5	PS1	SIO1 (I^2C) interrupt priority level
IP0.4	PS0	SIO0 (UART) interrupt priority level
IP0.3	PT1	Timer 1 interrupt priority level
IP0.2	PX1	External interrupt 1 priority level
IP0.1	PT0	Timer 0 interrupt priority level
IP0.0	PX0	External interrupt 0 priority level

	7	6	5	4	3	2	1	0
IP1 (F8H)	PT2	PCM2	PCM1	PCM0	PCT3	PCT2	PCT1	PCT0

Bit	Symbol	Function
IP1.7	PT2	T2 overflow interrupt(s) priority level
IP1.6	PCM2	T2 comparator 2 priority interrupt level
IP1.5	PCM1	T2 comparator 1 priority interrupt level
IP1.4	PCM0	T2 comparator 0 priority interrupt level
IP1.3	PCT3	T2 capture register 3 priority interrupt level
IP1.2	PCT2	T2 capture register 2 priority interrupt level
IP1.1	PCT1	T2 capture register 1 priority interrupt level
IP1.0	PCT0	T2 capture register 0 priority interrupt level

Interrupt priority levels are as follow

0 – low priority
1 – high priority

FUNCTIONAL DESCRIPTION (continued)

Table 3 shows the interrupt vectors. The vector indicates the ROM location where the appropriate interrupt service routine starts.

Table 3 Interrupt vectors

Source		Vector
External 0	X0	0003H
Timer 0 overflow	T0	000BH
External 1	X1	0013H
Timer 1 overflow	T1	001BH
Serial I/O 0 (UART)	S0	0023H
Serial I/O 1 (I^2C)	S1	002BH
T2 capture 0	CT0	0033H
T2 capture 1	CT1	003BH
T2 capture 2	CT2	0043H
T2 capture 3	CT3	004BH
ADC completion	ADC	0053H
T2 compare 0	CM0	005BH
T2 compare 1	CM1	0063H
T2 compare 2	CM2	006BH
T2 overflow	T2	0073H

Interrupt priority

Each interrupt source can be either high priority or low priority. If both priorities are requested simultaneously, the processor will branch to the high priority vector. If there are simultaneous requests from sources of the same priority, then interrupts will be serviced in the following order:

X0, S1, ADC, T0, CT0, CM0, X1, CT1, CM1, T1, CT2, CM2, S0, CT3, T2

A low priority interrupt routine can only be interrupted by a high priority interrupt. A high priority interrupt routine can not be interrupted.

Oscillator circuitry

The oscillator circuitry of the PCB83C552 is a single-stage inverting amplifier in a Pierce oscillator configuration. The circuitry between XTAL 1 and XTAL 2 is basically an inverter biased to the transfer point. Either a crystal or ceramic resonator can be used as the feedback element to complete the oscillator circuitry. Both are operated in parallel resonance. XTAL 1 (pin 35) is the high gain amplifier input, and XTAL 2 (pin 34) is the output (see Fig. 14). To drive the PCB83C552 externally, XTAL 1 is driven from an external source and XTAL 2 left open-circuit (see Fig. 15).

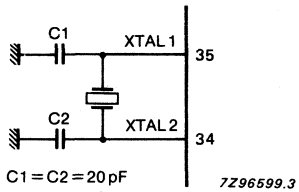


Fig. 14 PCB83C552 oscillator circuit.

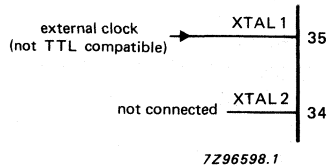


Fig. 15 Driving the PCB83C552 from an external source.

DEVELOPMENT DATA

Reset circuitry (see Fig. 16)

The reset circuitry for the PCB83C552 is connected to the reset pin RST. A Schmitt trigger is used at the input for noise rejection. The output of the Schmitt trigger is sampled by the reset circuitry every machine cycle.

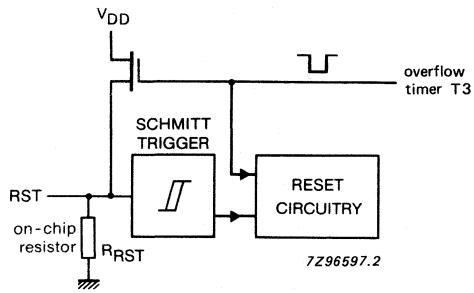


Fig. 16 On-chip reset configuration.

A reset is accomplished by holding the RST pin HIGH for at least two machine cycles (24 oscillator periods). The CPU responds by executing an internal reset. During reset ALE and $\overline{PSEN}$ output a HIGH level. In order to perform a correct reset, this level must not be affected by external elements.

Also with the PCB83C552, the RST line can be pulled HIGH internally by a pull-up transistor activated by the watchdog timer T3. The length of the output pulse from T3 is 3 machine cycles. A pulse of such short duration is necessary in order to recover from a processor or system fault as fast as possible.

FUNCTIONAL DESCRIPTION (continued)

N.B. It can be seen that the short reset pulse from Timer T3 cannot discharge the power-on reset capacitor (see Fig. 17). Consequently, when the watchdog timer is also used to reset external devices this capacitor arrangement should not be connected to the RST pin, and an extra circuit should be used to perform the power-on reset operation. It should be remembered that a Timer T3 overflow, if enabled, will force a reset condition to the PCB83C552 by an internal connection, whether the output RST is tied LOW or not.

The internal reset is executed during the second cycle in which RST is HIGH and is repeated every cycle until RST goes LOW. It leaves the internal registers as follows:

Register	Content
ACC	0000 0000
ADCON	xx00 0000
ADCH	xxxx xxxx
B	0000 0000
CML0 - CML2	0000 0000
CMH0 - CMH2	0000 0000
CTCON	0000 0000
CTL0 - CTL3	xxxx xxxx
CTH0 - CTH3	xxxx xxxx
DPL	0000 0000
DPH	0000 0000
IEN0	0000 0000
IEN1	0000 0000
IP0	x000 0000
IP1	0000 0000
PCH	0000 0000
PCL	0000 0000
PCON	0xx0 0000
PSW	0000 0000
PWM0	0000 0000
PWM1	0000 0000
PWMP	0000 0000
P0 - P4	1111 1111
P5	xxxx xxxx
RTE	0000 0000
S0BUF	xxxx xxxx
S0CON	0000 0000
S1ADR	0000 0000
S1CON	0000 0000
S1DAT	0000 0000
S1STA	1111 1000
SP	0000 0111
STE	1100 0000
TCON	0000 0000
TH0, TH1	0000 0000
TMH2	0000 0000
TL0, TL1	0000 0000
TML2	0000 0000
TMOD	0000 0000
TM2CON	0000 0000
TM2IR	0000 0000
T3	0000 0000

x = undefined state.

The internal RAM is not affected by reset. When V_{DD} is turned on, the RAM content is indeterminate.

Power-on reset (see Fig. 17)

When V_{DD} is turned on, and provided its rise-time does not exceed 10 ms, an automatic reset can be obtained by connecting the RST pin to V_{DD} via a 2,2 μF capacitor. When the power is switched on, the voltage on the RST pin is equal to V_{DD} minus the capacitor voltage, and decreases from V_{DD} as the capacitor charges through the internal resistor (R_{RST}) to ground. The larger the capacitor, the more slowly V_{RST} decreases. V_{RST} must remain above the lower threshold of the Schmitt trigger long enough to effect a complete reset. The time required is the oscillator start-up time, plus 2 machine cycles.

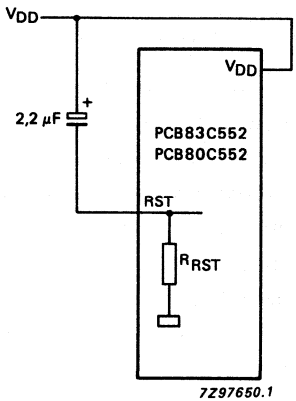


Fig. 17 Power-on reset.

INSTRUCTION SET

Table 4 Instruction set description

mnemonic		description	bytes/ cycles		opcode (hex.)
Arithmetic operations					
ADD	A,Rr	Add register to A	1	1	2*
ADD	A,direct	Add direct byte to A	2	1	25
ADD	A,@Ri	Add indirect RAM to A	1	1	26, 27
ADD	A,#data	Add immediate data to A	2	1	24
ADDC	A,Rr	Add register to A with carry flag	1	1	3*
ADDC	A,direct	Add direct byte to A with carry flag	2	1	35
ADDC	A,@Ri	Add indirect RAM to A with carry flag	1	1	36, 37
ADDC	A,#data	Add immediate data to A with carry flag	2	1	34
SUBB	A,Rr	Subtract register from A with borrow	1	1	9*
SUBB	A,direct	Subtract direct byte from A with borrow	2	1	95
SUBB	A,@Ri	Subtract indirect RAM from A with borrow	1	1	96, 97
SUBB	A,#data	Subtract immediate data from A with borrow	2	1	94
INC	A	Increment A	1	1	04
INC	Rr	Increment register	1	1	0*
INC	direct	Increment direct byte	2	1	05
INC	@Ri	Increment indirect RAM	1	1	06, 07
DEC	A	Decrement A	1	1	14
DEC	Rr	Decrement register	1	1	1*
DEC	direct	Decrement direct byte	2	1	15
DEC	@Ri	Decrement indirect RAM	1	1	16, 17
INC	DPTR	Increment data pointer	1	2	A3
MUL	AB	Multuply A & B	1	4	A4
DIV	AB	Divide A by B	1	4	84
DA	A	Decimal adjust A	1	1	D4

mnemonic		description	bytes/ cycles		opcode (hex.)
Logic operations					
ANL	A,Rr	AND register to A	1	1	5*
ANL	A,direct	AND direct byte to A	2	1	55
ANL	A,@Ri	AND indirect RAM to A	1	1	56, 57
ANL	A,#data	AND immediate data to A	2	1	54
ANL	direct,A	AND A to direct byte	2	1	52
ANL	direct,#data	AND immediate data to direct byte	3	2	53
ORL	A,Rr	OR register to A	1	1	4*
ORL	A,direct	OR direct byte to A	2	1	45
ORL	A,@Ri	OR indirect RAM to A	1	1	46, 47
ORL	A,#data	OR immediate data to A	2	1	44
ORL	direct, A	OR A to direct byte	2	1	42
ORL	direct,#data	OR immediate data to direct byte	3	2	43
XRL	A,Rr	Exclusive-OR register to A	1	1	6*
XRL	A,direct	Exclusive-OR direct byte to A	2	1	65
XRL	A,@Ri	Exclusive-OR indirect RAM to A	1	1	66, 67
XRL	A,#data	Exclusive-OR immediate data to A	2	1	64
XRL	direct,A	Exclusive-OR A to direct byte	2	1	62
XRL	direct,#data	Exclusive-OR immediate data to direct byte	3	2	63
CLR	A	Clear A	1	1	E4
CPL	A	Complement A	1	1	F4
RL	A	Rotate A left	1	1	23
RLC	A	Rotate A left through the carry flag	1	1	33
RR	A	Rotate A right	1	1	03
RRC	A	Rotate A right through the carry flag	1	1	13
SWAP	A	Swap nibbles within A	1	1	C4

INSTRUCTION SET (continued)

mnemonic		description	bytes/ cycles		opcode (hex.)
Data transfer					
MOV	A,Rr	Move register to A	1	1	E*
MOV	A,direct**	Move direct byte to A	2	1	E5
MOV	A,@Ri	Move indirect RAM to A	1	1	E6, E7
MOV	A,#data	Move immediate data to A	2	1	74
MOV	Rr,A	Move A to register	1	1	F*
MOV	Rr,direct	Move direct byte to register	2	2	A*
MOV	Rr,#data	Move immediate data to register	2	1	7*
MOV	direct,A	Move A to direct byte	2	1	F5
MOV	direct,Rr	Move register to direct byte	2	2	8*
MOV	direct,direct	Move direct byte to direct byte	3	2	85
MOV	direct,@Ri	Move indirect RAM to direct byte	2	2	86, 87
MOV	direct,#data	Move immediate data to direct byte	3	2	75
MOV	@Ri,A	Move A to indirect RAM	1	1	F6, F7
MOV	@Ri,direct	Move direct byte to indirect RAM	2	2	A6, A7
MOV	@Ri,#data	Move immediate data to indirect RAM	2	1	76, 77
MOV	DPTR,#data16	Load data pointer with a 16-bit constant	3	2	90
MOVC	A,@A+DPTR	Move code byte relative to DPTR to A	1	2	93
MOVC	A,@A+PC	Move code byte relative to PC to A	1	2	83
MOVX	A,@Ri	Move external RAM (8-bit address) to A	1	2	E2, E3
MOVX	A,@DPTR	Move external RAM (16-bit address) to A	1	2	E0
MOVX	@Ri,A	Move A to external RAM (8-bit address)	1	2	F2, F3
MOVX	@DPTR,A	Move A to external RAM (16-bit address)	1	2	F0
PUSH	direct	Push direct byte onto stack	2	2	C0
POP	direct	Pop direct byte from stack	2	2	D0
XCH	A,Rr	Exchange register with A	1	1	C*
XCH	A,direct	Exchange direct byte with A	2	1	C5
XCH	A,@Ri	Exchange indirect RAM with A	1	1	C6, C7
XCHD	A,@Ri	Exchange LOW-order nibble indirect RAM with A	1	1	D6, D7

** MOV A,ACC is not a valid instruction.

mnemonic		description	bytes/ cycles		opcode (hex.)
Boolean variable manipulation					
CLR	C	Clear carry flag	1	1	C3
CLR	bit	Clear direct bit	2	1	C2
SETB	C	Set carry flag	1	1	D3
SETB	bit	Set direct bit	2	1	D2
CPL	C	Complement carry flag	1	1	B3
CPL	bit	Complement direct bit	2	1	B2
ANL	C,bit	AND direct bit to carry flag	2	2	82
ANL	C,/bit	AND complement of direct bit to carry flag	2	2	B0
ORL	C,bit	OR direct bit to carry flag	2	2	72
ORL	C,/bit	OR complement of direct bit to carry flag	2	2	A0
MOV	C,bit	Move direct bit to carry flag	2	1	A2
MOV	bit,C	Move carry flag to direct bit	2	2	92
Program and machine control					
ACALL	addr11	Absolute subroutine call	2	2	●1addr
LCALL	addr16	Long subroutine call	3	2	12
RET		Return from subroutine	1	2	22
RETI		Return from interrupt	1	2	32
AJMP	addr11	Absolute jump	2	2	▲1addr
LJMP	addr16	Long jump	3	2	02
SJMP	rel	Short jump (relative address)	2	2	80
JMP	@A+DPTR	Jump indirect relative to the DPTR	1	2	73
JZ	rel	Jump if A is zero	2	2	60
JNZ	rel	Jump if A is not zero	2	2	70
JC	rel	Jump if carry flag is set	2	2	40
JNC	rel	Jump if carry flag is not set	2	2	50
JB	bit,rel	Jump if direct bit is set	3	2	20
JNB	bit,rel	Jump if direct bit is not set	3	2	30
JBC	bit,rel	Jump if direct bit is set and clear bit	3	2	10
CJNE	A,direct,rel	Compare direct to A and jump if not equal	3	2	B5
CJNE	A,#data,rel	Compare immediate to A and jump if not equal	3	2	B4
CJNE	Rr,#data,rel	Compare immediate to reg. and jump if not equal	3	2	B*
CJNE	@Ri,#data,rel	Compare immediate to ind. and jump if not equal	3	2	B6, B7
DJNZ	Rr,rel	Decrement register and jump if not zero	2	2	D*
DJNZ	direct,rel	Decrement direct and jump if not zero	3	2	D5
NOP		No operation	1	1	00

Notes to Table 4

Data addressing modes

Rr	Working register R0-R7.
direct	128 internal RAM locations and any special function register (SFR).
@Ri	Indirect internal RAM location addressed by register R0 or R1 of the actual register bank.
#data	8-bit constant included in instruction.
#data16	16-bit constant included as bytes 2 and 3 of instruction.
bit	direct addressed bit in internal RAM or SFR.
addr16	16-bit destination address. Used by LCALL and LJMP. The branch will be anywhere within the 64 K-byte program memory address space.
addr11	11-bit destination address. Used by ACALL and AJMP. The branch will be within the same 2 K-byte page of program memory as the first byte of the following instruction.
rel	Signed (two's complement) 8-bit offset byte. Used by SJMP and all conditional jumps. Range is −128 to + 127 bytes relative to first byte of the following instruction.

Hexadecimal opcode cross-reference to Table 4

* : 8, 9, A, B, C, D, E, F.
● : 11, 31, 51, 71, 91, B1, D1, F1.
▲ : 01, 21, 41, 61, 81, A1, C1, E1.

DEVELOPMENT DATA

Table 5 Instruction map

first hexadecimal character of opcode ↓ — second hexadecimal character of opcode →

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	NOP	AJMP addr11	LJMP addr16	RR A	INC A	INC dir	INC @Ri		INC Rr							
							0	1	0	1	2	3	4	5	6	7
1	JBC bit,rel	ACALL addr11	LCALL addr16	RRC A	DEC A	DEC dir	DEC @Ri		DEC Rr							
							0	1	0	1	2	3	4	5	6	7
2	JB bit,rel	AJMP addr11	RET	RL A	ADD A,#data	ADD A,dir	ADD A,@Ri		ADD A,Rr							
							0	1	0	1	2	3	4	5	6	7
3	JNB bit,rel	ACALL addr11	RETI	RLC A	ADDC A,#data	ADDC A,dir	ADDC A,@Ri		ADDC A,Rr							
							0	1	0	1	2	3	4	5	6	7
4	JC rel	AJMP addr11	ORL dir,A	ORL dir,#data	ORL A,#data	ORL A,dir	ORL A,@Ri		ORL A,Rr							
							0	1	0	1	2	3	4	5	6	7
5	JNC rel	ACALL addr11	ANL dir,A	ANL dir,#data	ANL A,#data	ANL A,dir	ANL A,@Ri		ANL A,Rr							
							0	1	0	1	2	3	4	5	6	7
6	JZ rel	AJMP addr11	XRL dir,A	XRL dir,#data	XRL A,#data	XRL A,dir	XRL A,@Ri		XRL A,Rr							
							0	1	0	1	2	3	4	5	6	7
7	JNZ rel	ACALL addr11	ORL C,bit	JMP @A+DPTR	MOV A,#data	MOV dir,#data	MOV @Ri,#data		MOV Rr,#data							
							0	1	0	1	2	3	4	5	6	7
8	SJMP rel	AJMP addr11	ANL C,bit	MOVC A,@A+PC	DIV AB	MOV dir,dir	MOV dir,@Ri		MOV dir,Rr							
							0	1	0	1	2	3	4	5	6	7
9	MOV DPTR, #data 16	ACALL addr11	MOV bit,C	MOVC A,@A+DPTR	SUBB A,#data	SUBB A,dir	SUBB A,@Ri		SUBB A,Rr							
							0	1	0	1	2	3	4	5	6	7
A	ORL C,/bit	AJMP addr11	MOV C,bit	INC DPTR	MUL AB		MOV @Ri,dir		MOV Rr,dir							
							0	1	0	1	2	3	4	5	6	7
B	ANL C,/bit	ACALL addr11	CPL bit	CPL C	CJNE A, #data, rel	CJNE A,dir,rel	CJNE @Ri,#data, rel		CJNE Rr,#data, rel							
							0	1	0	1	2	3	4	5	6	7
C	PUSH dir	AJMP addr11	CLR bit	CLR C	SWAP A	XCH A,dir	XCH A,@Ri		XCH A,Rr							
							0	1	0	1	2	3	4	5	6	7
D	POP dir	ACALL addr11	SETB bit	SETB C	DA A	DJNZ dir,rel	XCHD A,@Ri		DJNZ Rr, rel							
							0	1	0	1	2	3	4	5	6	7
E	MOVX A,@DPTR	AJMP addr11	MOVX A,@Ri		CLR A	MOV A,dir *	MOV A,@Ri		MOV A,Rr							
			0	1			0	1	0	1	2	3	4	5	6	7
F	MOVX @DPTR,A	ACALL addr11	MOVX @Ri,A		CPL A	MOV dir,A	MOV @Ri,A		MOV Rr,A							
			0	1			0	1	0	1	2	3	4	5	6	7

* MOV A,ACC is not a valid instruction.

RATINGS

Limiting values in accordance with the Absolute Maximum System (IEC 134)

parameter	symbol	min.	max.	unit
Input voltage on any pin with respect to ground (V_{SS})	V_I	−0.5	+ 6.5	V
Input, output current	I_I, I_O	−	5.0	mA
Input, output current on any single pin	I_I, I_O	−	10.0	mA
Total power dissipation	P_{tot}	−	1.0	W
Storage temperature range	T_{stg}	−65	+ 150	°C
Operating ambient temperature range				
PCB83C552	T_{amb}	0	+ 70	°C
PCF83C552	T_{amb}	−40	+ 85	°C
PCA83C552	T_{amb}	−40	+ 125	°C

* Value to be fixed.

DEVELOPMENT DATA

DC CHARACTERISTICS

V_{DD} = 5 V (± 10%); V_{SS} = 0 V; T_{amb} = 0 to 70 °C (PCB83C552); −40 to + 85 °C (PCF83C552); −40 to + 125 °C (PCA83C552).

parameter	conditions	symbol	min.	max.	unit
Supply voltage	T_{amb} = 0 to + 70 °C	V_{DD}	4.0	6.0	V
	T_{amb} = −40 to + 80 °C	V_{DD}	4.0	6.0	V
	T_{amb} = −40 to + 125 °C	V_{DD}	4.5	5.5	V
Supply current operating	(note 1) f_{CLK} = 12 MHz; V_{DD} = 6.0 V	I_{DD}	–	34	mA
	f_{CLK} = 12 MHz; V_{DD} = 6.0 V	I_{DD}	–	30	mA
	(note 2) f_{CLK} = 12 MHz; V_{DD} = 5 V + 20%	I_{DD}	–	8	mA
	f_{CLK} = 12 MHz; V_{DD} = 5 V + 10%	I_{DD}	–	7	mA
Power-down current	2 V < V_{PD} < V_{DD} max.; (note 3)				
	T_{amb} = −40 to + 85 °C	I_{PD}	–	50	μA
	T_{amb} = −40 to + 125 °C	I_{PD}	–	100	μA
Inputs					
LOW level input voltage (except $\overline{EA}$, P1.6/SCL, P1.7/SDA)		V_{IL}	−0.5	$0.2V_{DD}$−0.1	V
LOW level input voltage ($\overline{EA}$)		V_{IL1}	−0.5	$0.2V_{DD}$−0.3	V
HIGH level input voltage (except RST, XTAL1, P1.6/SCL, P1.7/SDA)		V_{IH}	$0.2V_{DD}$+0.9	V_{DD}+0.5	V
LOW level input voltage (P1.6/SCL, P1.7/SDA)		V_{IL2}	−0.5	$0.3V_{DD}$	V
HIGH level input voltage (RST and XTAL1)		V_{IH1}	$0.7V_{DD}$	V_{DD}+0.5	V
HIGH level input voltage (P1.6/SCL, P1.7/SDA)		V_{IH2}	$0.7V_{DD}$	6.0	V
Input current logic 0 (Ports 1, 2, 3 and 4; except P1.6/SDA, P1.7/SCL)	V_I = 0.45 V	$-I_{IL}$	–	50	μA
Input current logic 1 to 0 transition (Ports 1, 2, 3 and 4; except P1.6/SCL, P1.7/SDA)	V_I = 2.0 V	$-I_{TL}$	–	650	μA

DC CHARACTERISTICS (continued)

parameter	conditions	symbol	min.	max.	unit
Input leakage current (Port 0, Port 5, $\overline{EA}$, STADC, $\overline{EW}$)	0,45 V < V_I < V_{DD}	$\pm I_{LI1}$	–	10	μA
Input leakage current (P1.6/SCL, P1.7/SDA)	0 V < V_I < 6.0 V 0 V < V_I < 6.0 V	$\pm I_{LI2}$	–	10	μA
Outputs					
LOW level output voltage (Ports 1, 2, 3 and 4; except P1.6/SCL, P1.7/SDA)	(note 10) I_{OL} = 1.6 mA	V_{OL}	–	0.45	V
LOW level output voltage (Port 0, ALE, $\overline{PSEN}$, $\overline{PWM0}$, $\overline{PWM1}$)	(note 10) I_{OL} = 3.2 mA	V_{OL1}	–	0.45	V
LOW level output voltage (P1.6/SCL, P1.7/SDA)	I_{OL} = 3.0 mA	V_{OL2}	–	0.40	V
HIGH level output voltage (Ports 1, 2, 3 and 4; except P1.6/SCL, P1.7/SDA)	V_{DD} = 5 V + 10% $-I_{OH}$ = 60 μA $-I_{OH}$ = 25 μA $-I_{OH}$ = 10 μA	V_{OH} V_{OH} V_{OH}	2.4 0.75V_{DD} 0.9V_{DD}	– – –	V V V
HIGH level output voltage (Port 0 in external bus mode, ALE, $\overline{PSEN}$, $\overline{PWM0}$, $\overline{PWM1}$)	(note 11) V_{DD} = 5 V + 10% $-I_{OH}$ = 400 μA $-I_{OH}$ = 150 μA $-I_{OH}$ = 40 μA	V_{OH1} V_{OH1} V_{OH1}	2.4 0.75V_{DD} 0.9V_{DD}	– – –	V V V
HIGH level output voltage (RST)	$-I_{OH}$ = 400 μA $-I_{OH}$ = 120 μA	V_{OH2} V_{OH2}	2.4 0.8V_{DD}	– –	V V
RST pull-down resistor		R_{RST}	50	150	kΩ
Capacitance of I/O buffer	Test freq. = 1 MHz T_{amb} = 25 °C	$C_{I/O}$	–	10	pF

DEVELOPMENT DATA

DC CHARACTERISTICS

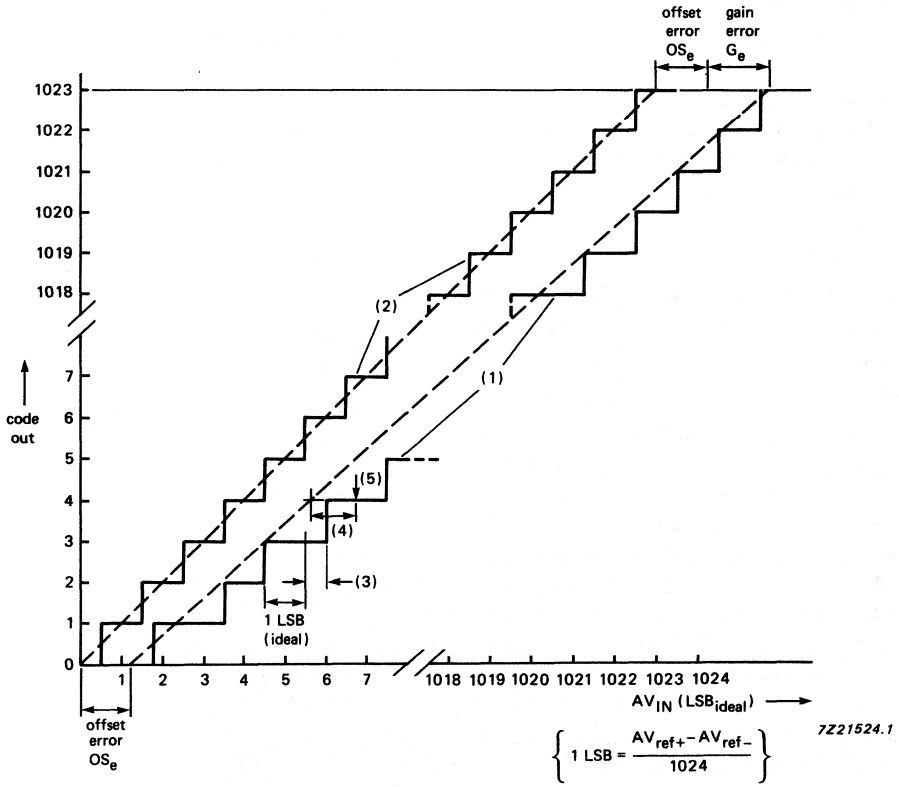
parameter	conditions	symbol	min.	max.	unit
Analogue inputs	(note 10)				
Analogue supply voltage	$AV_{DD} = V_{DD} \pm 0.2$ V	AV_{DD}	4.0	6.0	V
Analogue supply current operating	Port 5 = 0 to AV_{DD}	AI_{DD}	–	1.2	mA
Idle mode	T_{amb} = –40 to + 85 °C	AI_{ID}	–	50	μA
	T_{amb} = –40 to + 125 °C	AI_{ID}	–	100	μA
Power-down	2 V < AV_{PD} < AV_{DD} max				
	T_{amb} = –40 to + 85 °C	AI_{PD}	–	50	μA
	T_{amb} = –40 to + 125 °C	AI_{PD}	–	100	μA
Analogue input voltage		AV_{IN}	AV_{SS}–0.2	AV_{DD}+0.2	V
Reference voltage		AV_{REF-}	AV_{SS}–0.2	–	V
		AV_{REF+}	–	AV_{DD}+0.2	V
Resistance between AV_{REF+} and AV_{REF-}		R_{REF}	10	50	kΩ
Analogue input capacitance		C_{IA}	–	15	pF
Sampling time		t_{ADS}	–	$8t_{CY}$	μs
Conversion time (including sample time)		t_{ADC}	–	$50t_{CY}$	μs
Differential non-linearity	(note 4 and 5)	DL_e	–	$-1 < DL_e < 2$	LSB
Integral non-linearity	(note 4 and 6)	IL_e	–	± 2	LSB
Offset error	(note 9)	OS_e	–	± 2	mV
Gain error	(note 7)	G_e	–	± 0.4	%
Channel to channel matching		M_{ctc}	–	± 1	LSB
Crosstalk	0 to 100 kHz (note 8)	C_t	–	–60	dB

Notes to the DC characteristics

1. The operating supply current is measured with all output pins disconnected; XTAL1 driven with $t_r = t_f = 10$ ns; $V_{IL} = V_{SS} + 0.5$ V; $V_{IH} = V_{DD} - 0.5$ V; XTAL2 not connected; $\overline{EA}$ = RST = Port 0 = $\overline{EW}$ = V_{DD}; STADC = V_{SS}.
2. The Idle mode supply current is measured with all output pins disconnected; XTAL1 driven with $t_r = t_f = 10$ ns; $V_{IL} = V_{SS} + 0.5$ V; $V_{IH} = V_{DD} - 0.5$ V; XTAL2 not connected; $\overline{EA}$ = Port 0 = $\overline{EW}$ = V_{DD}; RST = STADC = V_{SS}.
3. The power-down current is measured with all output pins disconnected; XTAL2 not connected; $\overline{EA}$ = Port 0 = $\overline{EW}$ = V_{DD}; RST = STADC = V_{SS}.
4. $AV_{REF+} = 5.12$ V; $AV_{REF-} = 0$ V; $AV_{DD} = 5.0$ V.
5. DLe: The differential non-linearity is the difference between the actual step width and the ideal step width.
6. ILe: The integral non-linearity is the peak difference between the actual and the ideal quantization levels after appropriate adjustment of gain and offset error.
7. The gain error is the relative difference in per cent between the straight line fitting the actual transfer curve (after removing gain error), and the straight line which fits the ideal transfer curve. Gain error is constant at every point on the transfer curve.
8. This should be considered when both analogue and digital signals are input simultaneously.

Notes to the DC characteristics (continued)

9. OSe: The offset error is the absolute difference between the straight line which fits the actual transfer curve (after removing gain error), and a straight line which fits the ideal transfer curve. For an uncalibrated A/D converter, the offset error is constant at every point of the actual transfer curve.
10. Capacitive loading on Port 0 and Port 2 may cause spurious noise pulses to be superimposed on the LOW level output voltage of ALE, Port 1 and Port 3. This noise is due to external bus capacitance discharging into the Port 0 and Port 2 pins when these pins make a 1-to-0 transition during bus operations. In the most adverse condition (capacitive loading > 100 pF) the noise pulse on ALE may exceed 0.8 V. In this event it may be necessary to qualify ALE with a Schmitt trigger, or use an address latch with a Schmitt trigger STROBE input.
11. When the address bits are stabilizing, capacitive loading on Port 0 and Port 2 may cause the HIGH level output voltage on ALE and $\overline{PSEN}$ to momentarily fall below the 0.9 V_{DD} specification.
12. The ADC is monotonic; there are no missing codes.



(1) Actual step curve.
(2) Ideal step curve.
(3) Differential non-linearity.
(4) Integral non-linearity.
(5) Centre of step.

Fig.18 ADC conversion characteristic.

After alignment of the offset and gain errors the integral non-linearity is equal to the absolute error.

DEVELOPMENT DATA

AC CHARACTERISTICS

V_{DD} = 5 V ± 20% (PCB83C552), V_{SS} = 0 V, $t_{CK\ min} = 1/f_{max}$ (maximum operating frequency)
V_{DD} = 5 V ± 20% (PCF83C552), V_{SS} = 0 V, $t_{CK\ min} = 1/f_{max}$ (maximum operating frequency)
V_{DD} = 5 V ± 10% (PCA83C552), V_{SS} = 0 V, $t_{CK\ min} = 1/f_{max}$ (maximum operating frequency)
T_{amb} = 0 to + 70 °C, $t_{CK\ min}$ = 83 ns (PCB83C552, see note 1)
T_{amb} = −40 to + 85 °C, $t_{CK\ min}$ = 83 ns (PCF83C552, see note 1)
T_{amb} = −40 to + 125 °C, $t_{CK\ min}$ = 83 ns (PCA83C552, see note 1)
C_L = 100 pF for Port 0, ALE and $\overline{PSEN}$; C_L = 80 pF for all other outputs unless otherwise specified.
(See waveforms Figs 21, 22 and 23).

parameter	symbol	10 MHz		12 MHz		variable clock		unit
		min.	max.	min.	max.	min.	max.	
Program memory								
ALE pulse duration	t_{LL}	160	–	127	–	$2t_{CK}-40$	–	ns
Address set-up time to ALE	t_{AL}	45	–	28	–	$t_{CK}-55$	–	ns
Address hold time after ALE	t_{LA}	65	–	48	–	$t_{CK}-35$	–	ns
Time from ALE to valid instruction input	t_{LIV}	–	300	–	233	–	$4t_{CK}-100$	ns
Time from ALE to control pulse $\overline{PSEN}$	t_{LC}	60	–	43	–	$t_{CK}-40$	–	ns
Control pulse duration $\overline{PSEN}$	t_{CC}	255	–	205	–	$3t_{CK}-45$	–	ns
Time from $\overline{PSEN}$ to valid instruction input	t_{CIV}	–	195	–	145	–	$3t_{CK}-105$	ns
Input instruction hold time after $\overline{PSEN}$	t_{CI}	0	–	0	–	0	–	ns
Input instruction float delay after $\overline{PSEN}$	t_{CIF}	–	75	–	59	–	$t_{CK}-25$	ns
Address to valid instruction input	t_{AIV}	–	395	–	312	–	$5t_{CK}-105$	ns
Address float delay after $\overline{PSEN}$	t_{AFC}	–	10	–	10	–	10	ns

AC CHARACTERISTICS (continued)

parameter	symbol	10 MHz		12 MHz		variable clock		unit
		min.	max.	min.	max.	min.	max.	
External data memory								
RD pulse duration	t_{RR}	500	–	400	–	$6t_{CK}-100$	–	ns
WR pulse duration	t_{WW}	500	–	400	–	$6t_{CK}-100$	–	ns
Address set up time to ALE	t_{AL}	65	–	48	–	$t_{CK}-40$	–	ns
Address hold time after ALE	t_{LA}	–	–	48	–	$t_{CK}-35$	–	ns
RD to valid data input	t_{RD}	–	355	–	252	–	$5t_{CK}-165$	ns
Data hold time after $\overline{RD}$	t_{DR}	0	–	0	–	0	–	ns
Data float delay after $\overline{RD}$	t_{DFR}	–	130	–	97	–	$2t_{CK}-70$	ns
Time from ALE to valid data input	t_{LD}	–	650	–	517	–	$8t_{CK}-150$	ns
Address to valid data input	t_{AD}	–	735	–	585	–	$9t_{CK}-165$	ns
Time from ALE to $\overline{RD}$ or $\overline{WR}$	t_{LW}	250	350	200	300	$3t_{CK}-50$	$3t_{CK}+50$	ns
Time from address to $\overline{RD}$ or $\overline{WR}$	t_{AW}	270	–	203	–	$4t_{CK}-130$	–	ns
Time from $\overline{RD}$ or $\overline{WR}$ HIGH to ALE HIGH	t_{WHLH}	60	140	43	123	$t_{CK}-40$	$t_{CK}+40$	ns
Data valid to $\overline{WR}$ transition	t_{DWX}	40	–	23	–	$t_{CK}-60$	–	ns
Data set-up time before $\overline{WR}$	t_{DW}	550	–	433	–	$7t_{CK}-150$	–	ns
Data hold time after $\overline{WR}$	t_{WD}	50	–	33	–	$t_{CK}-50$	–	ns
Address float delay after $\overline{RD}$	t_{AFR}	–	0	–	0	–	0	ns

Note to AC characteristics

The maximum and minimum operating frequency for all versions is 12 MHz and 1.2 MHz respectively.

Where:

$1/t_{CK}$ = 1.2 to 12 MHz (see Fig.20 and Table 6).

$t_{CY} = 12\ t_{CK}$ (see Fig.21 and DC Characteristics).

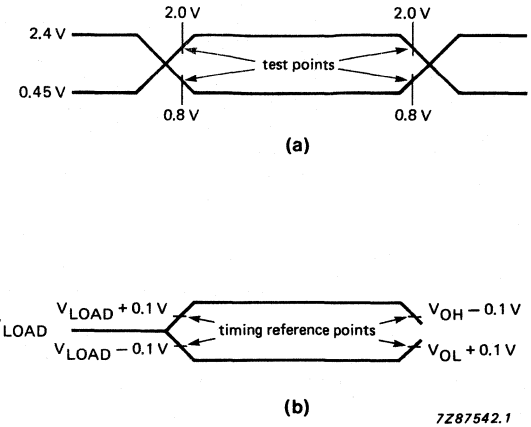


Fig. 19 AC testing input, output waveform (a) and float waveform (b).

AC testing inputs are driven at 2,4 V for a logic 1 and 0,45 V for a logic 0. Timing measurements are taken at 2,0 V for a logic 1 and 0,8 V for logic 0. The float state is defined as the point at which a Port 0 pin sinks 3,2 mA or sources 400 μA at the voltage test levels.

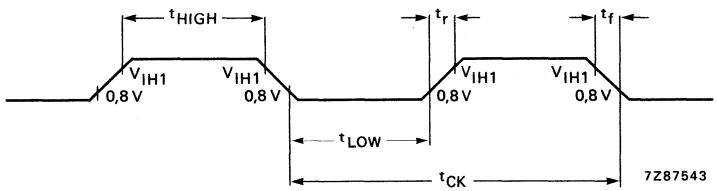


Fig. 20 External clock drive XTAL 1 (see Table 6).

Table 6 External clock drive XTAL 1 (see Fig. 20).

parameter	symbol	variable clock (f = 1,2 to 12 MHz)		unit
		min.	max.	
oscillator clock period	t_{CK}	83,3	833.3	ns
HIGH time	t_{HIGH}	20	t_{CK}-t_{LOW}	ns
LOW time	t_{LOW}	20	t_{CK}-t_{HIGH}	ns
rise time	t_r	–	20	ns
fall time	t_f	–	20	ns
cycle time	t_{CY}1	1	10	μs

1. $t_{CY} = 12t_{CK}$.

AC CHARACTERISTICS (continued)

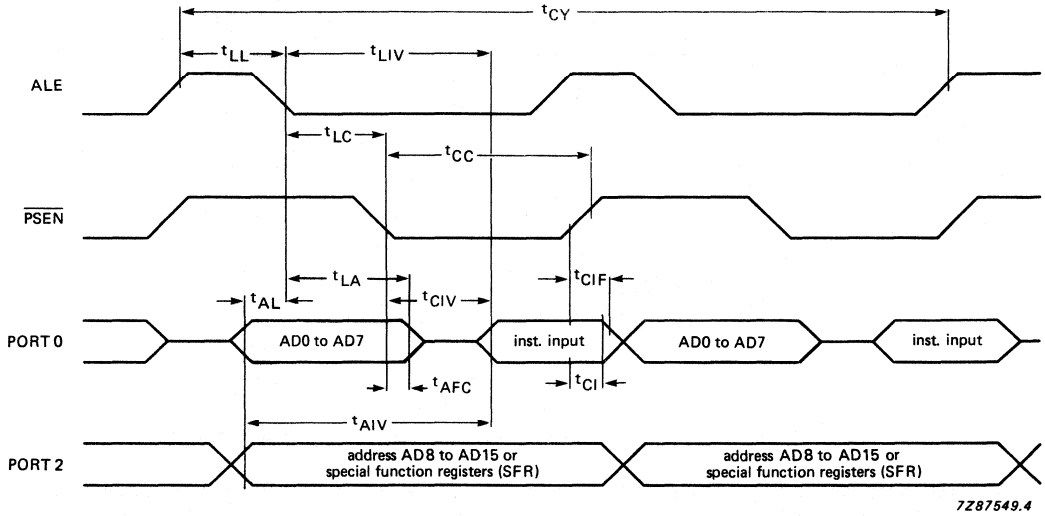


Fig. 21 Read from program memory.

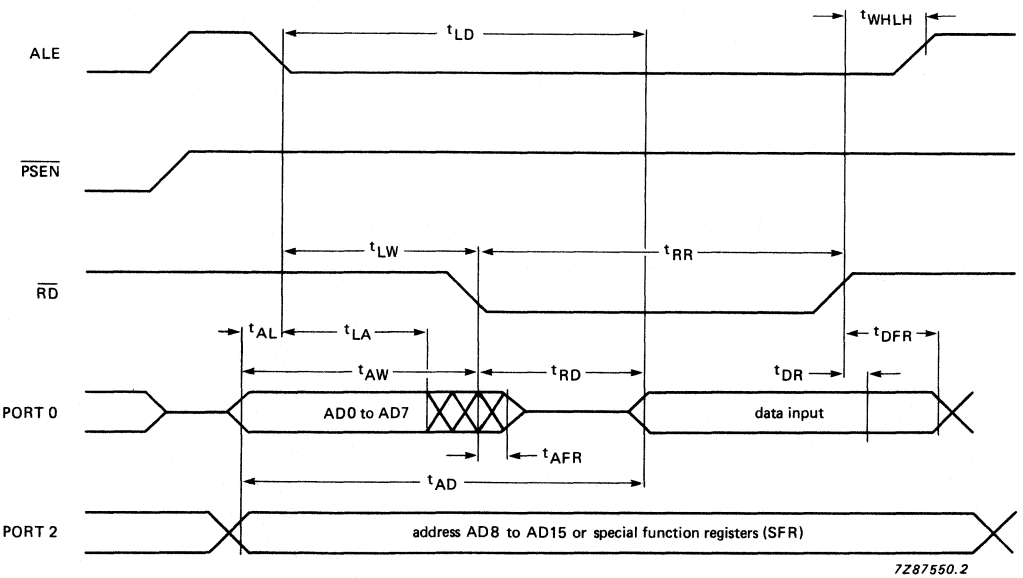


Fig. 22 Read from data memory.

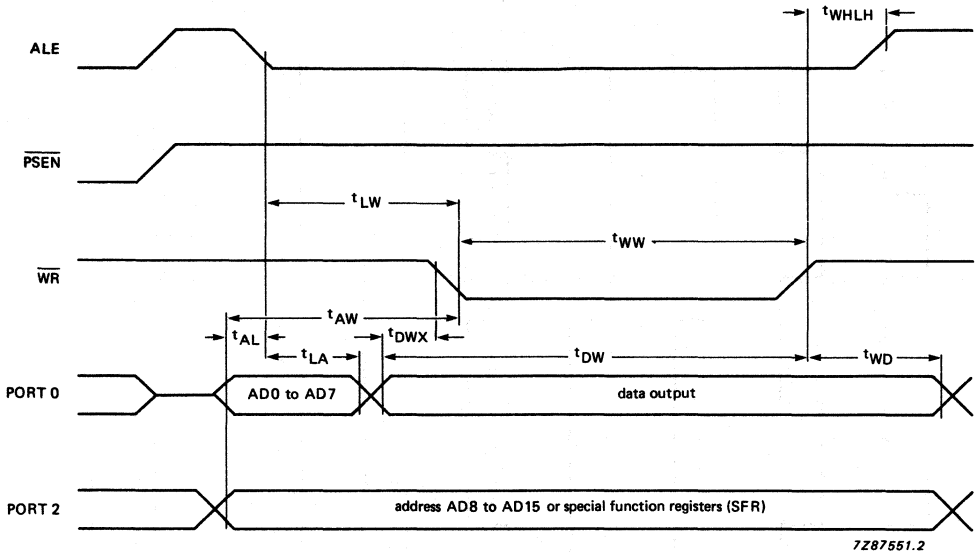


Fig. 23 Write to data memory.

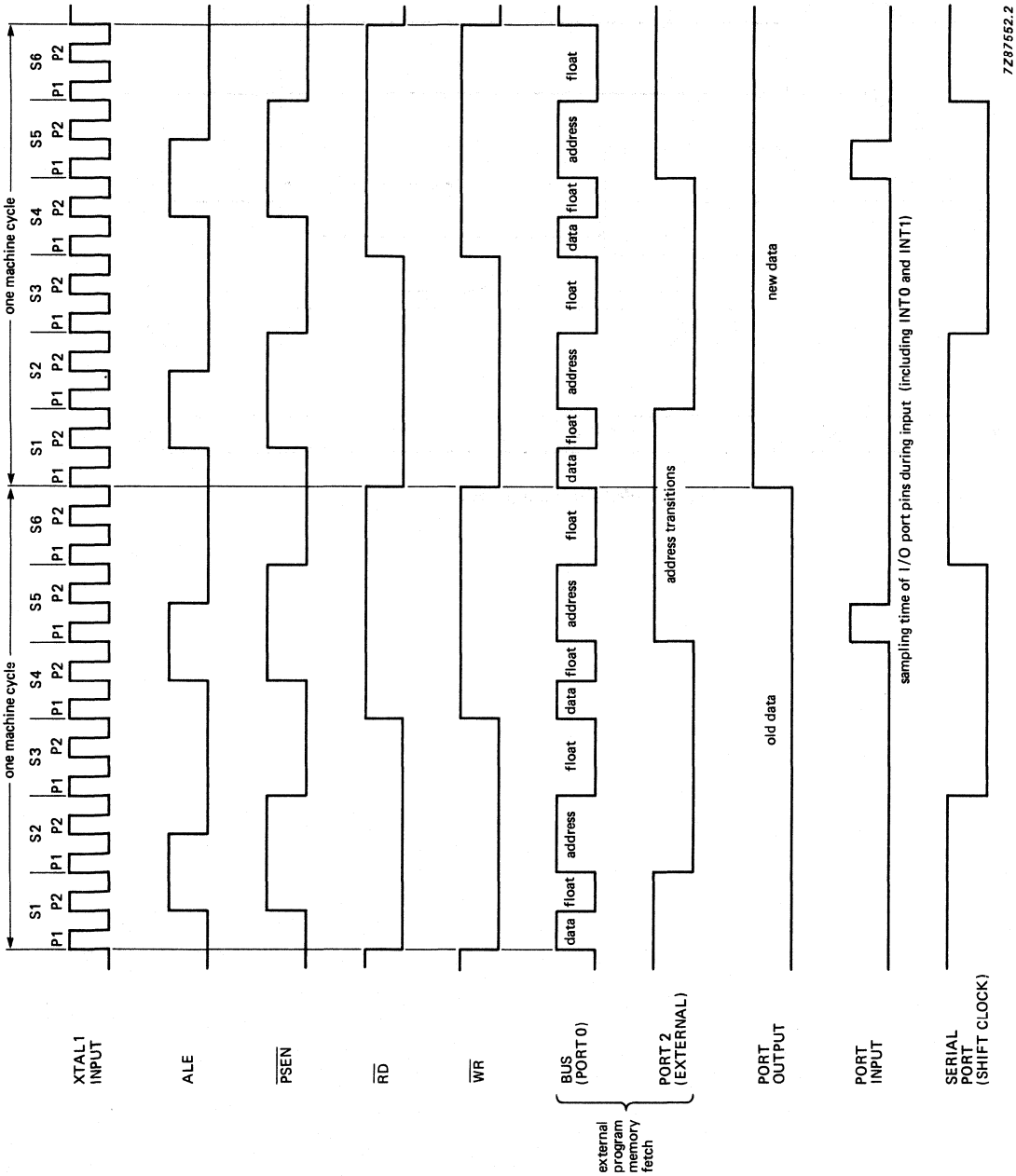


Fig. 24 Instruction cycle timing.

SINGLE-CHIP 8-BIT MICROCONTROLLER

GENERAL DESCRIPTION

The PCB83C562 single-chip 8-bit microcontroller is manufactured in an advanced CMOS process and is a derivative of the PCB80C51 microcontroller family. PCB83C562 has the same instruction set as the PCB80C51. Two versions of the derivative exist although the generic term "PCB83C562" is used to refer to both family members:

- PCB80C562: ROM-less version of the PCB83C562
- PCB83C562: 8 K bytes mask-programmable ROM, 256 bytes RAM

The PCB83C562 contains a non-volatile 8 K x 8 read-only program memory (not ROM-less version), a volatile 256 x 8 read/write data memory; six 8-bit I/O ports; two 16-bit timer/event counters (identical to the timers of the PCB80C51); an additional 16-bit timer coupled to capture and compare latches; a fourteen source, two-priority-level, nested interrupt structure; an ADC and DAC pulse width modulated interface, a UART serial interface, a 'watchdog' timer and on-chip oscillator and timing circuits. For systems that require extra capability, the PCB83C562 can be expanded using standard TTL compatible memories and logic.

The device also functions as an arithmetic processor having facilities for both binary and BCD arithmetic plus bit-handling capabilities. The instruction set consists of over 100 instructions; 44% one-byte, 41% two-byte and 15% three-byte. With a 12 MHz crystal, 58% of the instructions are executed in 1 μs and 40% in 2 μs. Multiply and divide instructions require 4 μs.

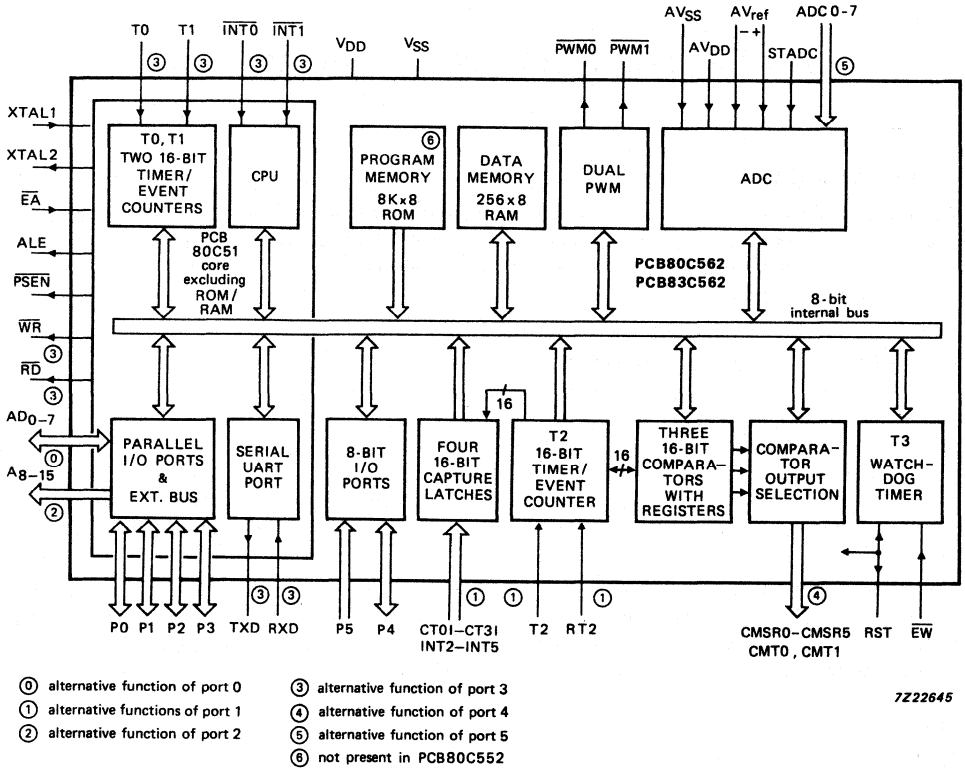
Features

- PCB80C51 central processing unit
- 8 K x 8 ROM, expandable externally to 64 K x 8 bytes
- 256 x 8 RAM, expandable externally to 64 K x 8 bytes
- Two standard 16-bit timer/counters
- An additional 16-bit timer/counter coupled to four capture registers and three compare registers
- An ADC with 8 multiplexed analogue inputs and 8-bit resolution
- Two 8-bit resolution, Pulse Width Modulated analogue outputs
- Five 8-bit I/O ports plus one 8-bit input port shared with analogue inputs
- Full-duplex UART compatible with the standard PCB80C51
- On-chip watchdog timer
- Operating ambient temperature range:

0 to + 70 °C	(PCB83C562)
−40 to + 85 °C	(PCF83C562)
−40 to + 125 °C	(PCA83C562)

PACKAGE OUTLINE

PCB/PCF/PCA83C562: 68-lead PLCC; plastic 'pocket' version (SOT188AA).



⓪ alternative function of port 0
① alternative functions of port 1
② alternative function of port 2
③ alternative function of port 3
④ alternative function of port 4
⑤ alternative function of port 5
⑥ not present in PCB80C552

7Z22645

POWER SUPPLY
V_{DD} + 5 V MAIN SUPPLY
V_{SS} GROUND

Fig. 1 Block diagram.

DEVELOPMENT DATA

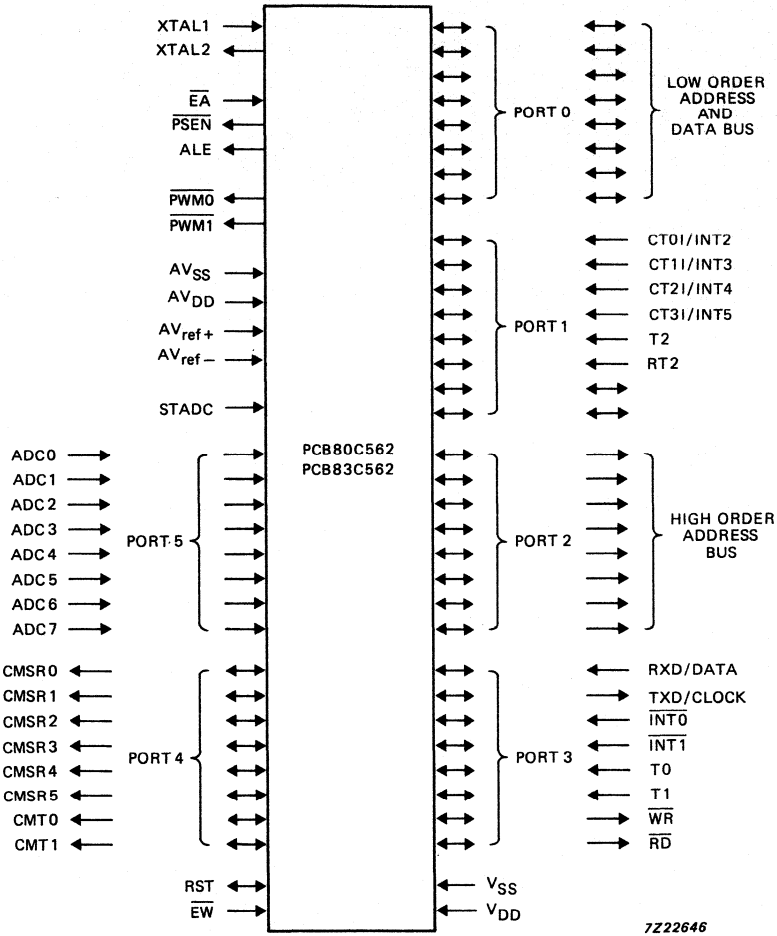


Fig. 2 Functional diagram.

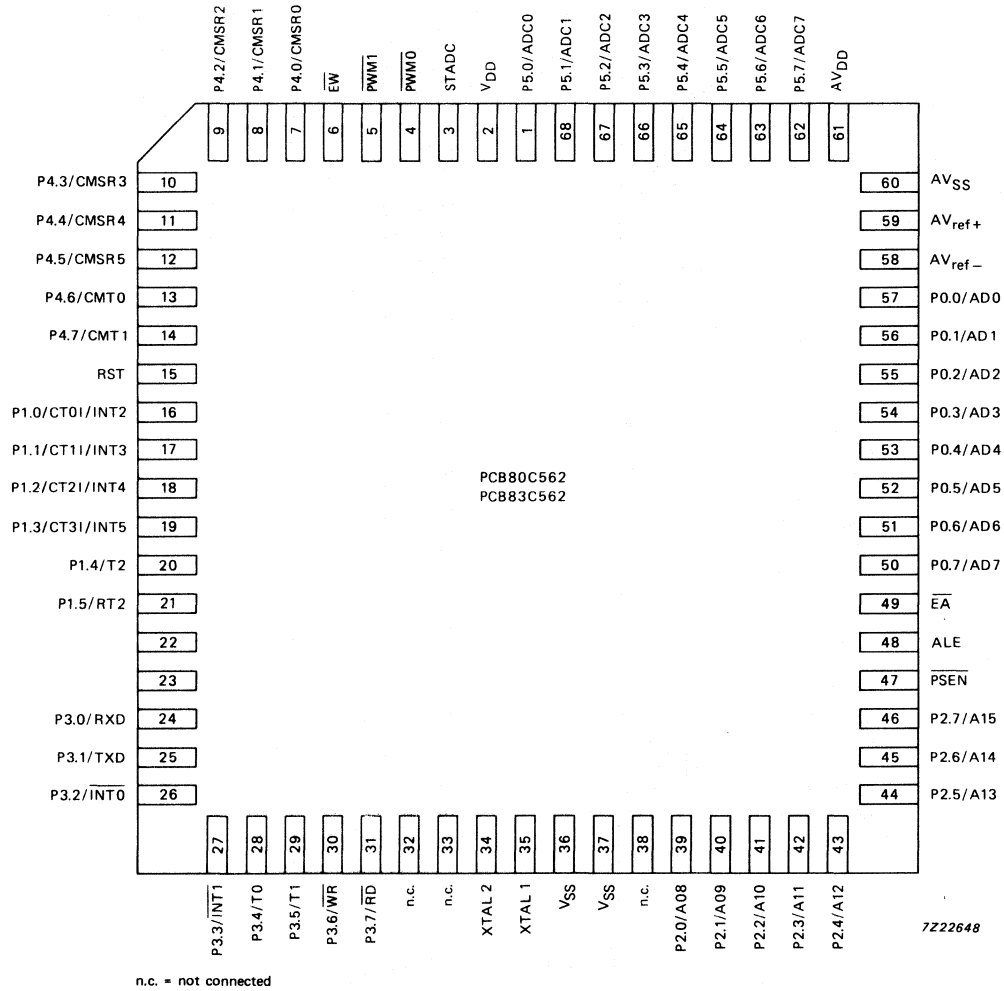


Fig. 3 Pinning diagram for PCB83C562.

DEVELOPMENT DATA

PINNING

2	V_{DD}	**Digital power supply**: + 5 V power supply pin during normal operation, Idle mode and Power-down mode
3	STADC	**Start ADC operation**: Input starting analogue to digital conversion (ADC operation can also be started by software)
4	$\overline{PWM0}$	**Pulse width** modulation output 0
5	$\overline{PWM1}$	**Pulse width** modulation output 1
6	$\overline{EW}$	**Enable watchdog timer**: Enable for T3 watchdog timer and disable Power-down mode
7-14	P4.0-P4.7	**Port 4** 8-bit quasi-bidirectional I/O port

Port pin	Alternative function	
P4.0	CMSR0	Timer T2: compare and set/reset outputs on a match with timer T2
P4.1	CMSR1	
P4.2	CMSR2	
P4.3	CMSR3	
P4.4	CMSR4	
P4.5	CMSR5	
P4.6	CMT0	Timer T2: compare and toggle outputs on a match with timer T2
P4.7	CMT1	

15	RST	**Reset**: Input to reset the PCB83C562. It also provides a reset pulse as output when timer T3 overflows.
16-23	P1.0-P1.7	**Port 1** 8-bit quasi-bidirectional I/O port

Port pin	Alternative function	
P1.0	CT0I/INT2	Capture timer input signals for timer T2 or Interrupts INT2 to INT5
P1.1	CT1I/INT3	
P1.2	CT2I/INT4	
P1.3	CT3I/INT5	
P1.4	T2	: T2 event input
P1.5	RT2	: T2 timer reset signal. Rising edge triggered
P1.6		
P1.7		

24-31	P3.0-P3.7	**Port 3** 8-bit quasi-bidirectional I/O port

Port pin	Alternative function	
P3.0	RXD	: Serial input port
P3.1	TXD	: Serial output port
P3.2	$\overline{INT0}$	: External interrupt
P3.3	$\overline{INT1}$	: External interrupt
P3.4	T0	: Timer 0 external input
P3.5	T1	: Timer 1 external input
P3.6	$\overline{WR}$	: External data memory write strobe
P3.7	$\overline{RD}$	: External data memory read strobe

32, 33 Not connected

PINNING (continued)

34	XTAL2	**Crystal input 2:** output of the inverting amplifier that forms the oscillator. Left open-circuit when an external oscillator clock is used.
35	XTAL1	**Crystal input 1:** input to the inverting amplifier that forms the oscillator, and input to the internal clock generator. Receives the external oscillator clock signal when an external oscillator is used.
36, 37	V_{SS}	**Two digital** ground pins
38		Not connected
		Port 2
39-46	P2.0-P2.7	8-bit quasi-bidirectional I/O port Port pin — Alternative function P2.0-P2.7 — High-order address byte for external memory (A08-A15)
47	$\overline{PSEN}$	**Program store enable:** active LOW read strobe to external program memory
48	ALE	**Address latch enable:** latches the low byte of the address during accesses to external memory. It is activated every six oscillator periods. During an external data memory access one ALE pulse is skipped. ALE can drive up to 8 LSTTL inputs and handles CMOS inputs without an external pull-up.
49	$\overline{EA}$	**External access:** When $\overline{EA}$ is held at TTL level HIGH, the CPU executes out of the internal program ROM provided the program counter is less than 8192. When $\overline{EA}$ is held at TTL LOW level, the CPU executes out of external program memory. $\overline{EA}$ is not allowed to float.
		Port 0
50-57	P0.7-P0.0	8-bit binary I/O port Port pin — Alternative function P0.7-P0.0 — Multiplexed low-order address and data bus of external memory (AD7-AD0)
58	$AV_{ref}-$	**Low end** of analogue to digital conversion reference resistor
59	$AV_{ref}+$	**High end** of analogue to digital conversion reference resistor
60	AV_{SS}	**Analogue** ground
61	AV_{DD}	**Analogue** power supply
		Port 5
62-68,1	P5.7-P5.0	8-bit input port Port pin — Alternative function P5.0-P5.7 — Eight input channels to ADC (ADC0-ADC7)

To avoid a 'latch-up' effect at power-on, the voltage on any pin at any time must not be higher or lower than V_{DD} + 0,5 V or V_{SS} – 0,5 V respectively.

FUNCTIONAL DESCRIPTION

General

The PCB83C562 is a stand-alone high-performance microcontroller designed for use in real-time applications such as instrumentation, industrial control and specific automotive control applications.

The device provides in addition to the 80C51 standard functions, a number of dedicated hardware functions for these applications.

The PCB83C562 is a control-oriented CPU with on-chip program and data memory. It can be extended with external program memory up to 64 K bytes. It can also access up to 64 K bytes of external data memory. For systems requiring extra capability, the PCB83C562 can be expanded using standard memories and peripherals.

The PCB83C562 has two software selectable modes of reduced activity for further power reduction – Idle and Power-down. The Idle mode freezes the CPU while allowing the RAM, timers, serial ports and interrupt system to continue functioning. The Power-down mode saves the RAM contents but freezes the oscillator causing all other chip functions to be inoperative.

Memory organisation

The central processing unit (CPU) manupilates operands in three memory spaces; these are the 64 K-byte external data memory, 256-byte internal data memory and the 64 K-byte internal and external program memory. The internal data memory address space is sub-divided into the 256-byte internal data RAM and 128-byte Special Function Register (SFR) address spaces, as shown in Fig. 4. Figures 5(a) and (b) show the Special Function Register memory map. Internal RAM locations 0-127 are directly and indirectly addressable. Internal RAM locations 128-255 are only indirectly addressable. The special function register locations 128-255 are only directly addressable.

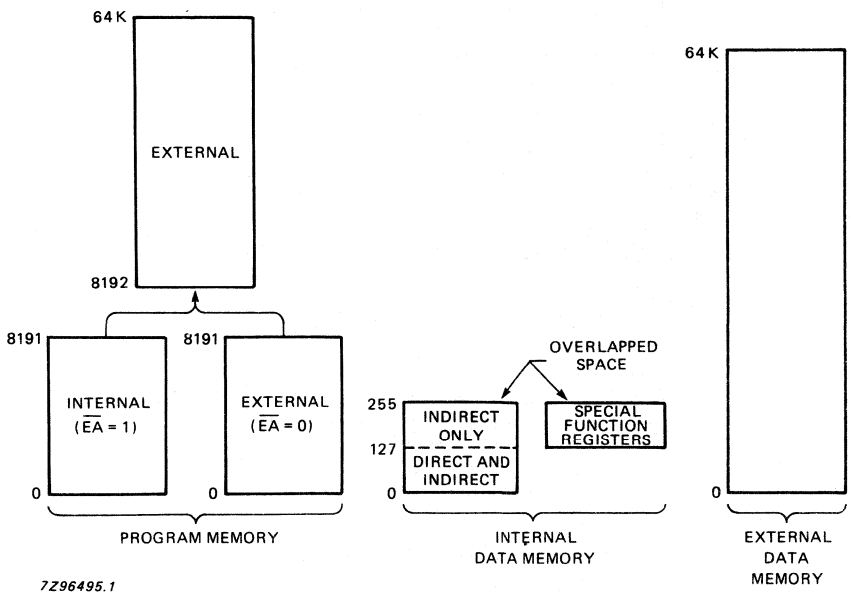
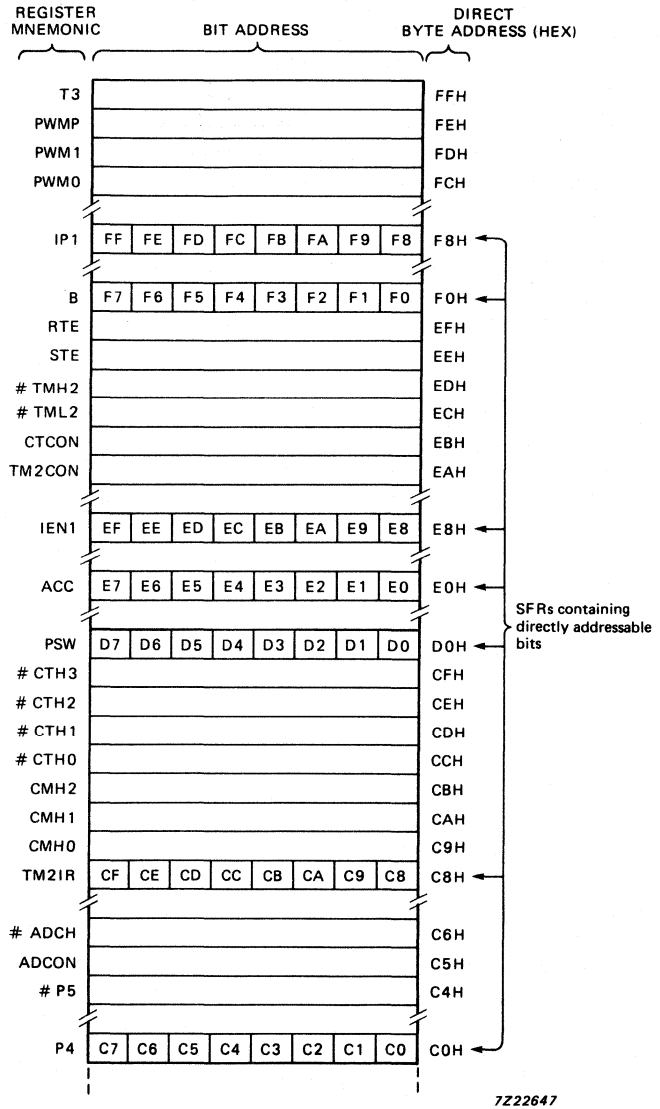


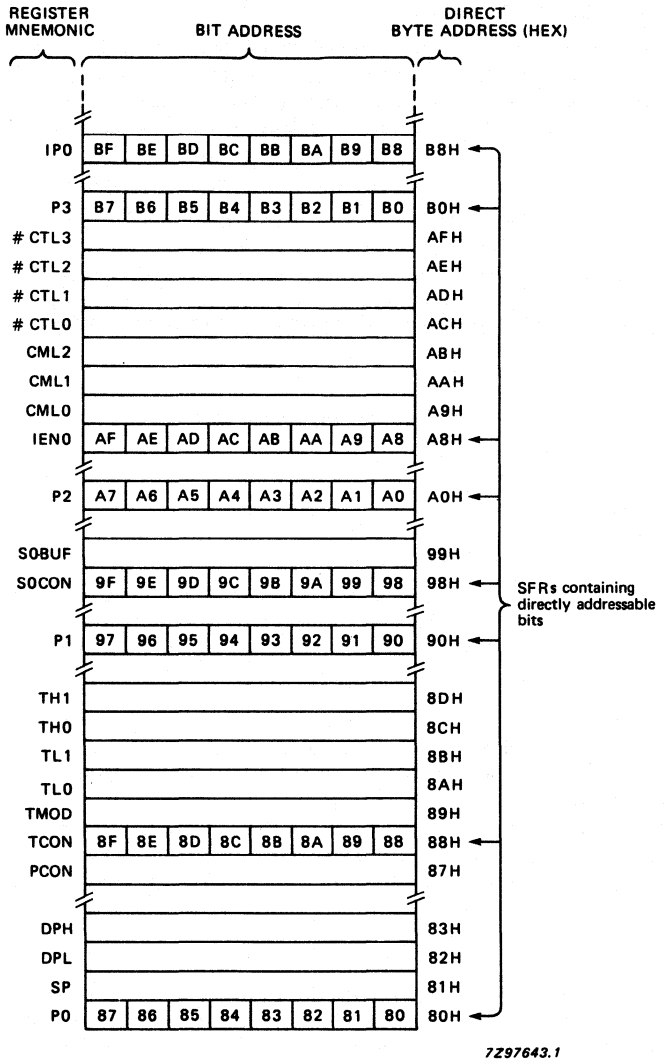
Fig. 4 Memory map.

FUNCTIONAL DESCRIPTION (continued)



# denotes read only registers

Fig. 5(a) Special Function Register memory map (continued in Fig. 5(b)).



# denotes read only registers

Fig. 5(b) Special Function Register memory map (continued from Fig. 5(a)).

FUNCTIONAL DESCRIPTION (continued)

The internal data RAM contains four register banks (each with eight registers), 128 addressable bits, a scratchpad area and the stack. The stack depth is limited by the available internal data RAM and its location is determined by the 8-bit stack pointer. All registers except the program counter and the four 8-register banks reside in the special function register address space. These memory mapped registers include arithmetic registers, pointers, I/O ports, interrupt system registers, ADC and PWM registers, timers and serial port registers. There are 128 addressable bit locations in the SFR address space.

The PCB83C562 contains 256 bytes of internal data RAM and 56 special function registers. It provides a non-paged program memory address space to accommodate relocatable code. Conditional branches are performed relative to the program counter. The register-indirect jump permits branching relative to a 16-bit base register with an offset provided by an 8-bit index register. 16-bit jumps and calls permit branching to any location in the contiguous 64 K program memory address space.

Addressing

The PCB83C562 has five methods for addressing source operands:

- Register
- Direct
- Register-Indirect
- Immediate
- Base-Register plus Index-Register-Indirect

The first three methods can be used for addressing destination operands. Most instructions have a "destination/source" field that specifies the data type, addressing methods and operands involved. For operations other than MOVs, the destination operand is also a source operand.

Access to memory addressing is as follows:

- Registers in one of the four 8-register banks through Register, Direct, or Register-Indirect
- 256 bytes of internal data RAM through Direct or Register-Indirect. Bytes 0-127 may be addressed directly/indirectly. Bytes 128-255 share their address locations with the SFR registers and so may only be addressed indirectly as data RAM
- Special function registers through Direct at address locations 128-255
- External data memory through Register-Indirect
- Program memory look-up tables through Base-Register plus Index-Register-Indirect

Instruction set

The PCB83C562 uses the powerful instruction set of the PCB80C51. Additional special function registers are incorporated to control the on-chip peripherals. The instruction set consists of 49 single-byte, 45 two-byte and 17 three-byte instructions. When using a 12 MHz oscillator, 64 instructions execute in 1 μs and 45 instructions execute in 2 μs. Multiply and divide instructions execute in 4 μs.

I/O facilities

The PCB83C562 has six 8-bit ports. Ports 0 - 3 are the same as in the PCB80C51, with the exception of the additional functions of Port 1. The parallel I/O function of Port 4 is equal to that of Ports 1, 2 and 3. Port 5 has a parallel input port function, but has no function as an output port. Port lines P1.7 and P1.6 are provided with push-pull drivers.

Ports 0, 1, 2, 3, 4 and 5 perform the following alternative functions:

- Port 0: provides the multiplexed low-order address and data bus used for expanding the PCB83C562 with standard memories and peripherals.
- Port 1: Port 1 is used for a number of special functions;
  - 4 capture inputs
  - external counter input
  - external counter reset input

Bits whose alternate function is not used may be used as normal bidirectional I/O pins.

- Port 2: provides the high-order address bus when expanding the PCB83C562 with external program memory and/or external data memory.
- Port 3: pins can be configured individually to provide:
  - external interrupt request inputs
  - counter inputs
  - serial port receiver input and transmitter output
  - control signals to READ and WRITE external data memory

The generation or use of a Port 3 pin as an alternative function is carried out automatically by the PCB83C562 provided the associated Special Function Register bit is set high.

- Port 4: can be configured to provide signals indicating a match between timer counter T2 and its compare registers.
- Port 5: may be used in conjunction with the ADC interface. Unused analogue inputs can be used as digital inputs. As Port 5 lines may be used as inputs to the ADC, these digital inputs have an inherent hysteresis to prevent the input logic from drawing too much current from the power lines when driven by analogue signals. Channel to channel crosstalk should be taken into consideration when both digital and analogue signals are simultaneously input to Port 5 (see DC characteristics).

All ports are bidirectional with the exception of Port 5 which is an input port.

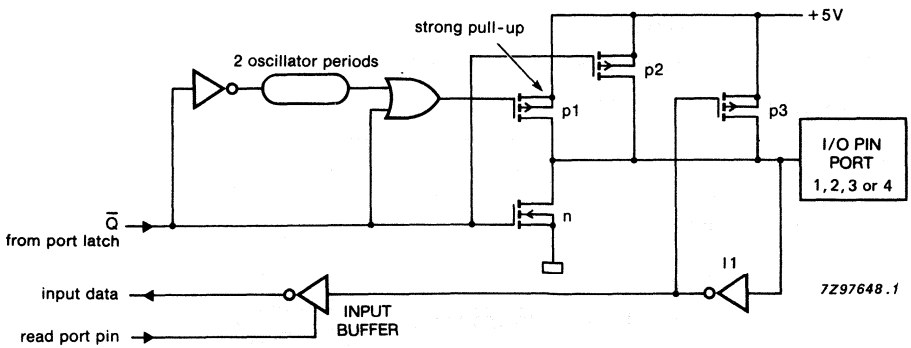


Fig. 6 I/O buffers in the PCB83C562 (Ports 2, 3, 4 and P1.0 to P1.5).

FUNCTIONAL DESCRIPTION (continued)

In addition to the standard 8-bit ports, the I/O facilities of the PCB83C562 also include a number of special I/O lines:

Pulse width modulated outputs

Two pulse width modulated output channels are provided with the PCB83C552. These channels output pulses of programmable length and interval. The repetition frequency is defined by an 8-bit prescaler PWMP which generates the clock for the counter. Both the prescaler and counter are common to both PWM channels. The 8-bit counter counts modulo 255 i.e. from 0 to 254 inclusive. The value of the 8-bit counter is compared to the contents of two registers: PWM0 and PWM1. Provided the contents of either of these registers is greater than the counter value, the output of $\overline{PWM0}$ or $\overline{PMW1}$ is set LOW. If the contents of these registers are equal to, or less than the counter value, the output will be HIGH. The pulse-width-ratio is therefore defined by the contents of the registers PWM0 and PWM1. The pulse-width-ratio is in the range of 0 to 255/255 and may be programmed in increments of 1/255.

The repetition frequency f_{PWM}, at the $\overline{PWMn}$ outputs is given by:

$$f_{PWM} = \frac{f_{osc}}{2 \times (1 + PWMP) \times 255}$$

When using an oscillator frequency of 12 MHz for example, the above formula gives a repetition frequency range of 92 Hz to 23,5 kHz.

By loading the PWM registers with either 00H or FFH, the PWM outputs can be retained at a constant HIGH or LOW level respectively. When loading FFH to the PWM registers, the 8-bit counter will never actually reach this value.

Both $\overline{PWMn}$ output pins are driven by push-pull drivers, and are not shared with any other function.

Prescaler frequency control register PWMP

PWMP (FEH)	7	6	5	4	3	2	1	0
	MSB							LSB

Bit	Function
PWMP.0-7	Prescaler division factor = (PWMP) + 1

Pulse width registers PWM0 and PWM1

PWM0 (FCH) PWM1 (FDH)	7	6	5	4	3	2	1	0
	MSB							LSB

Bit	Function
PWM0.0-7 PWM1.0-7	Low/high ratio of $\overline{PWMn}$ signals $= \frac{(PWMn)}{255 - (PWMn)}$

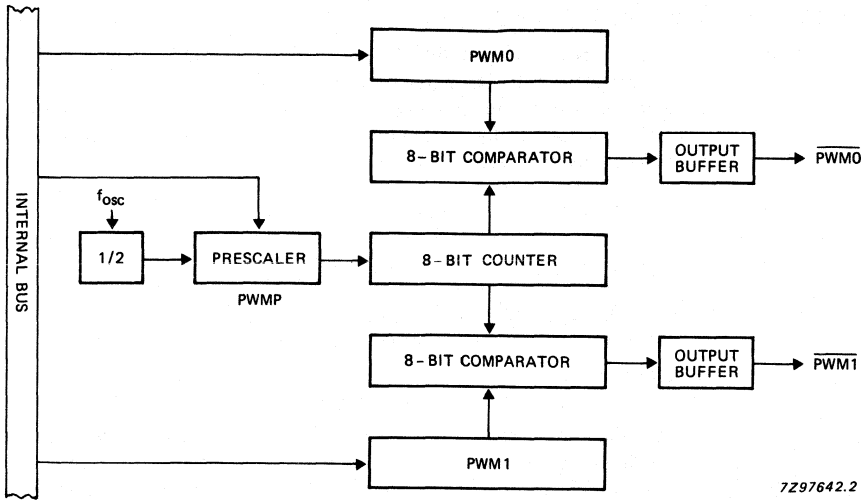


Fig. 7 Functional diagram of pulse width modulated outputs.

DEVELOPMENT DATA

Analogue input pins

The analogue input circuitry consists of an 8-input analogue multiplexer and an ADC with 8-bit resolution. The analogue reference voltage and analogue power supplies are connected via separate input pins. The conversion takes 24 machine cycles i.e. 24 μs at 12 MHz oscillator frequency.

The ADC is controlled using the ADCON control register. Input channels are selected by the analogue multiplexer, care of ADCON register bits 0-2.

ADC control register ADCON

	7	6	5	4	3	2	1	0
ADCON (C5H)			ADEX	ADCI	ADCS	AADR2	AADR1	AADR0

Bit	Symbol	Function
ADCON.5	ADEX	Enable external start of conversion by STADC 0 = Conversion cannot be started externally by STADC 1 = Conversion can be started externally by STADC
ADCON.4	ADCI	ADC interrupt flag: this flag is set when an ADC conversion result is ready to be read. An interrupt is invoked if this is enabled. The flag must be cleared by software. It cannot be set by software.

FUNCTIONAL DESCRIPTION (continued)

ADCON.3 ADCS — ADC start and status: setting this bit starts an ADC conversion. It may be set by software or by the external signal STADC. The ADC logic ensures that this signal is HIGH while the ADC is busy. On completion of the conversion, ADCS is reset at the same time the interrupt flag ADCI is set. ADCS can not be reset by software.

ADCI	ADCS	OPERATION
0	0	ADC not busy, a conversion can be started
0	1	ADC busy, start of a new conversion is blocked
1	0	Conversion completed, start of a new conversion is blocked
1	1	Not possible

ADCON.2 ADDR2
ADCON.1 AADR1
ADCON.0 AADR0

Analogue input select: this binary coded address selects one of the eight analogue port bits of P5 to be input to the converter. It can only be changed when ADCI and ADCS are both LOW. AADR2 is the most significant bit (100 selects the ADC4 analogue input channel).

The completion of the 8-bit ADC conversion is flagged by ADCI in the ADCON register and the result is stored in special function register ADCH.

An ADC conversion in progress is unaffected by an external or software ADC start. The result of a completed conversion remains unaffected provided ADCI = logic 1. While ADCS = logic 1 or ADCI = logic 1, a new ADC START will be blocked and consequently lost. An ADC conversion already in progress is aborted when the Idle or Power-down mode is entered. The result of a completed conversion (ADCI = logic 1) remains unaffected when entering the Idle mode.

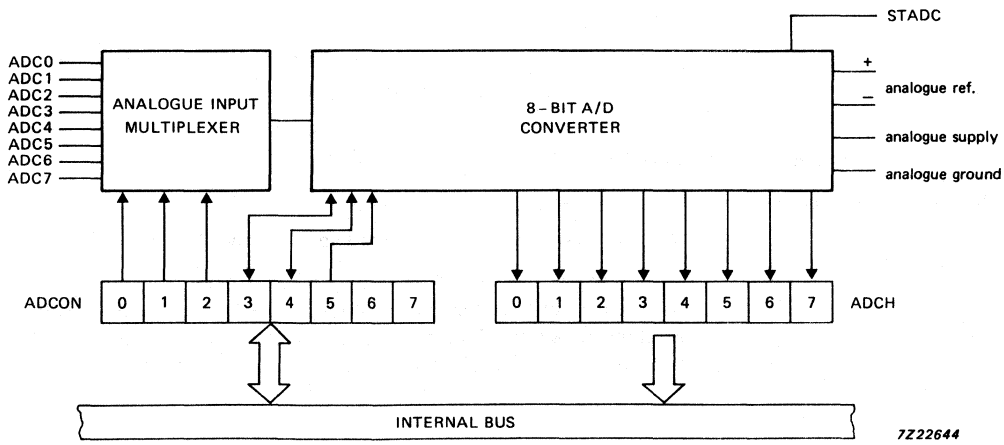


Fig. 8 Functional diagram of analogue input.

Timer/event counters

The PCB83C562 contains three 16-bit timer/event counters: Timer 0, Timer 1 and Timer T2 and one 8-bit timer, T3. Timer 0 and Timer 1 may be programmed to carry out the following functions:

- Measure time intervals and pulse durations
- Count events
- Generate interrupt requests

Timer 0 and Timer 1 can be programmed independently to operate in three modes:

- Mode 0: 8-bit timer or 8-bit counter each with divide-by-32 prescaler
- Mode 1: 16-bit time-interval or event counter
- Mode 2: 8-bit time-interval or event counter with automatic reload upon overflow

Timer 0 can be programmed to operate in an additional mode as follows:

- Mode 3: one 8-bit time-interval or event counter and one 8-bit time-interval counter

When Timer 0 is in Mode 3, Timer 1 can be programmed to operate in Modes 0, 1 or 2 but cannot set an interrupt request flag or generate an interrupt. However the overflow from Timer 1 can be used to pulse the serial port transmission-rate generator.

An example of the frequency handling range of these counters using a 12 MHz crystal is as follows:

- In the timer function, the timer is incremented at a frequency of 1 MHz – a division by 12 of the oscillator frequency
- 0 Hz to an upper limit of 0,5 MHz when programmed for external inputs

Both internal and external inputs can be gated to the counter by a second external source for directly measuring pulse durations.

The counters are started and stopped under software control. Each one sets its interrupt request flag when it overflows from all logic 1s to all logic 0s (or automatic reload value), with the exception of mode 3 as previously described.

Timer T2

Timer T2 is a 16 bit timer/counter which has, coupled to it, capture and compare facilities. The operational diagram is shown in Fig. 9.

The 16 bit timer/counter is clocked via a prescaler with a programmable division factor of 1, 2, 4 or 8. The input of the prescaler is clocked with 1/12 of the oscillator frequency, or with positive edges on the T2 input, or it is switched to the off position. This prescaler is cleared if its division factor or its input source is changed, or if the timer/counter is reset. T2 is readable 'on the fly', but posseses no extra read latches; this means that software precautions have to be taken against misinterpretation in overflow from least to most significant byte during read. T2 is not loadable and is reset by the RST signal or at the positive edge of the input signal RT2, if enabled. In the Idle mode the timer/counter and prescaler are reset and halted.

T2 is connected to four 16-bit capture registers: CT0, CT1, CT2 and CT3. These registers are loaded with the contents of T2 and an interrupt is requested upon receipt of the input signals CT0I, CT1I, CT2I or CT3I. These input signals are shared with Port 1. Using the capture register CTCON, these inputs may capture on a positive edge, a negative edge or on either a positive or negative edge.

FUNCTIONAL DESCRIPTION (continued)

The contents of the compare registers CM0, CM1 and CM2 are continually compared with the counter value of Timer T2. When a match is found an interrupt may be invoked. Using the match signal of CM0, the controller sets bits 0-5 of Port 4, if the corresponding bits of the set enable register STE are logic 1.

Considering a match with CM1, if the corresponding bits of the reset/toggle enable register RTE are logic 1, then the controller will use the match signal to reset bits 0-5 of Port 4. Bits 6 and 7 of Port 4 may be 'toggled' by the signal that indicates a match between Timer T2 and CM2 if the corresponding bits of RTE are logic 1. CM0, CM1 and CM2 are reset by the RST signal.

Port 4 can be read and written by software without affecting the toggle, set and reset signals. At byte overflow of the least significant byte, or at a 16-bit overflow of the timer/counter, an interrupt sharing the same interrupt vector is requested. Either or both of these overflows can be programmed to request an interrupt.

All interupt flags must be reset by software.

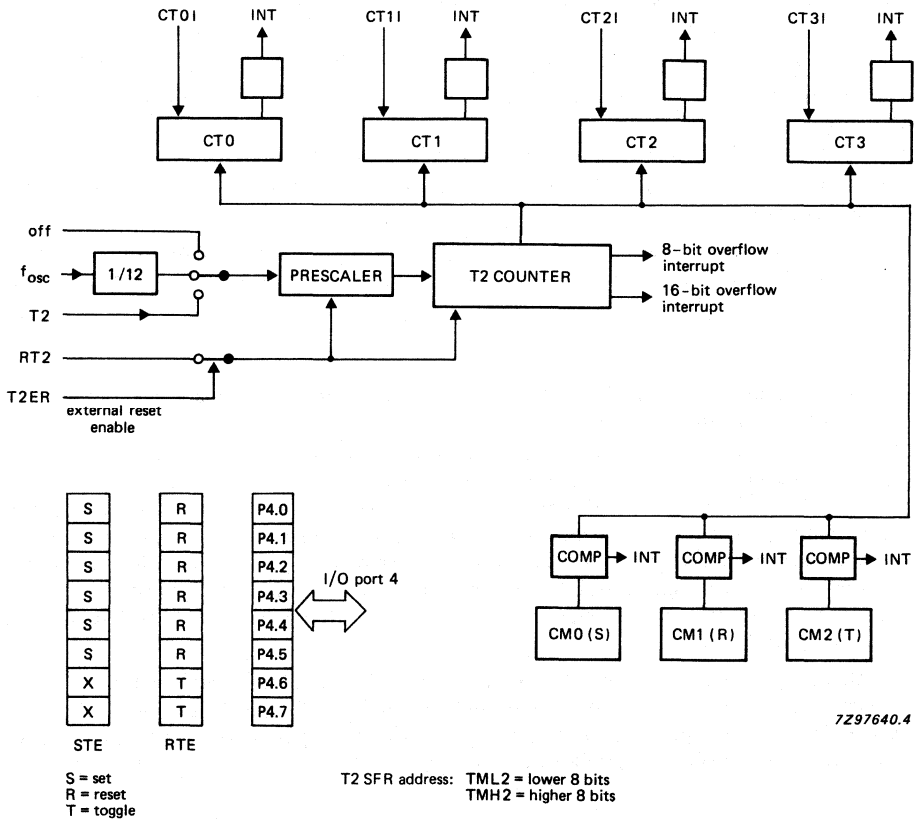


Fig. 9 Block diagram of Timer T2 configuration.

Counter control register TM2CON

	7	6	5	4	3	2	1	0
TM2CON (EAH)	T2IS1	T2IS0	T2ER	T2B0	T2P1	T2P0	T2MS1	T2MS0
	MSB							LSB

Bit	Symbol	Function
TM2CON.7	T2IS1	Timer 2 16 bit overflow interrupt select
TM2CON.6	T2IS0	Timer 2 byte overflow interrupt select
TM2CON.5	T2ER	Timer 2 external reset enable
TM2CON.4	T2B0	Timer 2 byte overflow interrupt flag
TM2CON.3	T2P1	Timer 2 prescaler select
TM2CON.2	T2P0	
TM2CON.1	T2MS1	Timer 2 mode select
TM2CON.0	T2MS0	

T2P1	T2P0	T2 clock
0	0	clock source
0	1	clock source/2
1	0	clock source/4
1	1	clock source/8

T2MS1	T2MS0	Mode Selected
0	0	Timer T2 is halted
0	1	T2 clock source = $f_{osc}/12$
1	0	Test mode; do not use
1	1	T2 clock source = pin T2

Capture control register CTCON

	7	6	5	4	3	2	1	0
CTCON (EBH)	CTN3	CTP3	CTN2	CTP2	CTN1	CTP1	CTN0	CTP0
	MSB							LSB

Bit	Symbol	Capture/Interrupt on:
CTCON.7	CTN3	CT3I negative edge
CTCON.6	CTP3	CT3I positive edge
CTCON.5	CTN2	CT2I negative edge
CTCON.4	CTP2	CT2I positive edge
CTCON.3	CTN1	CT1I negative edge
CTCON.2	CTP1	CT1I positive edge
CTCON.1	CTN0	CT0I negative edge
CTCON.0	CTP0	CT0I positive edge

FUNCTIONAL DESCRIPTION (continued)

Timer interrupt flag register TM2IR

	7	6	5	4	3	2	1	0
TM2IR (C8H)	T2OV	CMI2	CMI1	CMI0	CTI3	CTI2	CTI1	CTI0
	MSB							LSB

Bit	Symbol	Function
TM2IR.7	T2OV	T2 16-bit overflow interrupt flag
TM2IR.6	CMI2	CM2 interrupt flag
TM2IR.5	CMI1	CM1 interrupt flag
TM2IR.4	CMI0	CM0 interrupt flag
TM2IR.3	CTI3/INT5	CT3 interrupt flag
TM2IR.2	CTI2/INT4	CT2 interrupt flag
TM2IR.1	CTI1/INT3	CT1 interrupt flag
TM2IR.0	CTI0/INT2	CT0 interrupt flag

Interrupt enable register IEN1 is used to enable/disable timer 2 interrupts.
Interrupt priority register IP1 is used to determine the timer interrupt priority.

The set enable register STE

	7	6	5	4	3	2	1	0
STE (EEH)	TG4.7	TG4.6	SP45	SP44	SP43	SP42	SP41	SP40
	MSB							LSB

Bit	Symbol	Function
STE.7	TG4.7	
STE.6	TG4.6	
STE.5	SP45	if logic 1 then P4.5 is set on a match between CM0 and T2
STE.4	SP44	if logic 1 then P4.4 is set on a match between CM0 and T2
STE.3	SP43	if logic 1 then P4.3 is set on a match between CM0 and T2
STE.2	SP42	if logic 1 then P4.2 is set on a match between CM0 and T2
STE.1	SP41	if logic 1 then P4.1 is set on a match between CM0 and T2
STE.0	SP40	if logic 1 then P4.0 is set on a match between CM0 and T2

If STE.n is logic 0 then P4.n is not affected by a match between CM0 and T2.

The reset/toggle enable register RTE

	7	6	5	4	3	2	1	0
RTE (EFH)	TP47	TP46	RP45	RP44	RP43	RP42	RP41	RP40
	MSB							LSB

Bit	Symbol	Function
RTE.7	TP47	if logic 1 then P4.7 toggles on a match between CM0 and T2
RTE.6	TP46	if logic 1 then P4.6 toggles on a match between CM0 and T2
RTE.5	RP45	if logic 1 then P4.5 is reset on a match between CM1 and T2
RTE.4	RP44	if logic 1 then P4.4 is reset on a match between CM1 and T2
RTE.3	RP43	if logic 1 then P4.3 is reset on a match between CM1 and T2
RTE.2	RP42	if logic 1 then P4.2 is reset on a match between CM1 and T2
RTE.1	RP41	if logic 1 then P4.1 is reset on a match between CM1 and T2
RTE.0	RP40	if logic 1 then P4.0 is reset on a match between CM1 and T2

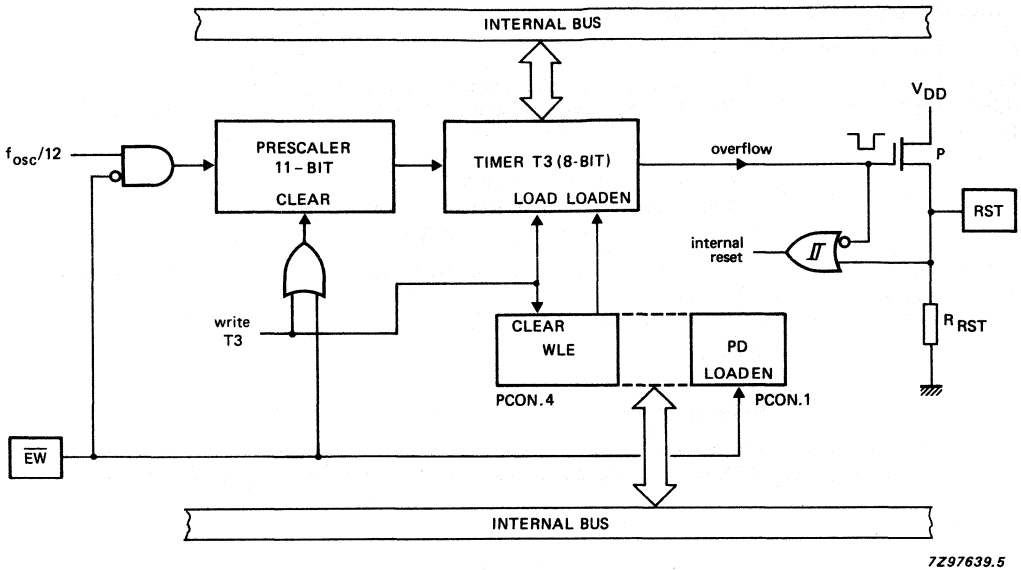


Fig. 10 Functional diagram of T3 watchdog timer.

DEVELOPMENT DATA

T3 The watchdog timer (see Fig. 10)

In addition to Timer T2 and the standard timers, a watchdog timer consisting of an 11-bit prescaler and an 8-bit timer are also incorporated.

The timer frequency is derived from the oscillator frequency using the following formula:

$$f_{timer} = \frac{f_{osc}}{12 \times 2048}$$

When a timer overflow occurs, the microcontroller is reset and a reset output pulse is generated at pin RST. To prevent a system reset the timer must be reloaded in time by the application software. If the processor suffers a hardware/software malfunction, the software will fail to reload the timer. This failure will produce a reset upon overflow thus preventing the processor running out of control.

The watchdog timer can only be reloaded if the condition flag WLE = PCON.4 has been previously set by software.

At the moment the counter is loaded the condition flag is automatically cleared.

The time interval between the timer's reloading and the occurance of a reset, is dependent upon the reloaded value. For example, this may range from 2 ms to 0,5 s when using an oscillator frequency of 12 MHz.

In the Idle state the watchdog timer and reset circuitry remain active.

The watchdog timer is controlled by the watchdog enable pin ($\overline{EW}$). A logic 0 enables the watchdog timer and disables the Power-down mode. A logic 1 disables and resets the watchdog timer and enables the Power-down mode.

Serial I/O

The PCB83C562 is equipped with a full duplex UART port identical to the serial port of the PCB80C51. For a more detailed description see the "Single-Chip 8-Bit Microcontrollers User Manual".

FUNCTIONAL DESCRIPTION (continued)

Idle and Power-down operation (see Fig. 11)

Idle mode operation permits the interrupt, serial ports and timer blocks to continue to function while the CPU is halted. The following functions are also switched off when the processor enters the Idle mode.

Timer T2	(stopped and reset)
PWM0, PWM1	(reset, output = HIGH)
ADC	(aborted if in progress)

The following functions remain active during Idle mode. These functions may generate an interrupt or reset and thus end the Idle mode.

Timer 0, Timer 1
Timer T3
SIO
External interrupt

The Power-down operation freezes the oscillator. The Power-down mode can only be activated by setting the PD bit in the PCON register. The PD bit can only be set if the EWN input is HIGH.

Idle mode

The instruction that sets PCON.0 is the last instruction executed in the normal operating mode before Idle mode is activated. Once in the Idle mode, the CPU status is preserved in its entirety: the Stack Pointer, Program Counter, Program Status Word, Accumulator, RAM and all other registers maintain their data during Idle mode. The status of the external pins during Idle mode is shown in Table 1.

There are two ways to terminate the Idle mode:

Activation of any enabled interrupt will cause PCON.0 to be cleared by hardware terminating Idle mode. The interrupt is serviced, and following return from interrupt instruction RETI, the next instruction to be executed will be the one which follows the instruction that wrote a logic 1 to PCON.0.

The flag bits GF0 and GF1 may be used to determine whether the interrupt was received during normal execution or during the Idle mode. For example, the instruction that writes to PCON.0 can also set or clear one or both flag bits. When Idle mode is terminated by an interrupt, the service routine can examine the status of the flag bits.

The second way of terminating the Idle mode is with an external hardware reset, or an internal reset caused by an overflow of Timer T3. Since the oscillator is still running, the hardware reset is required to be active for two machine cycles (24 oscillator periods) to complete the reset operation.

Power-down mode

The instruction that sets PCON.1 is the last executed prior to going into the Power-down mode. Once in Power-down mode, the oscillator is stopped. Only the contents of the on-chip RAM are preserved. The Special Function Registers are not saved. A hardware reset is the only way of exiting the Power-down mode.

In the Power-down mode, V_{DD} may be reduced to minimize circuit power consumption. The voltage must not be reduced until the Power-down mode is entered, but must be restored before the hardware reset is applied which will free the oscillator. Reset should not be released until the oscillator has restarted and stabilized.

The status of the external pins during Power-down mode is shown in Table 1. If the Power-down mode is activated while in external program memory, the port data that is held in the Special Function Register P2 is restored to Port 2. If the data is a logic 1, the port pin is held HIGH during the Power-down mode by the strong pull-up transistor p1 (see Fig. 6).

Table 1 Status of external pins during Idle and Power-down modes

mode	memory	ALE	$\overline{PSEN}$	Port 0	Port 1	Port 2	Port 3	Port 4	$\overline{PWM0}/\overline{PWM1}$
Idle (1)	internal	1	1	port data	port data	port data	port data	port data	HIGH
Idle (1)	external	1	1	floating	port data	address	port data	port data	HIGH
Power-down	internal	0	0	port data	port data	port data	port data	port data	HIGH
Power-down	external	0	0	floating	port data	port data	port data	port data	HIGH

Power control register (PCON)

These special modes are activated by software via the Special Function Register PCON. Its hardware address is 87H. PCON is not bit addressable.

	7	6	5	4	3	2	1	0
PCON (87H)	SMOD	–	–	WLE	GF1	GF0	PD	IDL

Bit	Symbol	Function
PCON.7	SMOD	Double Baud rate bit. When set to logic 1 the baud rate is doubled when the serial port SIO0 is being used in modes 1, 2 or 3
PCON.6	–	(reserved)
PCON.5	–	(reserved)
PCON.4	WLE	Watchdog load enable. This flag must be set by software prior to loading T3 (watchdog timer). It is cleared when T3 is loaded
PCON.3	GF1	General-purpose flag bit
PCON.2	GF0	General-purpose flag bit
PCON.1	PD	Power-down bit. Setting this bit activates Power-down mode. It can only be set if input $\overline{EW}$ is high.
PCON.0	IDL	Idle mode bit. Setting this bit activates the Idle mode

If logic 1s are written to PD and IDL at the same time, PD takes precedence. The reset value of PCON is (0XX00000).

DEVELOPMENT DATA

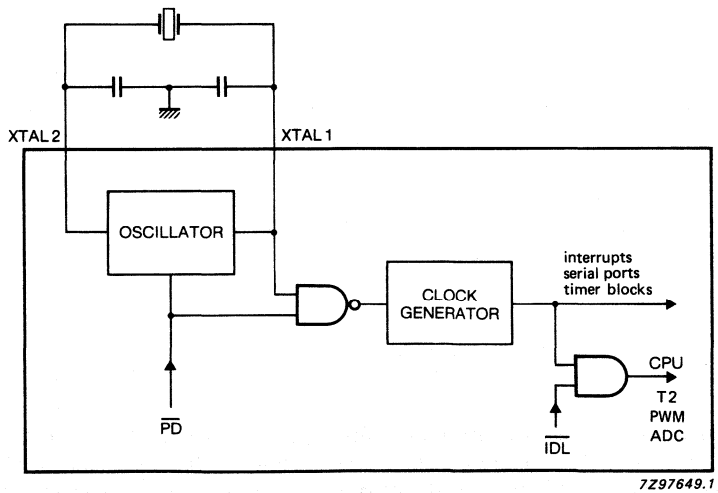


Fig. 11 Internal Idle and Power-down clock configuration.

FUNCTIONAL DESCRIPTION (continued)

Interrupt system (see Fig. 12)

External events and the real-time-driven on-chip peripherals require service by the CPU asynchronous to the execution of any particular section of code. To tie the asynchronous activities of these functions to normal program execution a multiple-source, two-priority-level, nested interrupt system is provided. Interrupt response latency is from 3 μs to 7 μs when using a 12 MHz crystal.

The PCB83C562 acknowledges interrupt requests from fourteen sources as follows:

- INT0 and INT1; externally via pins 26 and 27 respectively
- Timer 0 and Timer 1; from the two internal counters
- Timer T2, 8 separate interrupts; 4 capture interrupts*, 3 compare interrupts and an overflow interrupt
- ADC end-of-conversion interrupt
- UART serial I/O port interrupt
- INT2 to INT5; externally via pins 16 to 19 respectively*

Each interrupt vectors to a separate location in program memory for its service program. Each source can be individually enabled or disabled by a corresponding bit in the IE0 or IE1 register, moreover each interrupt may be programmed to a high or low priority level using a corresponding bit in the IP0 or IP1 register. Also all enabled sources can be globally disabled or enabled. Both external interrupts can be programmed to be level-activated or transition-activated, and an active LOW level allows "wire-ORing" of several interrupt sources to the input pin.

* Pins 16 to 19 share the dual input function of capture register interrupts CT0I to CT3I (when used in conjunction with Timer T2), or external interrupts INT2 to INT5 (used exclusive of Timer T2). Note, these two functions are both externally edge triggered and share the same ROM based interrupt vectors CT0 to CT3.

DEVELOPMENT DATA

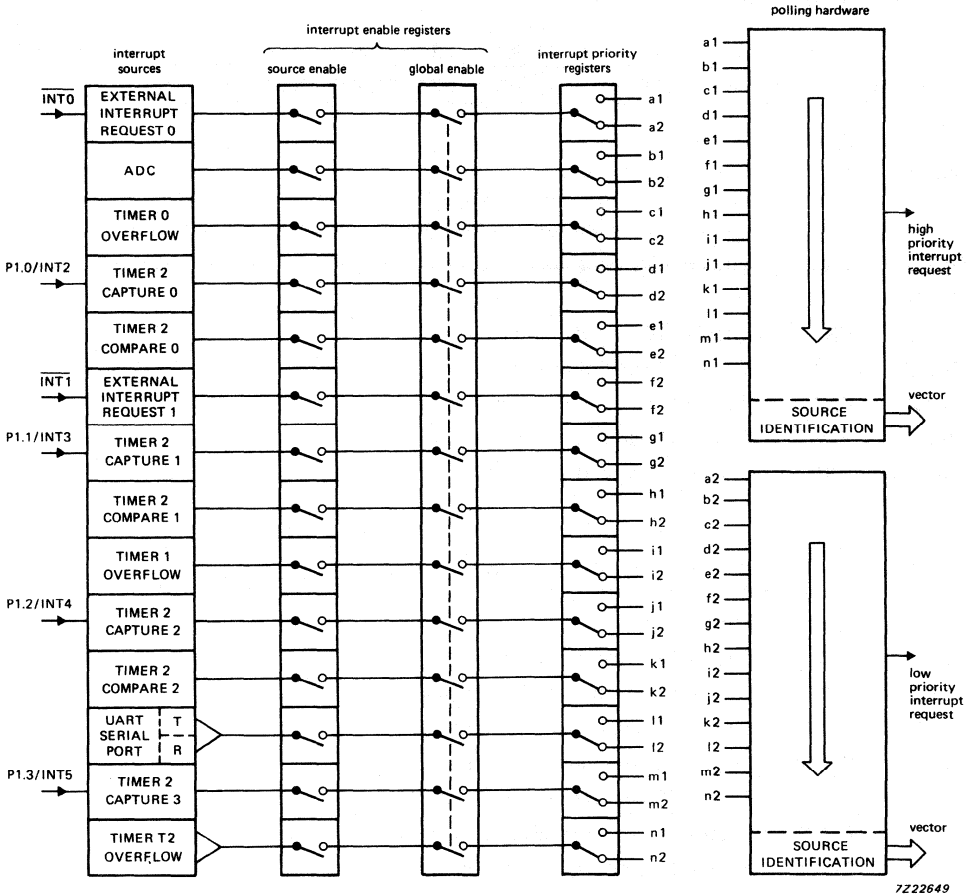


Fig. 12 Interrupt system.

FUNCTIONAL DESCRIPTION (continued)

Interrupt enable registers

	7	6	5	4	3	2	1	0
IEN0 (A8H)	EA	EAD	–	ES	ET1	EX1	ET0	EX0

Bit	Symbol	Function
IEN0.7	EA	General enable/disable control Logic 0 = No interrupt is enabled Logic 1 = Any individually enabled interrupt will be accepted
IEN0.6	EAD	Enable ADC interrupt
IEN0.5	–	Unused
IEN0.4	ES	Enable SIO (UART) interrupt
IEN0.3	ET1	Enable Timer 1 interrupt
IEN0.2	EX1	Enable External 1 interrupt
IEN0.1	ET0	Enable Timer 0 interrupt
1EN0.0	EX0	Enable External 0 interrupt

	7	6	5	4	3	2	1	0
IEN1 (E8H)	ET2	ECM2	ECM1	ECM0	ECT3	ECT2	ECT1	ECT0

Bit	Symbol	Function
IEN1.7	ET2	Enable T2 overflow interrupt(s)
IEN1.6	ECM2	Enable T2 comparator 2 interrupt
IEN1.5	ECM1	Enable T2 comparator 1 interrupt
IEN1.4	ECM0	Enable T2 comparator 0 interrupt
IEN1.3	ECT3	Enable T2 capture register 3 interrupt
IEN1.2	ECT2	Enable T2 capture register 2 interrupt
IEN1.1	ECT1	Enable T2 capture register 1 interrupt
IEN1.0	ECT0	Enable T2 capture register 0 interrupt

Where logic 0 = interrupt disabled
and logic 1 = interrupt enabled

Interrupt priority registers

	7	6	5	4	3	2	1	0
IP0 (B8H)	–	PAD	–	PS0	PT1	PX1	PT0	PX0

Bit	Symbol	Function
IP0.7	–	Unused
IP0.6	PAD	ADC interrupt priority level
IP0.5	–	Unused
IP0.4	PS0	SIO0 (UART) interrupt priority level
IP0.3	PT1	Timer 1 interrupt priority level
IP0.2	PX1	External interrupt 1 priority level
IP0.1	PT0	Timer 0 interrupt priority level
IP0.0	PX0	External interrupt 0 priority level

	7	6	5	4	3	2	1	0
IP1 (F8H)	PT2	PCM2	PCM1	PCM0	PCT3	PCT2	PCT1	PCT0

Bit	Symbol	Function
IP1.7	PT2	T2 overflow interrupt(s) priority level
IP1.6	PCM2	T2 comparator 2 priority interrupt level
IP1.5	PCM1	T2 comparator 1 priority interrupt level
IP1.4	PCM0	T2 comparator 0 priority interrupt level
IP1.3	PCT3	T2 capture register 3 priority interrupt level
IP1.2	PCT2	T2 capture register 2 priority interrupt level
IP1.1	PCT1	T2 capture register 1 priority interrupt level
IP1.0	PCT0	T2 capture register 0 priority interrupt level

Interrupt priority levels are as follows: logic 0 = low priority
logic 1 = high priority

DEVELOPMENT DATA

FUNCTIONAL DESCRIPTION (continued)

Table 2 shows the interrupt vectors. The vector indicates the ROM location where the appropriate interrupt service routine starts.

Table 2 Interrupt vectors

Source		Vector
External 0	X0	0003H
Timer 0 overflow	T0	000BH
External 1	X1	0013H
Timer 1 overflow	T1	001BH
Serial I/O (UART)	S0	0023H
T2 capture 0	CT0	0033H
T2 capture 1	CT1	003BH
T2 capture 2	CT2	0043H
T2 capture 3	CT3	004BH
ADC completion	ADC	0053H
T2 compare 0	CM0	005BH
T2 compare 1	CM1	0063H
T2 compare 2	CM2	006BH
T2 overflow	T2	0073H

Interrupt priority

Each interrupt source can be either high priority or low priority. If both levels are requested simultaneously, the processor will branch to the high priority vector. If there are simultaneous requests from sources of the same priority, then interrupts will be serviced in the following order:

X0, ADC, T0, CT0, CM0, X1, CT1, CM1, T1, CT2, CM2, S0, CT3, T2

A low priority interrupt can only be interrupted by high priority interrupt. A high priority interrupt can not be interrupted.

Oscillator circuitry

The oscillator circuitry of the PCB83C562 is a single-stage inverting amplifier in a Pierce oscillator configuration. The circuitry between XTAL 1 and XTAL 2 is basically an inverter biased to the transfer point. Either a crystal or ceramic resonator can be used as the feedback element to complete the oscillator circuitry. Both are operated in parallel resonance. XTAL 1 (pin 35) is the high gain amplifier input, and XTAL 2 (pin 34) is the output (see Fig. 13). To drive the PCB83C562 externally, XTAL 1 is driven from an external source and XTAL 2 left open-circuit (see Fig. 14).

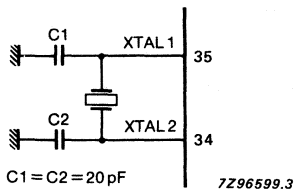


Fig. 13 PCB83C562 oscillator circuit.

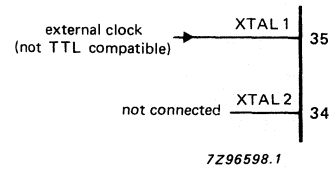


Fig. 14 Driving the PCB83C562 from an external source.

DEVELOPMENT DATA

Reset circuitry (see Fig. 15)

The reset circuitry for the PCB83C562 is connected to the reset pin RST. A Schmitt trigger is used at the input for noise rejection. The output of the Schmitt trigger is sampled by the reset circuitry every machine cycle.

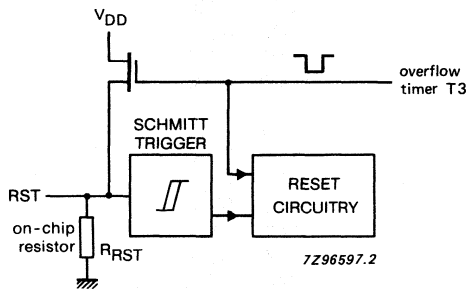


Fig. 15 On-chip reset configuration.

A reset is accomplished by holding the RST pin HIGH for at least two machine cycles (24 oscillator periods). The CPU responds by executing an internal reset. During reset ALE and $\overline{PSEN}$ output a HIGH level. In order to perform a correct reset, this level must not be affected by external elements.

Also with the PCB83C562, the RST line can be pulled HIGH internally by a pull-up transistor activated by the watchdog timer T3. The length of the output pulse from T3 is 3 machine cycles. A pulse of such short duration is necessary in order to recover from a processor or system fault as fast as possible.

FUNCTIONAL DESCRIPTION (continued)

N.B. It can be seen that the short reset pulse from Timer T3 cannot discharge the power-on reset capacitor (see Fig. 16). Consequently, when the watchdog timer is also used to reset external devices this capacitor arrangement should not be connected to the RST pin, and an extra circuit should be used to perform the power-on reset operation. It should be remembered that a Timer T3 overflow, if enabled, will force a reset condition to the PCB83C562 by an internal connection, whether the output RST is tied LOW or not.

The internal reset is executed during the second cycle in which RST is HIGH and is repeated every cycle until RST goes LOW. It leaves the internal registers as follows:

Register	Content
ACC	0000 0000
ADCON	xx00 0000
ADCH	xxxx xxxx
B	0000 0000
CML0 - CML2	0000 0000
CMH0 - CMH2	0000 0000
CTCON	0000 0000
CTL0 - CTL3	xxxx xxxx
CTH0 - CTH3	xxxx xxxx
DPL	0000 0000
DPH	0000 0000
IEN0	0000 0000
IEN1	0000 0000
IP0	x000 0000
IP1	0000 0000
PCH	0000 0000
PCL	0000 0000
PCON	0xx0 0000
PSW	0000 0000
PWM0	0000 0000
PWM1	0000 0000
PWMP	0000 0000
P0 - P4	1111 1111
P5	xxxx xxxx
RTE	0000 0000
S0BUF	xxxx xxxx
S0CON	0000 0000
SP	0000 0111
STE	1100 0000
TCON	0000 0000
TH0, TH1	0000 0000
TMH2	0000 0000
TL0, TL1	0000 0000
TML2	0000 0000
TMOD	0000 0000
TM2CON	0000 0000
TM2IR	0000 0000
T3	0000 0000

x = undefined content

The internal RAM is not affected by reset. When V_{DD} is turned on, the RAM content is indeterminate.

Power-on reset (see Fig. 16)

When V_{DD} is turned on, and provided its rise-time does not exceed 10 ms, an automatic reset can be obtained by connecting the RST pin to V_{DD} via a 2,2 μF capacitor. When the power is switched on, the voltage on the RST pin is equal to V_{DD} minus the capacitor voltage, and decreases from V_{DD} as the capacitor charges through the internal resistor (R_{RST}) to ground. The larger the capacitor, the more slowly V_{RST} decreases. V_{RST} must remain above the lower threshold of the Schmitt trigger long enough to effect a complete reset. The time required is the oscillator start-up time, plus 2 machine cycles.

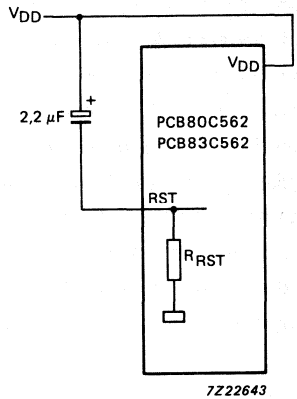


Fig. 16 Power-on reset.

INSTRUCTION SET

Table 3 Instruction set description

mnemonic		description	bytes/ cycles		opcode (hex.)
Arithmetic operations					
ADD	A,Rr	Add register to A	1	1	2*
ADD	A,direct	Add direct byte to A	2	1	25
ADD	A,@Ri	Add indirect RAM to A	1	1	26, 27
ADD	A,#data	Add immediate data to A	2	1	24
ADDC	A,Rr	Add register to A with carry flag	1	1	3*
ADDC	A,direct	Add direct byte to A with carry flag	2	1	35
ADDC	A,@Ri	Add indirect RAM to A with carry flag	1	1	36, 37
ADDC	A,#data	Add immediate data to A with carry flag	2	1	34
SUBB	A,Rr	Subtract register from A with borrow	1	1	9*
SUBB	A,direct	Subtract direct byte from A with borrow	2	1	95
SUBB	A,@Ri	Subtract indirect RAM from A with borrow	1	1	96, 97
SUBB	A,#data	Subtract immediate data from A with borrow	2	1	94
INC	A	Increment A	1	1	04
INC	Rr	Increment register	1	1	0*
INC	direct	Increment direct byte	2	1	05
INC	@Ri	Increment indirect RAM	1	1	06, 07
DEC	A	Decrement A	1	1	14
DEC	Rr	Decrement register	1	1	1*
DEC	direct	Decrement direct byte	2	1	15
DEC	@Ri	Decrement indirect RAM	1	1	16, 17
INC	DPTR	Increment data pointer	1	2	A3
MUL	AB	Multuply A & B	1	4	A4
DIV	AB	Divide A by B	1	4	84
DA	A	Decimal adjust A	1	1	D4

mnemonic		description	bytes/ cycles		opcode (hex.)
Logic operations					
ANL	A,Rr	AND register to A	1	1	5*
ANL	A,direct	AND direct byte to A	2	1	55
ANL	A,@Ri	AND indirect RAM to A	1	1	56, 57
ANL	A,#data	AND immediate data to A	2	1	54
ANL	direct,A	AND A to direct byte	2	1	52
ANL	direct,#data	AND immediate data to direct byte	3	2	53
ORL	A,Rr	OR register to A	1	1	4*
ORL	A,direct	OR direct byte to A	2	1	45
ORL	A,@Ri	OR indirect RAM to A	1	1	46, 47
ORL	A,#data	OR immediate data to A	2	1	44
ORL	direct, A	OR A to direct byte	2	1	42
ORL	direct,#data	OR immediate data to direct byte	3	2	43
XRL	A,Rr	Exclusive-OR register to A	1	1	6*
XRL	A,direct	Exclusive-OR direct byte to A	2	1	65
XRL	A,@Ri	Exclusive-OR indirect RAM to A	1	1	66, 67
XRL	A,#data	Exclusive-OR immediate data to A	2	1	64
XRL	direct,A	Exclusive-OR A to direct byte	2	1	62
XRL	direct,#data	Exclusive-OR immediate data to direct byte	3	2	63
CLR	A	Clear A	1	1	E4
CPL	A	Complement A	1	1	F4
RL	A	Rotate A left	1	1	23
RLC	A	Rotate A left through the carry flag	1	1	33
RR	A	Rotate A right	1	1	03
RRC	A	Rotate A right through the carry flag	1	1	13
SWAP	A	Swap nibbles within A	1	1	C4

INSTRUCTION SET (continued)

mnemonic		description	bytes/ cycles		opcode (hex.)
Data transfer					
MOV	A,Rr	Move register to A	1	1	E*
MOV	A,direct**	Move direct byte to A	2	1	E5
MOV	A,@Ri	Move indirect RAM to A	1	1	E6, E7
MOV	A,#data	Move immediate data to A	2	1	74
MOV	Rr,A	Move A to register	1	1	F*
MOV	Rr,direct	Move direct byte to register	2	2	A*
MOV	Rr,#data	Move immediate data to register	2	1	7*
MOV	direct,A	Move A to direct byte	2	1	F5
MOV	direct,Rr	Move register to direct byte	2	2	8*
MOV	direct,direct	Move direct byte to direct byte	3	2	85
MOV	direct,@Ri	Move indirect RAM to direct byte	2	2	86, 87
MOV	direct,#data	Move immediate data to direct byte	3	2	75
MOV	@Ri,A	Move A to indirect RAM	1	1	F6, F7
MOV	@Ri,direct	Move direct byte to indirect RAM	2	2	A6, A7
MOV	@Ri,#data	Move immediate data to indirect RAM	2	1	76, 77
MOV	DPTR,#data16	Load data pointer with a 16-bit constant	3	2	90
MOVC	A,@A+DPTR	Move code byte relative to DPTR to A	1	2	93
MOVC	A,@A+PC	Move code byte relative to PC to A	1	2	83
MOVX	A,@Ri	Move external RAM (8-bit address) to A	1	2	E2, E3
MOVX	A,@DPTR	Move external RAM (16-bit address) to A	1	2	E0
MOVX	@Ri,A	Move A to external RAM (8-bit address)	1	2	F2, F3
MOVX	@DPTR,A	Move A to external RAM (16-bit address)	1	2	F0
PUSH	direct	Push direct byte onto stack	2	2	C0
POP	direct	Pop direct byte from stack	2	2	D0
XCH	A,Rr	Exchange register with A	1	1	C*
XCH	A,direct	Exchange direct byte with A	2	1	C5
XCH	A,@Ri	Exchange indirect RAM with A	1	1	C6, C7
XCHD	A,@Ri	Exchange LOW-order nibble indirect RAM with A	1	1	D6, D7

** MOV A,ACC is not a valid instruction.

mnemonic		description	bytes/ cycles		opcode (hex.)
Boolean variable manipulation					
CLR	C	Clear carry flag	1	1	C3
CLR	bit	Clear direct bit	2	1	C2
SETB	C	Set carry flag	1	1	D3
SETB	bit	Set direct bit	2	1	D2
CPL	C	Complement carry flag	1	1	B3
CPL	bit	Complement direct bit	2	1	B2
ANL	C,bit	AND direct bit to carry flag	2	2	82
ANL	C,/bit	AND complement of direct bit to carry flag	2	2	B0
ORL	C,bit	OR direct bit to carry flag	2	2	72
ORL	C,/bit	OR complement of direct bit to carry flag	2	2	A0
MOV	C,bit	Move direct bit to carry flag	2	1	A2
MOV	bit,C	Move carry flag to direct bit	2	2	92
Program and machine control					
ACALL	addr11	Absolute subroutine call	2	2	●1addr
LCALL	addr16	Long subroutine call	3	2	12
RET		Return from subroutine	1	2	22
RETI		Return from interrupt	1	2	32
AJMP	addr11	Absolute jump	2	2	▲1addr
LJMP	addr16	Long jump	3	2	02
SJMP	rel	Short jump (relative address)	2	2	80
JMP	@A+DPTR	Jump indirect relative to the DPTR	1	2	73
JZ	rel	Jump if A is zero	2	2	60
JNZ	rel	Jump if A is not zero	2	2	70
JC	rel	Jump if carry flag is set	2	2	40
JNC	rel	Jump if carry flag is not set	2	2	50
JB	bit,rel	Jump if direct bit is set	3	2	20
JNB	bit,rel	Jump if direct bit is not set	3	2	30
JBC	bit,rel	Jump if direct bit is set and clear bit	3	2	10
CJNE	A,direct,rel	Compare direct to A and jump if not equal	3	2	B5
CJNE	A,#data,rel	Compare immediate to A and jump if not equal	3	2	B4
CJNE	Rr,#data,rel	Compare immediate to reg. and jump if not equal	3	2	B*
CJNE	@Ri,#data,rel	Compare immediate to ind. and jump if not equal	3	2	B6, B7
DJNZ	Rr,rel	Decrement register and jump if not zero	2	2	D*
DJNZ	direct,rel	Decrement direct and jump if not zero	3	2	D5
NOP		No operation	1	1	00

DEVELOPMENT DATA

Notes to Table 3

Data addressing modes

Rr	Working register R0-R7.
direct	128 internal RAM locations and any special function register (SFR).
@Ri	Indirect internal RAM location addressed by register R0 or R1 of the actual register bank.
#data	8-bit constant included in instruction.
#data16	16-bit constant included as bytes 2 and 3 of instruction.
bit	direct addressed bit in internal RAM or SFR.
addr16	16-bit destination address. Used by LCALL and LJMP. The branch will be anywhere within the 64 K-byte program memory address space.
addr11	11-bit destination address. Used by ACALL and AJMP. The branch will be within the same 2 K-byte page of program memory as the first byte of the following instruction.
rel	Signed (two's complement) 8-bit offset byte. Used by SJMP and all conditional jumps. Range is −128 to + 127 bytes relative to first byte of the following instruction.

Hexadecimal opcode cross-reference to Table 3

* : 8, 9, A, B, C, D, E, F.
● : 11, 31, 51, 71, 91, B1, D1, F1.
▲ : 01, 21, 41, 61, 81, A1, C1, E1.

Table 4 Instruction map

first hexadecimal character of opcode ↓	second hexadecimal character of opcode →															
	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	NOP	AJMP addr11	LJMP addr16	RR A	INC A	INC dir	INC @Ri		INC Rr							
							0	1	0	1	2	3	4	5	6	7
1	JBC bit,rel	ACALL addr11	LCALL addr16	RRC A	DEC A	DEC dir	DEC @Ri		DEC Rr							
							0	1	0	1	2	3	4	5	6	7
2	JB bit,rel	AJMP addr11	RET	RL A	ADD A,#data	ADD A,dir	ADD A,@Ri		ADD A,Rr							
							0	1	0	1	2	3	4	5	6	7
3	JNB bit,rel	ACALL addr11	RETI	RLC A	ADDC A,#data	ADDC A,dir	ADDC A,@Ri		ADDC A,Rr							
							0	1	0	1	2	3	4	5	6	7
4	JC rel	AJMP addr11	ORL dir,A	ORL dir,#data	ORL A,#data	ORL A,dir	ORL A,@Ri		ORL A,Rr							
							0	1	0	1	2	3	4	5	6	7
5	JNC rel	ACALL addr11	ANL dir,A	ANL dir,#data	ANL A,#data	ANL A,dir	ANL A,@Ri		ANL A,Rr							
							0	1	0	1	2	3	4	5	6	7
6	JZ rel	AJMP addr11	XRL dir,A	XRL dir,#data	XRL A,#data	XRL A,dir	XRL A,@Ri		XRL A,Rr							
							0	1	0	1	2	3	4	5	6	7
7	JNZ rel	ACALL addr11	ORL C,bit	JMP @A+DPTR	MOV A,#data	MOV dir,#data	MOV @Ri,#data		MOV Rr,#data							
							0	1	0	1	2	3	4	5	6	7
8	SJMP rel	AJMP addr11	ANL C,bit	MOVC A,@A+PC	DIV AB	MOV dir,dir	MOV dir,@Ri		MOV dir,Rr							
							0	1	0	1	2	3	4	5	6	7
9	MOV DPTR, #data 16	ACALL addr11	MOV bit,C	MOVC A,@A+DPTR	SUBB A,#data	SUBB A,dir	SUBB A,@Ri		SUBB A,Rr							
							0	1	0	1	2	3	4	5	6	7
A	ORL C,/bit	AJMP addr11	MOV C,bit	INC DPTR	MUL AB		MOV @Ri,dir		MOV Rr,dir							
							0	1	0	1	2	3	4	5	6	7
B	ANL C,/bit	ACALL addr11	CPL bit	CPL C	CJNE A, #data, rel	CJNE A,dir,rel	CJNE @Ri,#data, rel		CJNE Rr,#data,rel							
							0	1	0	1	2	3	4	5	6	7
C	PUSH dir	AJMP addr11	CLR bit	CLR C	SWAP A	XCH A,dir	XCH A,@Ri		XCH A,Rr							
							0	1	0	1	2	3	4	5	6	7
D	POP dir	ACALL addr11	SETB bit	SETB C	DA A	DJNZ dir,rel	XCHD A,@Ri		DJNZ Rr, rel							
							0	1	0	1	2	3	4	5	6	7
E	MOVX A,@DPTR	AJMP addr11	MOVX A,@Ri		CLR A	MOV * A,dir	MOV A,@Ri		MOV A,Rr							
			0	1			0	1	0	1	2	3	4	5	6	7
F	MOVX @DPTR,A	ACALL addr11	MOVX @Ri,A		CPL A	MOV dir,A	MOV @Ri,A		MOV Rr,A							
			0	1			0	1	0	1	2	3	4	5	6	7

* MOV A,ACC is not a valid instruction.

RATINGS

Limiting values in accordance with the Absolute Maximum System (IEC 134)

parameter	symbol	min.	max.	unit
Input voltage on any pin with respect to ground (V_{SS})	V_I	−0,5	+ 6,5	V
Input, output current	I_I, I_O	–	± 5,0	mA
Input, output current on any single pin	I_I, I_O	–	± 10	mA
Total power dissipation	P_{tot}	–	1,0	W
Storage temperature range	T_{stg}	−65	+ 150	°C
Operating ambient temperature range				
PCB83C562	T_{amb}	0	+ 70	°C
PCF83C562	T_{amb}	−40	+ 85	°C
PCA83C562	T_{amb}	−40	+ 125	°C

DEVELOPMENT DATA

DC CHARACTERISTICS

V_{DD} = 5 V (± 10%); V_{SS} = 0 V; T_{amb} = 0 to + 70 °C; all voltages with respect to V_{SS} unless otherwise specified.

parameter	conditions	symbol	min.	max.	unit
Supply voltage	T_{amb} = 0 to + 70 °C	V_{DD}	4,0	6,0	V
	T_{amb} = −40 to + 80 °C	V_{DD}	4,0	6,0	V
	T_{amb} = −40 to + 125 °C	V_{DD}	4,5	5,5	V
Supply current operating	(note 1)				
	f_{CLK} = 16 MHz; V_{DD} = 6,0 V	I_{DD}	–	40	mA
	f_{CLK} = 12 MHz; V_{DD} = 6,0 V	I_{DD}	–	35	mA
	f_{CLK} = 12 MHz; V_{DD} = 5,5 V	I_{DD}	–	30	mA
Idle mode	(note 2)				
	f_{CLK} = 16 MHz; V_{DD} = 5 V + 20%	I_{DD}	–	9	mA
	f_{CLK} = 12 MHz; V_{DD} = 5 V + 20%	I_{DD}	–	8	mA
	f_{CLK} = 12 MHz; V_{DD} = 5 V + 10%	I_{DD}	–	7	mA
Power-down current	2 V < V_{PD} < V_{DD} max. (note 3)				
	T_{amb} = −40 to + 85 °C	I_{PD}	–	50	μA
	T_{amb} = −40 to + 125 °C	I_{PD}	–	100	μA
Inputs					
LOW level input voltage (except $\overline{EA}$)		V_{IL}	−0,5	$0{,}2V_{DD}-0{,}1$	V
LOW level input voltage ($\overline{EA}$)		V_{IL1}	−0,5	$0{,}2V_{DD}-0{,}3$	V
HIGH level input voltage (except RST, XTAL1)		V_{IH}	$0{,}2V_{DD}+0{,}9$	$V_{DD}+0{,}5$	V
HIGH level input voltage (RST and XTAL1)		V_{IH1}	$0{,}7V_{DD}$	$V_{DD}+0{,}5$	V
HIGH level input voltage	(note 6)	V_{IH2}	$0{,}7V_{DD}$	6,0	V
Input current logic 0 (Ports 1, 2, 3 and 4)	V_I = 0,45 V	$-I_{IL}$	–	50	μA
Input current logic 1 to 0 transition (Ports 1, 2, 3 and 4)	V_I = 2,0 V	$-I_{TL}$	–	650	μA
Input leakage current (Port 0, Port 5, $\overline{EA}$, STADC, $\overline{EW}$)	0,45 V < V_I < V_{DD}	$\pm I_{LI1}$	–	10	μA

DC CHARACTERISTICS (continued)

parameter	conditions	symbol	min.	max.	unit
Outputs					
LOW level output voltage (Ports 1, 2, 3 and 4)	(note 8) I_{OL} = 1,6 mA	V_{OL}	–	0,45	V
LOW level output voltage (Port 0, ALE, $\overline{PSEN}$, $\overline{PWM0}$, $\overline{PWM1}$)	(note 8) I_{OL} = 3,2 mA	V_{OL1}	–	0,45	V
HIGH level output voltage (Ports 1, 2, 3 and 4)	V_{DD} = 5 V ± 10% $-I_{OH}$ = 60 μA $-I_{OH}$ = 25 μA $-I_{OH}$ = 10 μA	V_{OH} V_{OH} V_{OH}	2,4 $0{,}75V_{DD}$ $0{,}9V_{DD}$	– – –	V V V
HIGH level output voltage (Port 0 in external bus) mode, ALE, $\overline{PSEN}$, $\overline{PWM0}$, $\overline{PWM1}$)	(note 9) V_{DD} = 5 V ± 10% $-I_{OH}$ = 400 μA $-I_{OH}$ = 150 μA $-I_{OH}$ = 40 μA	V_{OH1} V_{OH1} V_{OH1}	2,4 $0{,}75V_{DD}$ $0{,}9V_{DD}$	– – –	V V V
HIGH level output voltage (RST)	$-I_{OH}$ = 400 μA $-I_{OH}$ = 120 μA	V_{OH2} V_{OH2}	2,4 $0{,}8V_{DD}$	– –	V V
RST pull-down resistor		R_{RST}	50	150	kΩ
Capacitance of I/O buffer	Test freq. = 1 MHz T_{amb} = 25 °C	$C_{I/O}$	–	10	pF

DEVELOPMENT DATA

DC CHARACTERISTICS

ΔV_{DD} = 5,0 V ± 10%; ΔV_{SS} = 0 V; V_{REF} = 5,0 V; T_{amb} = 0 to + 70 °C unless otherwise specified.

parameter	conditions	symbol	min.	max.	unit
Analogue inputs					
Analogue supply voltage	$\Delta V_{DD} = V_{DD} \pm 0,2$	ΔV_{DD}	4,0	6,0	V
Analogue supply current operating	Port 5 = 0 to ΔV_{DD}	ΔI_{DD}		1,2	mA
Idle mode	2 V < V_{APD} < V_{DD} max	ΔI_{ID}		50	μA
Power-down	T_{amb} = −40 °C to + 85 °C	ΔI_{PD}	–	50	μA
	T_{amb} = −40 °C to + 125 °C	ΔI_{PD}	–	100	μA
Analogue input voltage		ΔV_{IN}	ΔV_{SS}−0,2	ΔV_{DD}+0,2	V
Reference voltage		$\Delta V_{REF}-$	ΔV_{SS}−0,2	–	V
		$\Delta V_{REF}+$	–	ΔV_{DD}+0,2	V
Resistance between $\Delta V_{REF}+$ and $\Delta V_{REF}-$		R_{REF}	5	25	kΩ
Analogue input capacitance		C_{IA}	–	15	pF
Sampling time		t_{ADS}	–	$6t_{CY}$	μs
Conversion time (including sample time)		t_{ADC}	–	$24t_{CY}$	μs
Differential non-linearity	(notes 4 and 5)	DL_e	–	± 1	LSB
Integral non-linearity	(notes 4 and 6)	IL_e	–	± 1	LSB
Offset error	(note 4)	OS_e	–	± 1	mV
Gain error	(note 4)	G_e	–	0,4	%
Channel to channel matching		M_{ctc}	–	± 1	LSB
Crosstalk	0 to 100 kHz (note 7)	C_t	–	60	dB

Notes to the DC characteristics

1. The operating supply current is measured with all output pins disconnected; XTAL1 driven with $t_r = t_f$ = 10 ns; $V_{IL} = V_{SS}$ + 0,5 V; $V_{IH} = V_{DD}$ −0,5 V; XTAL2 not connected; $\overline{EA}$ = RST = Port 0 = $\overline{EW}$ = V_{DD}; STADC = V_{SS}.
2. The Idle mode supply current is measured with all output pins disconnected; XTAL1 driven with $t_r = t_f$ = 10 ns; $V_{IL} = V_{SS}$ + 0,5 V; $V_{IH} = V_{DD}$ −0,5 V; XTAL2 not connected; $\overline{EA}$ = Port 0 = $\overline{EW}$ = V_{DD}; RST = STADC = V_{SS}.
3. The power-down current is measured with all output pins disconnected; XTAL2 not connected; $\overline{EA}$ = Port 0 = $\overline{EW}$ = V_{DD}; RST = STADC = V_{SS}.
4. $\Delta V_{REF}+ = 5,12$ V; $\Delta V_{REF}- = 0$ V; $\Delta V_{DD} = 5,0$ V.
5. Differential non-linearity: maximum deviation of the actual code width from the average code width (see Fig. 18).
6. Integral non-linearity: maximum deviation between the edges of the steps of the transfer curve and the edges of the steps of the ideal step curve. The ideal step curve follows the line of least squares.

Notes to the DC characteristics (continued)

7. This should be considered when both analogue and digital signals are input simultaneously.
8. Capacitive loading on Port 0 and Port 2 may cause spurious noise pulses to be superimposed on the LOW level output voltage of ALE, Port 1 and Port 3. The noise is due to external bus capacitance discharging into the Port 0 and Port 2 pins when these pins make a 1-to-0 transition during bus operations. In the most adverse condition (capacitive loading $>$ 100 pF) the noise pulse on ALE line may exceed 0,8 V. In this event it may be required to qualify ALE with a Schmitt trigger, or use an address latch with a Schmitt trigger STROBE input.
9. Capacitive loading on Port 0 and Port 2 may cause the HIGH level output voltage on ALE and $\overline{PSEN}$ to momentarily fall below the 0,9 V_{DD} specification when the address bits are stabilizing.

DEVELOPMENT DATA

AC CHARACTERISTICS

V_{DD} = 5 V ± 20% (PCB83C562), V_{SS} = 0 V, $t_{CKmin} = 1/f_{max}$ (maximum operating frequency)

V_{DD} = 5 V ± 20% (PCF83C562), V_{SS} = 0 V, $t_{CKmin} = 1/f_{max}$ (maximum operating frequency)

V_{DD} = 5 V ± 10% (PCA83C562), V_{SS} = 0 V, $t_{CKmin} = 1/f_{max}$ (maximum operating frequency)

T_{amb} = 0 to + 70 °C, t_{CKmin} = 63 ns (PCB83C562 see note 1)

T_{amb} = −40 to + 85 °C, t_{CKmin} = 83 ns (PCF83C562 see note 1)

T_{amb} = −40 to + 125 °C, t_{CKmin} = 83 ns (PCA83C562 see note 1)

C_L = 100 pF for Port 0, ALE and $\overline{PSEN}$; C_L = 80 pF for all other outputs unless otherwise specified. (See waveforms Figs 19, 20 and 21).

parameter	symbol	16 MHz		12 MHz		variable clock		unit
		min.	max.	min.	max.	min.	max.	
Program memory								
ALE pulse duration	t_{LL}	85	–	127	–	$2t_{CK}-40$	–	ns
Address set-up time to ALE	t_{AL}	8	–	28	–	$t_{CK}-55$	–	ns
Address hold time after ALE	t_{LA}	28	–	48	–	$t_{CK}-35$	–	ns
Time from ALE to valid instruction input	t_{LIV}	–	150	–	233	–	$4t_{CK}-100$	ns
Time from ALE to control pulse $\overline{PSEN}$	t_{LC}	23	–	43	–	$t_{CK}-40$	–	ns
Control pulse duration $\overline{PSEN}$	t_{CC}	143	–	205	–	$3t_{CK}-45$	–	ns
Time from $\overline{PSEN}$ to valid instruction input	t_{CIV}	–	83	–	145	–	$3t_{CK}-105$	ns
Input instruction hold time after $\overline{PSEN}$	t_{CI}	0	–	0	–	0	–	ns
Input instruction float delay after $\overline{PSEN}$	t_{CIF}	–	38	–	59	–	$t_{CK}-25$	ns
Address to valid instruction input	t_{AIV}	–	208	–	312	–	$5t_{CK}-105$	ns
Address float delay after $\overline{PSEN}$	t_{AFC}	–	10	–	10	–	10	ns

AC CHARACTERISTICS (continued)

parameter	symbol	16 MHz		12 MHz		variable clock		unit
		min.	max.	min.	max.	min.	max.	
External data memory								
$\overline{RD}$ pulse duration	t_{RR}	275	–	400	–	$6t_{CK}-100$	–	ns
$\overline{WR}$ pulse duration	t_{WW}	275	–	400	–	$6t_{CK}-100$	–	ns
Address set up time to ALE	t_{AL}	28	–	48	–	$t_{CK}-40$	–	ns
Address hold time after ALE	t_{LA}	8	–	48	–	$t_{CK}-35$	–	ns
$\overline{RD}$ to valid data input	t_{RD}	–	148	–	252	–	$5t_{CK}-165$	ns
Data hold time after $\overline{RD}$	t_{DR}	0	–	0	–	0	–	ns
Data float delay after $\overline{RD}$	t_{DFR}	–	35	–	97	–	$2t_{CK}-70$	ns
Time from ALE to valid data input	t_{LD}	–	350	–	517	–	$8t_{CK}-150$	ns
Address to valid data input	t_{AD}	–	398	–	585	–	$9t_{CK}-165$	ns
Time from ALE to $\overline{RD}$ or $\overline{WR}$	t_{LW}	138	238	200	300	$3t_{CK}-50$	$3t_{CK}+50$	ns
Time from address to $\overline{RD}$ or $\overline{WR}$	t_{AW}	120	–	203	–	$4t_{CK}-130$	–	ns
Time from $\overline{RD}$ or $\overline{WR}$ HIGH to ALE HIGH	t_{WHLH}	23	103	43	123	$t_{CK}-40$	$t_{CK}+40$	ns
Data valid to $\overline{WR}$ transition	t_{DWX}	3	–	23	–	$t_{CK}-60$	–	ns
Data set-up time before $\overline{WR}$	t_{DW}	288	–	433	–	$7t_{CK}-150$	–	ns
Data hold time after $\overline{WR}$	t_{WD}	13	–	33	–	$t_{CK}-50$	–	ns
Address float delay after $\overline{RD}$	t_{AFR}	–	0	–	0	–	0	ns

Notes

1. The maximum operating frequency is limited to:
   16 MHz for the PCB83C562
   12 MHz for the PCF83C562 and PCA83C562
2. The minimum operating frequency is 1,2 MHz for all versions.

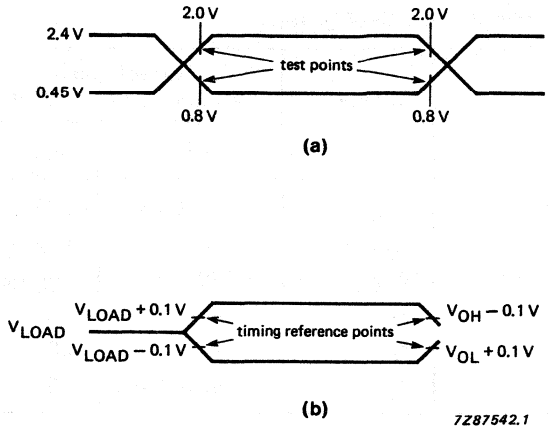


Fig. 17 AC testing input, output waveform (a) and float waveform (b).

AC testing inputs are driven at 2,4 V for a logic 1 and 0,45 V for a logic 0. Timing measurements are taken at 2,0 V for a logic 1 and 0,8 V for logic 0. The float state is defined as the point at which a Port 0 pin sinks 3,2 mA or sources 400 μA at the voltage test levels.

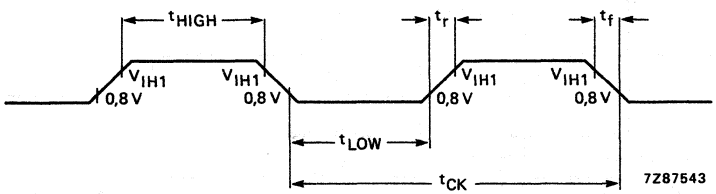


Fig. 18 External clock drive XTAL 1 (see Table 6).

AC CHARACTERISTICS (continued)

Table 5 External clock drive XTAL 1 (see Fig. 18).

parameter	symbol	variable clock f = 1,2 to 16 MHz (12 MHz *)		unit
		min.	max.	
oscillator clock period	t_{CK}	63 (83,3*)	8333,3	ns
HIGH time	t_{HIGH}	20	t_{CK}-t_{LOW}	ns
LOW time	t_{LOW}	20	t_{CL}-t_{HIGH}	ns
rise time	t_r	–	20	ns
fall time	t_f	–	20	ns
cycle time	t_{CY}**	0,75 (1,0*)	10	μs

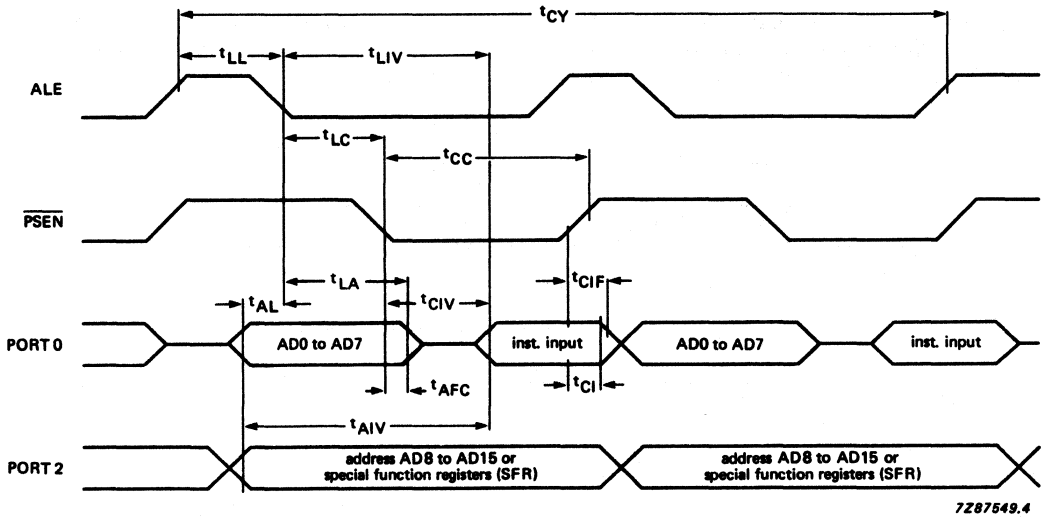


Fig. 19 Read from program memory.

* refers to PCF83C562 and PCA83C562

** $t_{CY} = 12\ t_{CK}$.

AC CHARACTERISTICS (continued)

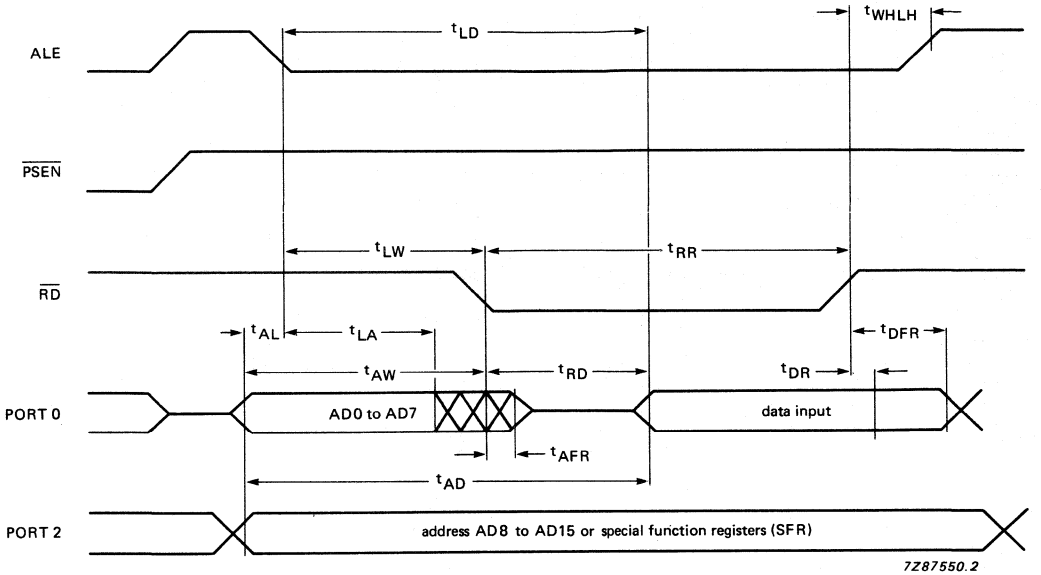


Fig. 20 Read from data memory.

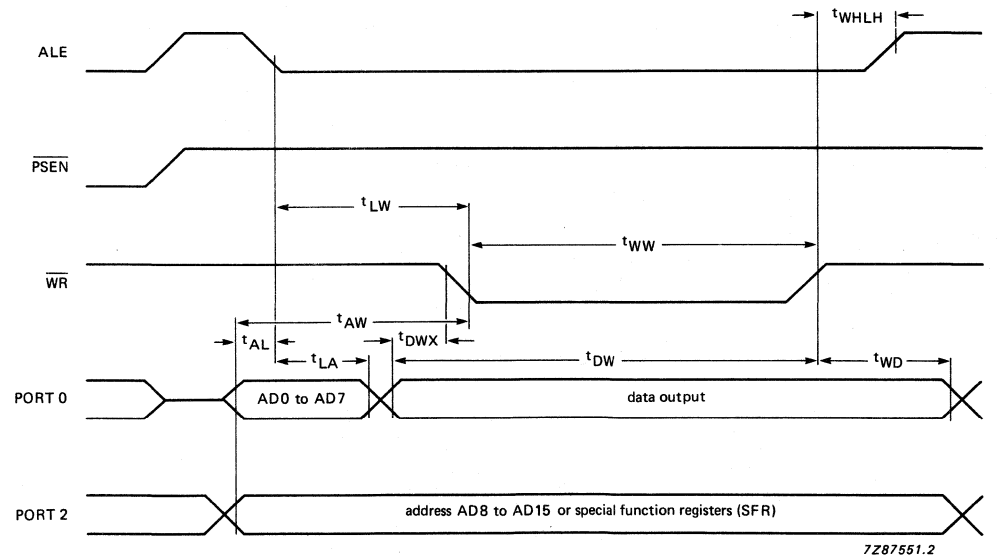


Fig. 21 Write to data memory.

DEVELOPMENT DATA

AC CHARACTERISTICS (continued)

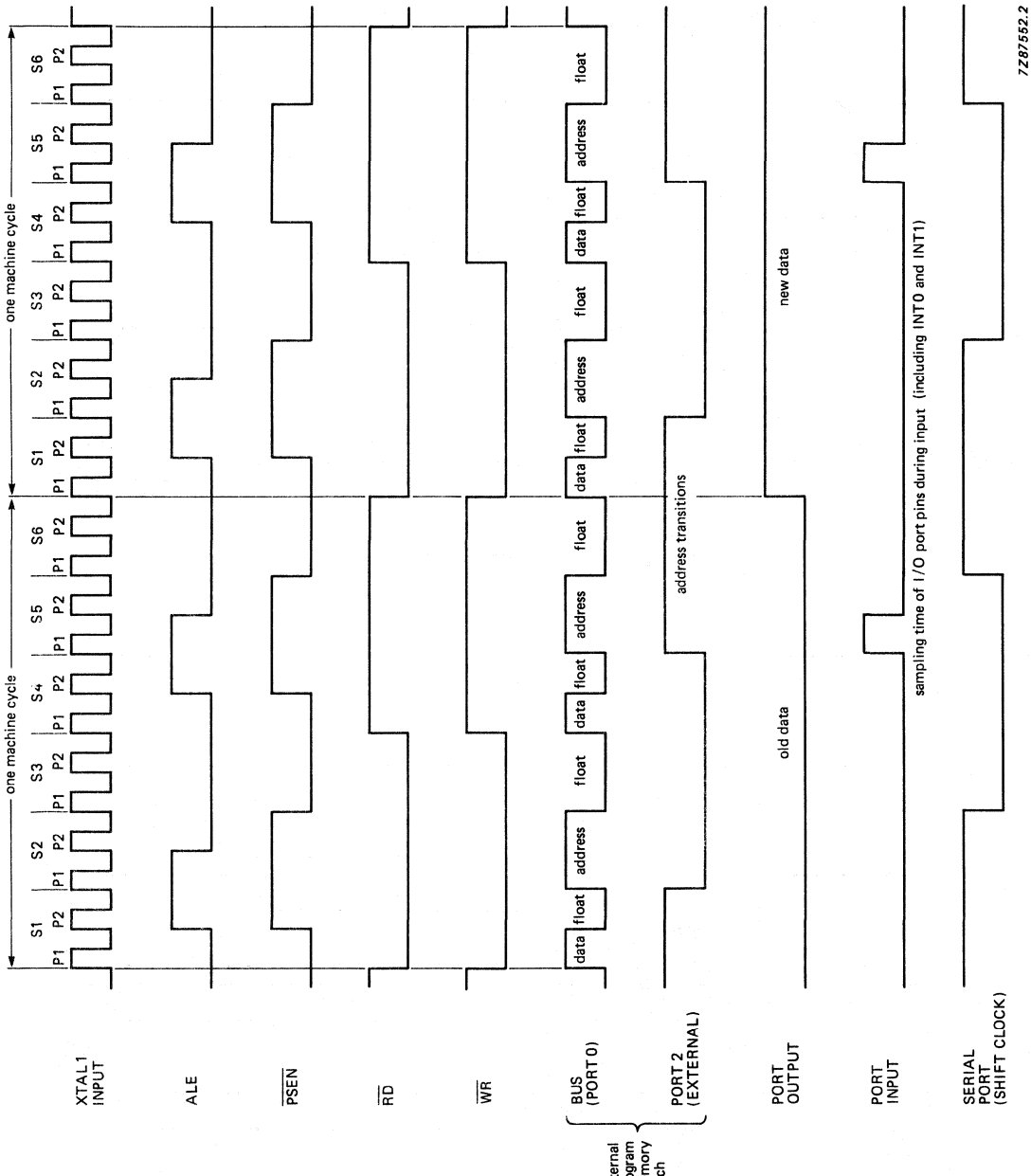


Fig. 22 Instruction cycle timing.

SINGLE-CHIP 8-BIT MICROCONTROLLER

GENERAL DESCRIPTION

The PCB83C652 single-chip 8-bit microcontroller is manufactured in an advanced CMOS process and is a derivative of the PCB80C51 microcontroller family. The PCB83C652 has the same instruction set as the PCB80C51. Two versions of the derivative exist although the generic term "PCB83C652" is used to refer to both family members:

- PCB83C652: 8 K bytes mask-programmable ROM, 256 bytes RAM
- PCB80C652: ROM-less version of the PCB83C652

This device provides architectural enhancements that make it applicable in a variety of applications in general control systems.

The PCB83C652 contains a non-volatile 8 K x 8 read-only program memory, a volatile 256 x 8 read/write data memory, four 8-bit I/O ports, two 16-bit timer/event counters (identical to the timers of the 80C51), a multi-source, two-priority-level, nested interrupt structure, an I^2C interface, UART and on-chip oscillator and timing circuits. For systems that require extra capability, the PCB83C652 can be expanded using standard TTL compatible memories and logic.

The device also functions as an arithmetic processor having facilities for both binary and BCD arithmetic plus bit-handling capabilities. The instruction set consists of over 100 instructions: 49 one-byte, 45 two-byte and 17 three-byte. With a 12 MHz crystal, 58% of the instructions are executed in 1 μs and 40% in 2 μs. Multiply and divide instructions require 4 μs.

Features

- 80C51 central processing unit
- 8 K x 8 ROM, expandable externally to 64 K bytes
- 256 x 8 RAM, expandable externally to 64 K bytes
- Two standard 16-bit timer/counters
- Four 8-bit I/O ports
- I^2C-bus serial I/O port with byte orientated master and slave functions
- Full-duplex UART facilities
  Three temperature ranges available
  0 to + 70 °C; PCB83C652 versions
  −40 to + 85 °C; PCF83C652 versions
  −40 to + 125 °C; PCA83C652 versions ←
- Extended frequency range: 1.2 MHz to 12 MHz ←

PACKAGE OUTLINES

PCA/PCB/PCF83C652P; PCA/PCB/PCF80C652P: 40-lead DIL; plastic (SOT129).
PCA/PCB/PCF83C652WP; PCA/PCB/PCF80C652WP: 44-lead plastic leaded-chip-carrier ←
(PLCC) (SOT187 pedestal or SOT187AA pocket versions, these are interchangeable). ←
PCA/PCB/PCF83C652H; PCA/PCB/PCF80C652H: 44-lead quad flat-pack (QFP). This is in preparation.

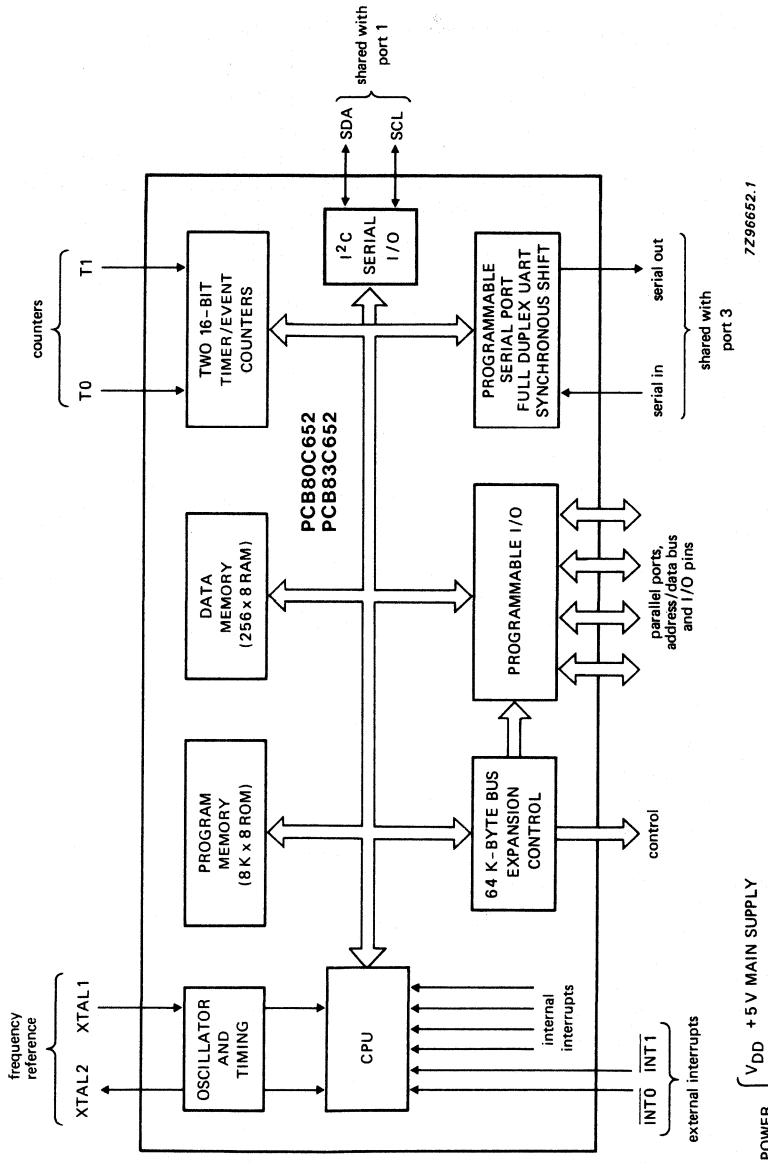


Fig. 1 Block diagram.

DEVELOPMENT DATA

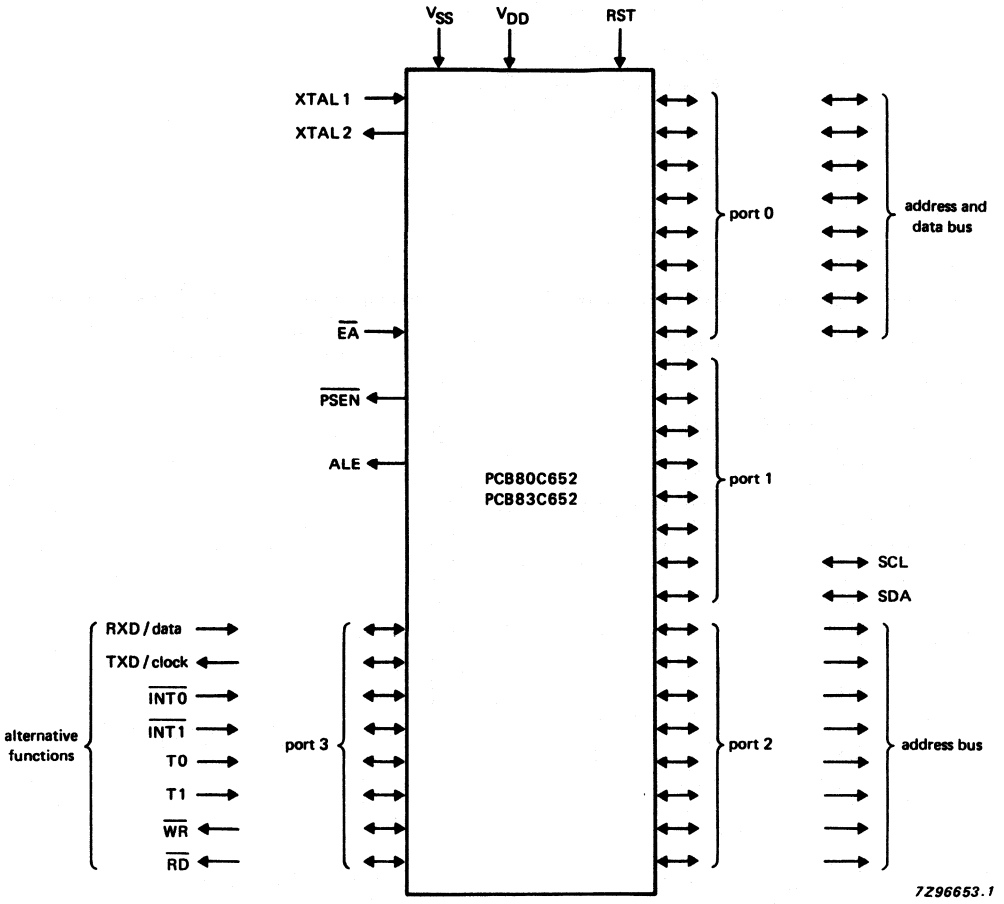


Fig. 2 Functional diagram.

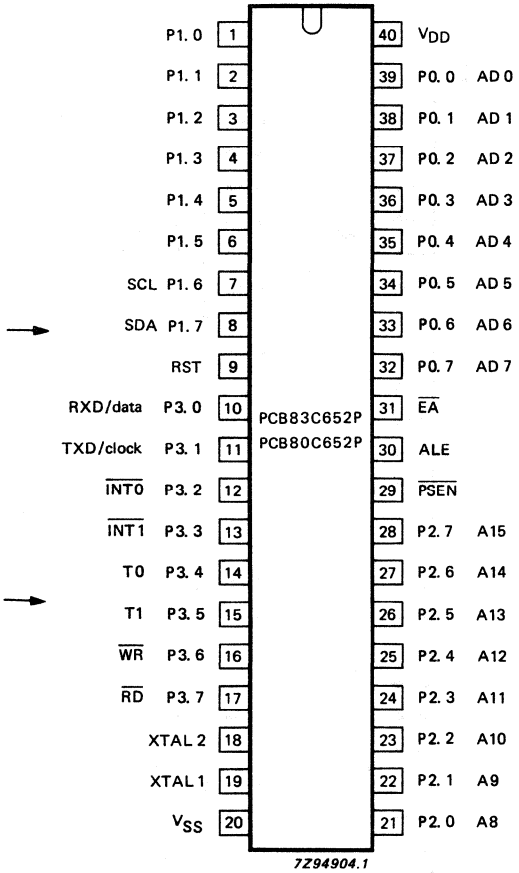


Fig. 3a Pinning diagram for 40-lead DIL package (SOT129).

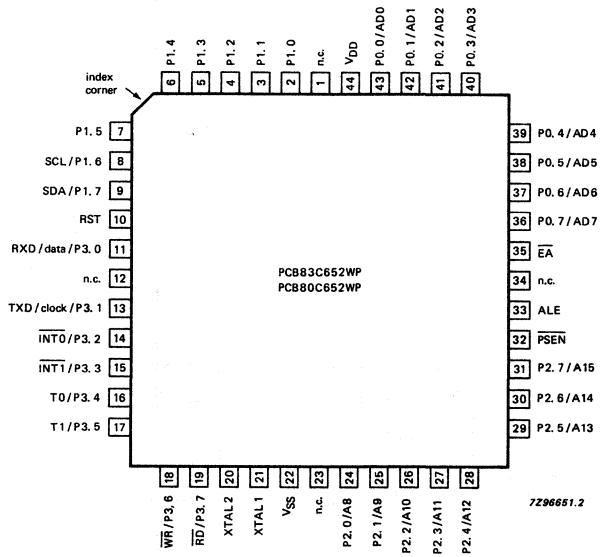


Fig. 3b Pinning diagram for 44-lead PLCC package (SOT187).

PINNING (PCB83C652P, PCB80C652P).

1-8	P1.0-P1.7	**Port 1:** 8-bit quasi-bidirectional I/O port. Port 1 can sink/source one TTL (= 4 LSTTL) input. It can drive CMOS inputs without external pull-ups, except P1.6 and P1.7 which have open drain outputs. Port pin — Alternative function P1.6 — SCL: I^2C-bus serial port clock line P1.7 — SDA: I^2C-bus serial port data line
9	RST	**RESET:** a high level on this pin for two machine cycles while the oscillator is running resets the device. An internal pull-down resistor permits Power-On reset using only a capacitor connected to V_{DD}.

DEVELOPMENT DATA

10-17	P3.0-P3.7	**Port 3:** 8-bit quasi-bidirectional I/O port with internal pull-ups. It also serves the following alternative functions:

Port pin	Alternative function
P3.0	RXD/data: serial port receiver data input (asynchronous) or data input/output (synchronous)
P3.1	TXD/clock: serial port transmitter data output (asynchronous) or clock output (synchronous)
P3.2	$\overline{INT0}$: external interrupt 0 or gate control input for timer/event counter 0
P3.3	$\overline{INT1}$: external interrupt 1 or gate control input for timer/event counter 1
P3.4	T0: external input for timer/event counter 0
P3.5	T1: external input for timer/event counter 1
P3.6	$\overline{WR}$: external data memory write strobe
P3.7	$\overline{RD}$: external data memory read strobe

The generation or use of a Port 3 pin as an alternative function is carried out automatically by the PCB83C652 provided the associated Special Function Register bit is set high. Port 3 can sink/source one TTL (= 4 LSTTL) input. It can drive CMOS inputs without external pull-ups.

18	XTAL 2	**Crystal input 2:** output of the inverting amplifier that forms the oscillator. Left open-circuit when an external oscillator clock is used (see Figs 10 and 11).
19	XTAL 1	**Crystal input 1:** input to the inverting amplifier that forms the oscillator, and input to the internal clock generator. Receives the external oscillator clock signal when an external oscillator is used (see Figs 10 and 11).
20	V_{SS}	**Ground:** circuit ground potential.
21-28	P2.0-P2.7	**Port 2:** 8-bit quasi-bidirectional I/O port with internal pull-ups. During access to external memories (RAM/ROM) that use 16-bit addresses (MOVX @DPTR) Port 2 emits the high order address byte. When external RAM is accessed with an 8-bit address (MOVX @Ri) Port 2 emits the contents of the P2 special function register. Port 2 can sink/source one TTL (= 4 LSTTL) input. It can drive CMOS inputs without external pull-ups.
29	$\overline{PSEN}$	**Program Store Enable output:** read strobe to the external program memory via Port 0 and 2. It is activated twice each machine cycle during fetches from external program memory. When executing out of external program memory two activations of $\overline{PSEN}$ are skipped during each access to external data memory. $\overline{PSEN}$ is not activated (remains HIGH) during no fetches from external program memory. $\overline{PSEN}$ can sink/source 8 LSTTL inputs. It can drive CMOS inputs without an external pull-up.
30	ALE	**Address Latch Enable output:** latches the low byte of the address during accesses to external memory in normal operation. It is activated every six oscillator periods except during an external data memory access. ALE can sink/source 8 LSTTL inputs. It can drive CMOS inputs without an external pull-up.

PINNING (continued)

31	$\overline{EA}$	**External Access input:** When $\overline{EA}$ is held at a TTL high level the CPU executes out of the internal program ROM provided the program counter is less than 8192. When $\overline{EA}$ is held at a TTL low level, the CPU executes out of external program memory via Port 0 and Port 2. $\overline{EA}$ is not allowed to float.
32-39	P0.7-P0.0	**Port 0:** 8-bit open drain bidirectional I/O port. It is also the multiplexed low-order address and data bus during accesses to external memory (during these accesses it activates internal pull-ups). Port 0 can sink/source eight LSTTL inputs.
40	V_{DD}	**Power supply:** + 5 V power supply pin during normal operation, Idle mode and Power-down mode.

To avoid a 'latch-up' effect at power-on, the voltage on any pin at any time must not be higher or lower than V_{DD} + 0,5 V or V_{SS} − 0,5 V respectively.

FUNCTIONAL DESCRIPTION

General

The PCB83C652 is a stand-alone high-performance microcontroller designed for use in real-time applications such as instrumentation and industrial control.

The device provides, in addition to the 80C51 standard functions, a serial I^2C-bus interface. As well as the parallel bus, functions may also be expanded using the I^2C-bus utilizing the complete line of the I^2C clips family.

The PCB83C652 is a control-oriented CPU with on-chip program and data memory. It can be extended with external program memory up to 64 K bytes. It can also access up to 64 K bytes of external data memory. For systems requiring extra capability, the PCB83C652 can be expanded using standard memories and peripherals.

The PCB83C652 has two software selectable modes of reduced activity for further power reduction – Idle and Power-down. The Idle mode freezes the CPU while allowing the RAM, timers, serial ports and interrupt system to continue functioning. The Power-down mode saves the RAM contents but freezes the oscillator causing all other chip functions to be inoperative.

Memory organization

The central processing unit (CPU) manipulates operands in three memory spaces; these are the 64 K-byte external data memory, 256-byte internal data memory and the 64 K-byte internal and external program memory. The internal data memory address space is sub-divided into the 256-byte internal data RAM and 128-byte Special Function Register (SFR) address spaces, as shown in Fig. 4. Figure 5 shows the Special Function Registers memory map. Internal RAM locations 0-127 are directly and indirectly addressable. Internal RAM locations 128-255 are only indirectly addressable. The special function register locations 128-255 are only directly addressable.

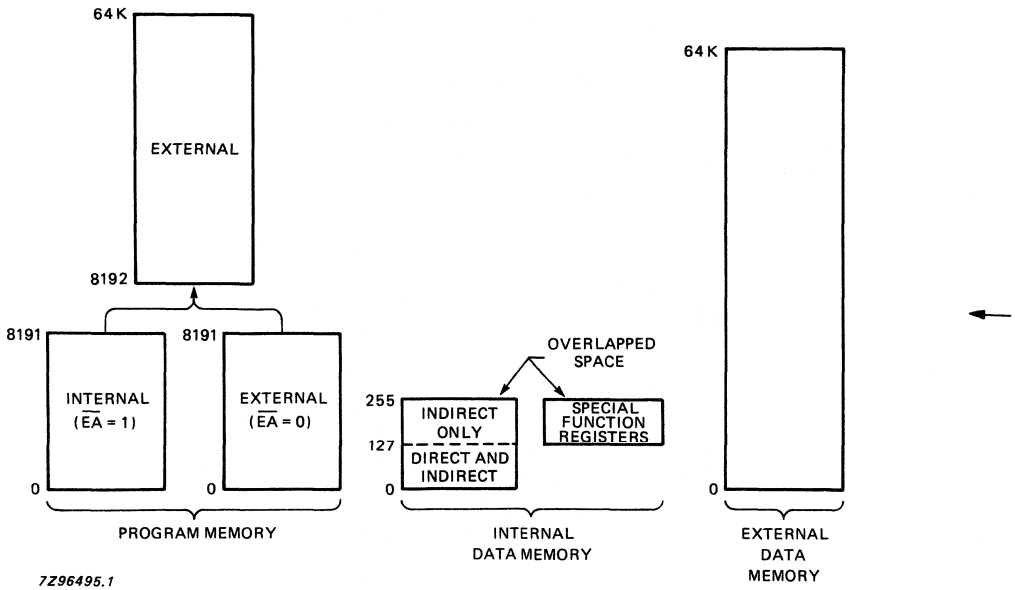
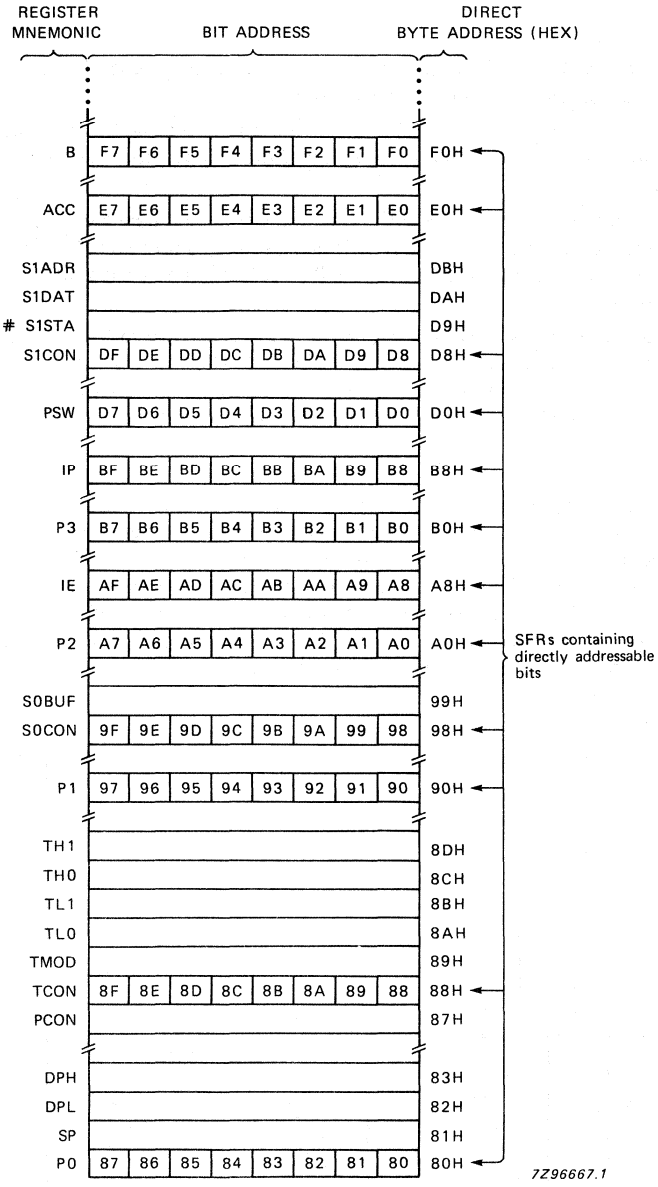


Fig. 4 Memory map.



S1STA is a read-only register.

Fig. 5 Special function registers memory map.

FUNCTIONAL DESCRIPTION (continued)

The internal data RAM contains four register banks (each with eight registers), 128 addressable bits, a scratchpad area and the stack. The stack depth is limited by the available internal data RAM and its location is determined by the 8-bit stack pointer. All registers except the program counter and the four 8-register banks reside in the special function register address space. These memory mapped registers include arithmetic registers, pointers, I/O ports, interrupt system registers, timers and serial port registers. There are 128 addressable bit locations in the SFR address space.

The PCB83C652 contains 256 bytes of internal data RAM and 25 special function registers. It provides a non-paged program memory address space to accommodate relocatable code. Conditional branches are performed relative to the program counter. The register-indirect jump permits branching relative to a 16-bit base register with an offset provided by an 8-bit index register. 16-bit jumps and calls permit branching to any location in the contiguous 64 K program memory address space.

Addressing

The PCB83C652 has five methods for addressing source operands:

- Register
- Direct
- Register-Indirect
- Immediate
- Base-Register plus Index-Register-Indirect

The first three methods can be used for addressing destination operands. Most instructions have a "destination/source" field that specifies the data type, addressing methods and operands involved. For operations other than MOVs, the destination operand is also a source operand.

Access to memory addressing is as follows:

- Registers in one of the four 8-register banks through Register, Direct, or Register-Indirect.
- 256 bytes of internal data RAM through Direct or Register-Indirect. Bytes 0-127 may be addressed directly/indirectly. Bytes 128-255 share their address locations with the SFR registers and so may only be addressed indirectly as data RAM.
- Special function registers through Direct at address locations 128-255.
- External data memory through Register-Indirect.
- Program memory look-up tables through Base-Register plus Index-Register-Indirect.

The PCB83C652 is classified as an 8-bit device since the internal ROM, RAM, Special Function Registers (SFR), Arithmetic Logic Unit (ALU) and external data bus are each 8-bits wide. It performs operations on bit, nibble, byte and double-byte data types.

Facilities are available for byte transfer, logic and integer arithmetic operations. Data transfer, logic and conditional branch operations can be performed directly on Boolean variables to provide excellent bit handling.

Instruction set

The PCB83C652 uses the powerful instruction set of the PCB80C51. Additional special function registers are incorporated to control the on-chip peripherals. The instruction set consists of 49 single-byte, 45 two-byte and 17 three-byte instructions. When using a 12 MHz oscillator, 64 instructions execute in 1 μs and 45 instructions execute in 2 μs. Multiply and divide instructions execute in 4 μs.

FUNCTIONAL DESCRIPTION (continued)

I/O facilities

The PCB83C652 has four 8-bit ports. Ports 0-3 are the same as in the 80C51, with the exception of the additional functions of Port 1. Port lines P1.7 and P1.6 may be selected as the SDA and SCL lines of serial port SIO1 (I^2C). Because the I^2C-bus may be active while the device is disconnected from V_{DD}, these pins are provided with open drain drivers.

N.B. Therefore pins P1.7 and P1.6 do not have pull-up devices when used as ports.

Ports 0, 1, 2 and 3 perform the following alternative functions:

- Port 0: provides the multiplexed low-order address and data bus used for expanding the PCB83C652 with standard memories and peripherals.
- Port 1: Port 1 is partly used for the I^2C-bus functions;
  - SCL and SDA for the I^2C interface, P1.6 and P1.7 respectively.

Bits whose alternate function is not used may be used as normal bidirectional I/O pins.

- Port 2: provides the high-order address bus when expanding the PCB83C652 with external program memory and/or external data memory.
- Port 3: pins can be configured individually to provide:
  - external interrupt request inputs
  - counter inputs
  - serial port receiver input and transmitter output
  - control signals to READ and WRITE external data memory

The generation or use of a Port 3 pin as an alternative function is carried out automatically by the PCB83C652 provided the associated Special Function Register bit is set high.

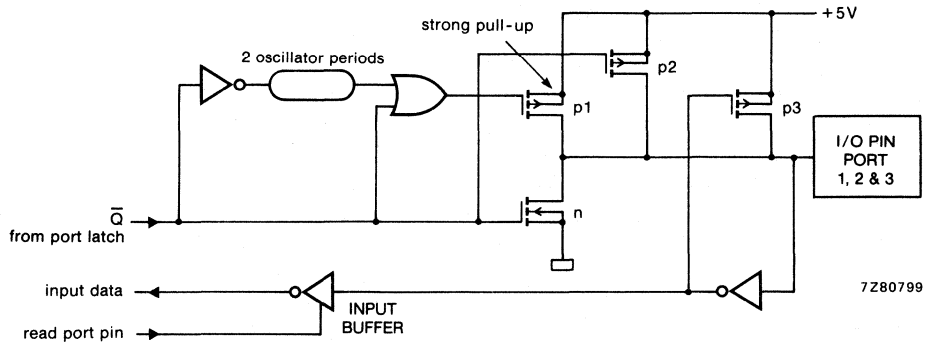


Fig. 6 I/O buffers in the PCB83C652 (Ports 2, 3, and P1.0 to P1.5).

Timer/event counters

The PCB83C652 contains two 16-bit timer/event counters: Timer 0 and Timer 1. Timer 0 and Timer 1 may be programmed to carry out the following functions:

- Measure time intervals and pulse durations
- Count events
- Generate interrupt requests.

Timer 0 and Timer 1 can be programmed independently to operate in three modes:

- Mode 0: 8-bit timer or 8-bit counter each with divide-by-32 prescaler
- Mode 1: 16-bit time-interval or event counter
- Mode 2: 8-bit time-interval or event counter with automatic reload upon overflow.

Timer 0 can be programmed to operate in an additional mode as follows:

- Mode 3: one 8-bit time-interval or event counter and one 8-bit time-interval counter.

When Timer 0 is in Mode 3, Timer 1 can be programmed to operate in Modes 0, 1 or 2 but cannot set an interrupt request flag or generate an interrupt. However the overflow from Timer 1 can be used to pulse the serial port transmission-rate generator.

The frequency handling range of these counters with a 12 MHz crystal is as follows:

- In the timer function, the timer is incremented at a frequency of 1 MHz – a division by 12 of the oscillator frequency
- 0 Hz to an upper limit of 0,5 MHz when programmed for external inputs.

Both internal and external inputs can be gated to the counter by a second external source for directly measuring pulse durations.

The counters are started and stopped under software control. Each one sets its interrupt request flag when it overflows from all logic 1s to all logic 0s (or automatic reload value), with the exception of mode 3 as previously described.

Serial I/O (see Fig. 7)

The PCB83C652 is equipped with two independent serial ports. SIO0 is the full duplex UART port and is identical to the serial port of the PCB80C51.

Serial port SIO1 supports the I^2C-bus, the function of which is controlled by the S1CON register. S1STA is the status register whose contents may also be used as a vector to various service routines. S1DAT is the data shift register and S1ADR the slave address register. The least significant bit of S1ADR enables/disables general call address recognition.

FUNCTIONAL DESCRIPTION (continued)

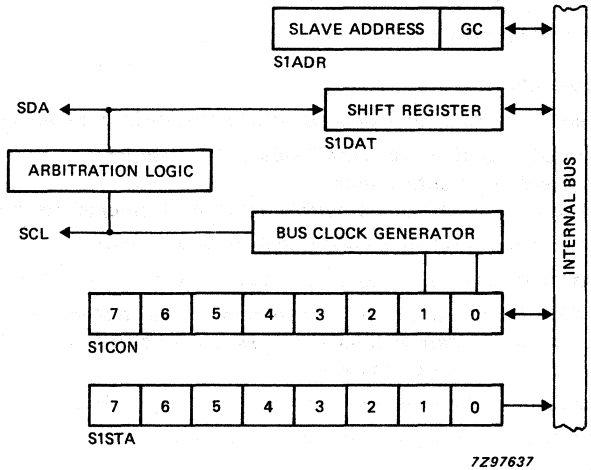


Fig. 7 Block diagram of I^2C serial I/O.

The I^2C serial I/O has complete autonomy in byte handling and operates in 4 modes:

1. Master transmitter
2. Master receiver
3. Slave transmitter
4. Slave receiver

Slave address recognition is performed by on-chip hardware.

The I^2C-bus consists of two lines: a data line (SDA) and a clock line (SCL). These lines also function as the I/O port lines on P1.7 and P1.6. The system is unique because data transport, clock generation, address recognition and bus control arbitration are all controlled by hardware.

Serial control register S1CON

	7	6	5	4	3	2	1	0
S1CON(D8H)	X	ENS1	STA	STO	SI	AA	CR1	CR0

Bits CR1 and CR0 determine the clock frequency that is generated in the master mode of operation. Table 1 displays the clock rate when using a 12 MHz crystal.

Table 1 Clock rate when using a 12 MHz crystal

CR1 / CR0	bit frequency	f_{osc} divided by
0 0	12,5 kHz	960
0 1	100 kHz	120
1 0	200 kHz	60 ($f_{osc} < 6$ MHz meeting I^2C)
1 1	62,5–0,5 kHz	96 x (256–reload value Timer 1) (reload value range: 0-254 in mode 2)

AA

Assert acknowledge bit. When this bit is set, an acknowledge is returned after any one of the following conditions:

- Own slave address is received
- General call address is received (S1ADR.0 = logic 1)
- A data byte is received, while the device is programmed to be a master receiver
- A data byte is received, while the device is a selected slave receiver.

When this bit is reset, no acknowledge is returned. Consequently, no interrupt is requested when the own address or general call address is received.

SI

SIO1 interrupt flag. This flag is set, and an interrupt request is generated, after any of the following events occur:

- A START condition is generated in MST mode
- The own slave address has been received during AA = logic 1
- The general call address has been received while S1ADR.0 and AA = logic 1
- A data byte has been received or transmitted in MST mode (even if arbitration is lost)
- A data byte has been received or transmitted as selected slave
- A STOP or START condition is received as selected slave receiver or transmitter.

STO

STOP flag. When in master mode,and this bit is set a STOP condition is generated. A STOP condition detected on the I^2C-bus clears this bit. This bit may also be set in slave mode in order to recover from an error condition. Then no STOP condition is generated to the I^2C-bus, but the hardware releases the SDA and SCL lines and switches to the not selected slave receiver mode. The STOP flag is cleared by the hardware.

STA

START flag. When this bit is set, the hardware checks the I^2C-bus and generates a START condition if the bus is free. If the device is already programmed as either master/transmitter or master/receiver, it will generate a repeated START condition.

ENS1

0 = Serial I/O Disabled and reset. Output ports P1.6 and P1.7 function with open drain

1 = Serial I/O Enabled. Output ports P1.6 and P1.7 must be set to logic 1.

FUNCTIONAL DESCRIPTION (continued)

Serial status register S1STA (S1STA is a read-only register)

	7	6	5	4	3	2	1	0
S1STA(D9H)	SC4	SC3	SC2	SC1	SC0	0	0	0

S1STA.3 - S1STA.7 hold a status code. S1STA.0 - S1STA.2 are held LOW. The contents of the status register may be used as a vector to a service routine. This optimizes the response time of the software and consequently that of the I^2C-bus.

Abbreviations used:

SLA : 7-bit slave address
R : Read bit
W : Write bit
ACK : Acknowledgement (acknowledge bit = logic 0)
$\overline{\text{ACK}}$: Not acknowledgement (acknowledge bit = logic 1)
DATA : 8-bit data byte to or from I^2C-bus
MST : Master
SLV : Slave
TRX : Transmitter
REC : Receiver

The following is a list of the status codes:

MST/TRX mode

S1STA value

08H - A START condition has been transmitted
10H - A repeated START condition has been transmitted
18H - SLA and W have been transmitted, ACK has been received
20H - SLA and W have been transmitted, $\overline{\text{ACK}}$ received
28H - DATA of S1DAT has been transmitted, ACK received
30H - DATA of S1DAT has been transmitted, $\overline{\text{ACK}}$ received
38H - Arbitration lost in SLA, R/W or DATA

MST/REC mode

S1STA value

38H - Arbitration lost while returning $\overline{\text{ACK}}$
40H - SLA and R have been transmitted, ACK received
48H - SLA and R have been transmitted, $\overline{\text{ACK}}$ received
50H - DATA has been received, ACK returned
58H - DATA has been received, $\overline{\text{ACK}}$ returned

SLV/REC mode

S1STA value

- 60H - Own SLA and W have been received, ACK returned
- 68H - Arbitration lost in SLA, R/W as MST. Own SLA and W have been received, ACK returned
- 70H - General CALL has been received, ACK returned
- 78H - Arbitration lost in SLA, R/W as MST. General call has been received
- 80H - Previously addressed with own SLA. DATA byte received, ACK returned
- 88H - Previously addressed with own SLA. DATA byte received, $\overline{\text{ACK}}$ returned
- 90H - Previously addressed with general call. DATA byte has been received, ACK has been returned
- 98H - Previously addressed with general call. DATA byte has been received, $\overline{\text{ACK}}$ has been returned
- A0H - A STOP condition or repeated START condition has been received while still addressed as SLV/REC or SLV/TRX.

SLV/TRX mode

S1STA value

- A8H - Own SLA and R have been received, ACK returned
- B0H - Arbitration lost in SLA, R/W as MST. Own SLA and R have been received, ACK returned
- B8H - DATA byte has been transmitted, ACK received
- C0H - DATA byte has been transmitted, $\overline{\text{ACK}}$ received
- C8H - Last DATA byte has been transmitted (AA = logic 0), ACK received.

Miscellaneous

S1STA value

- 00H - Bus error during MST mode or selected SLV mode, due to an erroneous START or STOP condition

The data shift register S1DAT

S1DAT(DAH)	7	6	5	4	3	2	1	0

This register contains the serial data to be transmitted or data which has just been received. Bit 7 is transmitted or received first; i.e. data is shifted from right to left.

Address register S1ADR

S1ADR (DBH)	7	6	5	4	3	2	1	0

S1ADR.0, GC : 0 = general call address is not recognized
1 = general call address is recognized

S1ADR.7-1 : own slave address

This 8-bit register may be loaded with the 7-bit slave address to which the controller will respond when programmed as a slave receiver/transmitter. The LSB (GC) is used to determine whether the general call address is recognized.

FUNCTIONAL DESCRIPTION (continued)

Idle and Power-down operation (see Fig. 8)

Idle mode operation permits the interrupt, serial ports and timer blocks to continue to function while the CPU is halted.

The following functions remain active during Idle mode. These functions may generate an interrupt or reset and thus end the Idle mode:

Timer 0, Timer 1
SIO0, SIO1
External interrupt

The Power-down operation freezes the oscillator. The Power-down mode can only be activated by setting the PD bit in the PCON register.

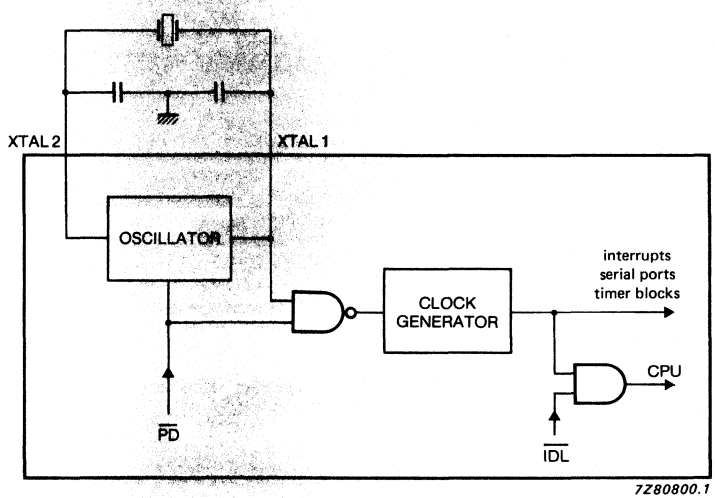


Fig. 8 Internal Idle and Power-down clock configuration.

Power control register

These special modes are activated by software via the Special Function Register PCON. Its hardware address is 87H. PCON is not bit addressable.

	7	6	5	4	3	2	1	0
PCON(87H)	SMOD	–	–	–	GF1	GF0	PD	IDL

Bit	Symbol	Function
PCON.7	SMOD	Double Baud rate bit. When set to logic 1 the baud rate is doubled when the serial port SIO0 is being used in modes 1, 2 or 3
PCON.6	–	(reserved)
PCON.5	–	(reserved)
PCON.4	–	(reserved)
PCON.3	GF1	General-purpose flag bit
PCON.2	GF0	General-purpose flag bit
PCON.1	PD	Power-down bit. Setting this bit activates Power-down mode
PCON.0	IDL	Idle mode bit. Setting this bit activates the Idle mode.

If logic 1s are written to PD and IDL at the same time, PD takes precedence. The reset value of PCON is 0XXX0000.

Idle mode

The instruction that sets PCON.0 is the last instruction executed in the normal operating mode before Idle mode is activated. Once in the Idle mode, the CPU status is preserved in its entirety: the Stack Pointer, Program Counter, Program Status Word, Accumulator, RAM and all other registers maintain their data during Idle mode. The status of the external pins during Idle mode is shown in Table 2.

There are two ways to terminate the Idle mode:

Activation of any enabled interrupt will cause PCON.0 to be cleared by hardware terminating Idle mode. The interrupt is serviced, and following return from interrupt instruction RETI, the next instruction to be executed will be the one which follows the instruction that wrote a logic 1 to PCON.0.

The flag bits GF0 and GF1 may be used to determine whether the interrupt was received during normal execution or during the Idle mode. For example, the instruction that writes to PCON.0 can also set or clear one or both flag bits. When Idle mode is terminated by an interrupt, the service routine can examine the status of the flags bits.

The second way of terminating the Idle mode is with an external hardware reset. Since the oscillator is still running, the hardware reset is required to be active for two machine cycles (24 oscillator periods) to complete the reset operation.

Power-down mode

The instruction that sets PCON.1 is the last executed prior to going into the Power-down mode. Once in Power-down mode, the oscillator is stopped. Only the contents of the on-chip RAM are preserved. The Special Function Registers are not saved. A hardware reset is the only way of exiting the Power-down mode.

In the Power-down mode, V_{DD} may be reduced to minimize circuit power consumption. The voltage must not be reduced until the Power-down mode is entered, but must be restored before the hardware reset is applied which will free the oscillator. Reset should not be released until the oscillator has restarted and stabilized.

The status of the external pins during Power-down mode is shown in Table 2. If the Power-down mode is activated while in external program memory, the port data that is held in the Special Function Register P2 is restored to Port 2. If the data is a logic 1, the port pin is held HIGH during the Power-down mode by the strong pull-up transistor p1 (see Fig. 6).

FUNCTIONAL DESCRIPTION (continued)

Table 2 Status of the external pins during Idle and Power-down modes.

mode	memory	ALE	$\overline{PSEN}$	Port 0	Port 1	Port 2	Port 3
Idle	internal	1	1	port data	port data	port data	port data
Idle	external	1	1	floating	port data	address	port data
Power-down	internal	0	0	port data	port data	port data	port data
Power-down	external	0	0	floating	port data	port data	port data

Note: Ports 1.7 and 1.6 if selected, function as SDA and SCL respectively in the Idle mode.

Interrupt system (see Fig. 9)

External events and the real-time-driven on-chip peripherals require service by the CPU asynchronous to the execution of any particular section of code. To tie the asynchronous activities of these functions to normal program execution a multiple-source, two-priority-level, nested interrupt system is provided. Interrupt response latency is from 3 μs to 8 μs when using a 12 MHz crystal. The PCB83C652 acknowledges interrupt requests from six sources as follows:

- $\overline{INT0}$ and $\overline{INT1}$: two external interrupts
- Timer 0 and Timer 1: two internal counters
- I^2C serial I/O interrupt
- UART serial I/O port interrupt

Each interrupt vectors to a separate location in program memory for its service program. Each source can be individually enabled or disabled by a corresponding bit in the IE register, moreover each interrupt may be programmed to a high or low priority level using a corresponding bit in the IP register. Also all enabled sources can be globally disabled or enabled. Both external interrupts can be programmed to be level-activated or transition-activated; an active LOW level allows "wire-ORing" of several interrupt sources to the input pin.

DEVELOPMENT DATA

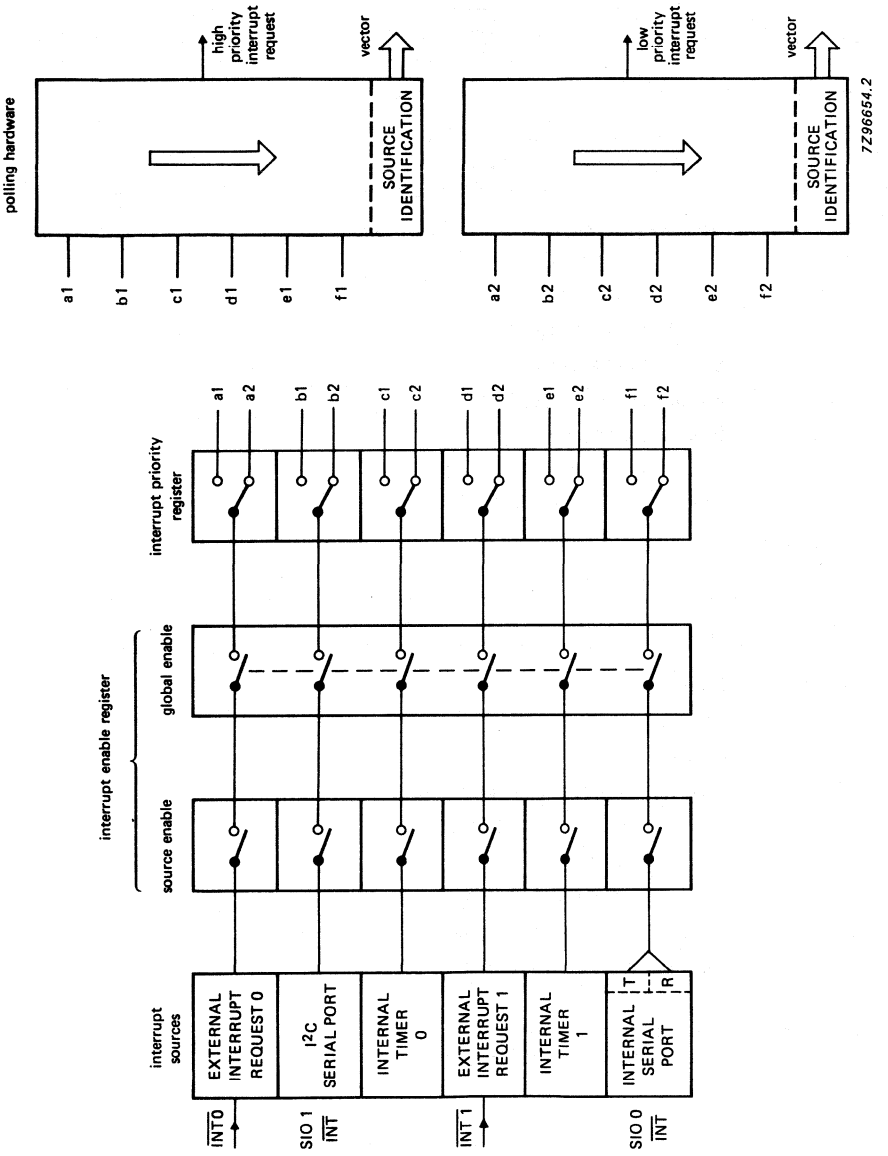


Fig. 9 Interrupt system.

FUNCTIONAL DESCRIPTION (continued)

Interrupt enable register

	7	6	5	4	3	2	1	0
IE(A8H)	EA	–	ES1	ES0	ET1	EX1	ET0	EX0

Bit	Symbol	Function
IE.7	EA	General enable/disable control 0 = No interrupt is enabled 1 = Any individually enabled interrupt will be accepted
IE.6	–	Unused
IE.5	ES1	Enable SIO1 (I^2C) interrupt
IE.4	ES0	Enable SIO0 (UART) interrupt
IE.3	ET1	Enable Timer 1 interrupt
IE.2	EX1	Enable External 1 interrupt
IE.1	ET0	Enable Timer 0 interrupt
IE.0	EX0	Enable External 0 interrupt
		where 0 = interrupt disabled and 1 = interrupt enabled

Interrupt priority register

	7	6	5	4	3	2	1	0
IP(B8H)	–	–	PS1	PS0	PT1	PX1	PT0	PX0

Bit	Symbol	Function
IP.7	–	Unused
IP.6	–	Unused
IP.5	PS1	SIO1 (I^2C) interrupt priority level
IP.4	PS0	SIO0 (UART) interrupt priority level
IP.3	PT1	Timer 1 interrupt priority level
IP.2	PX1	External interrupt 1 priority level
IP.1	PT0	Timer 0 interrupt priority level
IP.0	PX0	External interrupt 0 priority level

Interrupt priority levels are as follows: 0 = low priority
1 = high priority

Table 3 shows the interrupt vectors. The vector indicates the ROM location where the appropriate interrupt service routine starts.

Table 3 Interrupt vectors

source		vector
External 0	X0	0003H
Timer 0 overflow	T0	000BH
External 1	X1	0013H
Timer 1 overflow	T1	001BH
Serial I/O 0 (UART)	S0	0023H
Serial I/O 1 (I^2C)	S1	002BH

Interrupt priority

Each interrupt source can be either high priority or low priority. If both priorities are requested simultaneously, the processor will branch to the high priority vector. If there are simultaneous requests from sources of the same priority, then interrupts will be serviced in the following order:

X0, S1, T0, X1, T1, S0

A low priority interrupt routine can only be interrupted by a high priority interrupt. A high priority interrupt routine can not be interrupted.

Oscillator circuitry

The oscillator circuitry of the PCB83C652 is a single-stage inverting amplifier in a Pierce oscillator configuration. The circuitry between XTAL 1 and XTAL 2 is basically an inverter biased to the transfer point. Either a crystal or ceramic resonator can be used as the feedback element to complete the oscillator circuitry. Both are operated in parallel resonance. XTAL 1 is the high gain amplifier input, and XTAL 2 is the output (see Fig. 10). To drive the PCB83C652 externally, XTAL 1 is driven from an external source and XTAL 2 left open-circuit (see Fig. 11).

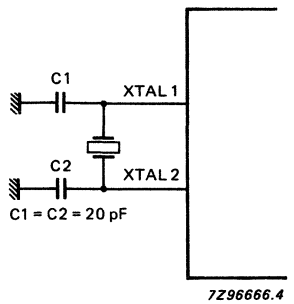


Fig. 10 PCB83C652 oscillator circuit.

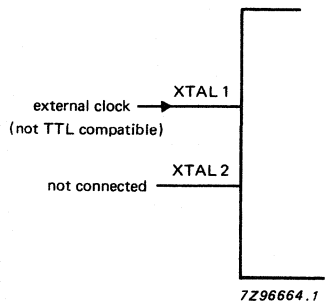


Fig. 11 Driving the PCB83C652 from an external source.

FUNCTIONAL DESCRIPTION (continued)

Reset circuitry (see Fig. 12)

The reset circuitry for the PCB83C652 is connected to the reset pin RST. A Schmitt trigger is used at the input for noise rejection. The output of the Schmitt trigger is sampled by the reset circuitry every machine cycle.

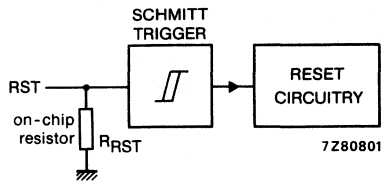


Fig. 12 On-chip reset configuration.

A reset is accomplished by holding the RST pin HIGH for at least two machine cycles (24 oscillator periods). The CPU responds by executing an internal reset. During reset ALE and $\overline{PSEN}$ output a HIGH level. In order to perform a correct reset, this level must not be affected by external elements.

The internal reset is executed during the second cycle in which RST is HIGH and is repeated every cycle until RST goes LOW. It leaves the internal registers as follows:

Register	Content
ACC	0000 0000
B	0000 0000
DPL	0000 0000
DPH	0000 0000
IE	0x00 0000
IP	xx00 0000
PCH	0000 0000
PCL	0000 0000
PCON	0xxx 0000
PSW	0000 0000
P0 to P3	1111 1111
S0BUF	xxxx xxxx
S0CON	0000 0000
S1ADR	0000 0000
S1CON	x000 0000
S1DAT	0000 0000
S1STA	1111 1000
SP	0000 0111
TCON	0000 0000
TH0, TH1	0000 0000
TL0, TL1	0000 0000
TMOD	0000 0000

The internal RAM is not affected by reset. When V_{DD} is turned on, the RAM content is indeterminate.

Power-on reset (see Fig. 13)

When V_{DD} is turned on an automatic reset can be obtained by connecting the RST pin to V_{DD} via a 2,2 μF capacitor. When the power is switched on, the voltage on the RST pin is equal to V_{DD} minus the capacitor voltage, and decreases from V_{DD} as the capacitor charges through the internal resistor (R_{RST}) to ground. The larger the capacitor, the more slowly V_{RST} decreases. V_{RST} must remain above the lower threshold of the Schmitt trigger long enough to effect a complete reset. The time required is the oscillator start-up time, plus 2 machine cycles.

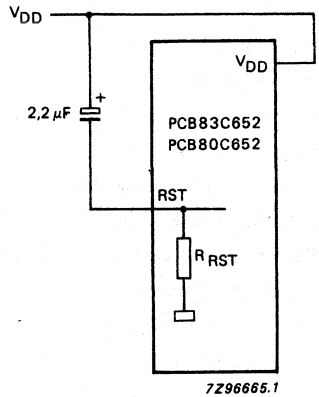


Fig. 13 Power-on reset.

INSTRUCTION SET

Table 4 Instruction set description

mnemonic		description	bytes/ cycles		opcode (hex.)
Arithmetic operations					
ADD	A,Rr	Add register to A	1	1	2*
ADD	A,direct	Add direct byte to A	2	1	25
ADD	A,@Ri	Add indirect RAM to A	1	1	26, 27
ADD	A,#data	Add immediate data to A	2	1	24
ADDC	A,Rr	Add register to A with carry flag	1	1	3*
ADDC	A,direct	Add direct byte to A with carry flag	2	1	35
ADDC	A,@Ri	Add indirect RAM to A with carry flag	1	1	36, 37
ADDC	A,#data	Add immediate data to A with carry flag	2	1	34
SUBB	A,Rr	Subtract register from A with borrow	1	1	9*
SUBB	A,direct	Subtract direct byte from A with borrow	2	1	95
SUBB	A,@Ri	Subtract indirect RAM from A with borrow	1	1	96, 97
SUBB	A,#data	Subtract immediate data from A with borrow	2	1	94
INC	A	Increment A	1	1	04
INC	Rr	Increment register	1	1	0*
INC	direct	Increment direct byte	2	1	05
INC	@Ri	Increment indirect RAM	1	1	06, 07
DEC	A	Decrement A	1	1	14
DEC	Rr	Decrement register	1	1	1*
DEC	direct	Decrement direct byte	2	1	15
DEC	@Ri	Decrement indirect RAM	1	1	16, 17
INC	DPTR	Increment data pointer	1	2	A3
MUL	AB	Multiply A & B	1	4	A4
DIV	AB	Divide A by B	1	4	84
DA	A	Decimal adjust A	1	1	D4

mnemonic		description	bytes/ cycles		opcode (hex.)
Logic operations					
ANL	A,Rr	AND register to A	1	1	5*
ANL	A,direct	AND direct byte to A	2	1	55
ANL	A,@Ri	AND indirect RAM to A	1	1	56, 57
ANL	A,#data	AND immediate data to A	2	1	54
ANL	direct,A	AND A to direct byte	2	1	52
ANL	direct,#data	AND immediate data to direct byte	3	2	53
ORL	A,Rr	OR register to A	1	1	4*
ORL	A,direct	OR direct byte to A	2	1	45
ORL	A,@Ri	OR indirect RAM to A	1	1	46, 47
ORL	A,#data	OR immediate data to A	2	1	44
ORL	direct,A	OR A to direct byte	2	1	42
ORL	direct,#data	OR immediate data to direct byte	3	2	43
XRL	A,Rr	Exclusive-OR register to A	1	1	6*
XRL	A,direct	Exclusive-OR direct byte to A	2	1	65
XRL	A,@Ri	Exclusive-OR indirect RAM to A	1	1	66, 67
XRL	A,#data	Exclusive-OR immediate data to A	2	1	64
XRL	direct,A	Exclusive-OR A to direct byte	2	1	62
XRL	direct,#data	Exclusive-OR immediate data to direct byte	3	2	63
CLR	A	Clear A	1	1	E4
CPL	A	Complement A	1	1	F4
RL	A	Rotate A left	1	1	23
RLC	A	Rotate A left through the carry flag	1	1	33
RR	A	Rotate A right	1	1	03
RRC	A	Rotate A right through the carry flag	1	1	13
SWAP	A	Swap nibbles within A	1	1	C4

INSTRUCTION SET (continued)

Table 4 (continued)

mnemonic	description	bytes/ cycles		opcode (hex.)
Data transfer				
MOV A,Rr	Move register to A	1	1	E*
MOV A,direct **	Move direct byte to A	2	1	E5
MOV A,@Ri	Move indirect RAM to A	1	1	E6, E7
MOV A,#data	Move immediate data to A	2	1	74
MOV Rr,A	Move A to register	1	1	F*
MOV Rr,direct	Move direct byte to register	2	2	A*
MOV Rr,#data	Move immediate data to register	2	1	7*
MOV direct,A	Move A to direct byte	2	1	F5
MOV direct,Rr	Move register to direct byte	2	2	8*
MOV direct,direct	Move direct byte to direct byte	3	2	85
MOV direct,@Ri	Move indirect RAM to direct byte	2	2	86, 87
MOV direct,#data	Move immediate data to direct byte	3	2	75
MOV @Ri,A	Move A to indirect RAM	1	1	F6, F7
MOV @Ri,direct	Move direct byte to indirect RAM	2	2	A6, A7
MOV @Ri,#data	Move immediate data to indirect RAM	2	1	76, 77
MOV DPTR,#data 16	Load data pointer with a 16-bit constant	3	2	90
MOVC A,@A+DPTR	Move code byte relative to DPTR to A	1	2	93
MOVC A,@A+PC	Move code byte relative to PC to A	1	2	83
MOVX A,@Ri	Move external RAM (8-bit address) to A	1	2	E2, E3
MOVX A,@DPTR	Move external RAM (16-bit address) to A	1	2	E0
MOVX @Ri,A	Move A to external RAM (8-bit address)	1	2	F2, F3
MOVX @DPTR,A	Move A to external RAM (16-bit address)	1	2	F0
PUSH direct	Push direct byte onto stack	2	2	C0
POP direct	Pop direct byte from stack	2	2	D0
XCH A,Rr	Exchange register with A	1	1	C*
XCH A,direct	Exchange direct byte with A	2	1	C5
XCH A,@Ri	Exchange indirect RAM with A	1	1	C6, C7
XCHD A,@Ri	Exchange LOW-order nibble indirect RAM with A	1	1	D6, D7

** MOV A,ACC is not a valid instruction.

mnemonic		description	bytes/ cycles		opcode (hex.)
Boolean variable manipulation					
CLR	C	Clear carry flag	1	1	C3
CLR	bit	Clear direct bit	2	1	C2
SETB	C	Set carry flag	1	1	D3
SETB	bit	Set direct bit	2	1	D2
CPL	C	Complement carry flag	1	1	B3
CPL	bit	Complement direct bit	2	1	B2
ANL	C,bit	AND direct bit to carry flag	2	2	82
ANL	C,/bit	AND complement of direct bit to carry flag	2	2	B0
ORL	C,bit	OR direct bit to carry flag	2	2	72
ORL	C,/bit	OR complement of direct bit to carry flag	2	2	A0
MOV	C,bit	Move direct bit to carry flag	2	1	A2
MOV	bit,C	Move carry flag to direct bit	2	2	92
Program and machine control					
ACALL	addr11	Absolute subroutine call	2	2	●1addr
LCALL	addr16	Long subroutine call	3	2	12
RET		Return from subroutine	1	2	22
RETI		Return from interrupt	1	2	32
AJMP	addr11	Absolute jump	2	2	▲1addr
LJMP	addr16	Long jump	3	2	02
SJMP	rel	Short jump (relative address)	2	2	80
JMP	@A+DPTR	Jump indirect relative to the DPTR	1	2	73
JZ	rel	Jump if A is zero	2	2	60
JNZ	rel	Jump if A is not zero	2	2	70
JC	rel	Jump if carry flag is set	2	2	40
JNC	rel	Jump if carry flag is not set	2	2	50
JB	bit,rel	Jump if direct bit is set	3	2	20
JNB	bit,rel	Jump if direct bit is not set	3	2	30
JBC	bit,rel	Jump if direct bit is set and clear bit	3	2	10
CJNE	A,direct,rel	Compare direct to A and jump if not equal	3	2	B5
CJNE	A,#data,rel	Compare immediate to A and jump if not equal	3	2	B4
CJNE	Rr,#data,rel	Compare immed. to reg. and jump if not equal	3	2	B*
CJNE	@Ri,#data,rel	Compare immed. to ind. and jump if not equal	3	2	B6, B7
DJNZ	Rr,rel	Decrement register and jump if not zero	2	2	D*
DJNZ	direct,rel	Decrement direct and jump if not zero	3	2	D5
NOP		No operation	1	1	00

Notes to Table 4

Data addressing modes

Rr	Working register R0-R7.
direct	128 internal RAM locations and any special function register (SFR).
@Ri	Indirect internal RAM location addressed by register R0 or R1 of the actual register bank.
#data	8-bit constant included in instruction.
#data16	16-bit constant included as bytes 2 and 3 of instruction.
bit	direct addressed bit in internal RAM or SFR.
addr16	16-bit destination address. Used by LCALL and LJMP. The branch will be anywhere within the 64 K-byte program memory address space.
addr11	11-bit destination address. Used by ACALL and AJMP. The branch will be within the same 2 K-byte page of program memory as the first byte of the following instruction.
rel	Signed (two's complement) 8-bit offset byte. Used by SJMP and all conditional jumps. Range is −128 to + 127 bytes relative to first byte of the following instruction.

Hexadecimal opcode cross-reference to Table 4

* : 8, 9, A, B, C, D, E, F.
● : 11, 31, 51, 71, 91, B1, D1, F1.
▲ : 01, 21, 41, 61, 81, A1, C1, E1.

Table 5 Instruction map

first hexadecimal chacter of opcode ↓ / second hexadecimal character of opcode →

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	NOP	AJMP addr11	LJMP addr16	RR A	INC A	INC dir	INC @Ri 0	1	INC Rr 0	1	2	3	4	5	6	7
1	JBC bit,rel	ACALL addr11	LCALL addr16	RRC A	DEC A	DEC dir	DEC @Ri 0	1	DEC Rr 0	1	2	3	4	5	6	7
2	JB bit,rel	AJMP addr11	RET	RL A	ADD A,#data	ADD A,dir	ADD A,@Ri 0	1	ADD A,Rr 0	1	2	3	4	5	6	7
3	JNB bit,rel	ACALL addr11	RETI	RLC A	ADDC A,#data	ADDC A,dir	ADDC A,@Ri 0	1	ADDC A,Rr 0	1	2	3	4	5	6	7
4	JC rel	AJMP addr11	ORL dir,A	ORL dir,#data	ORL A,#data	ORL A,dir	ORL A,@Ri 0	1	ORL A,Rr 0	1	2	3	4	5	6	7
5	JNC rel	ACALL addr11	ANL dir,A	ANL dir,#data	ANL A,#data	ANL A,dir	ANL A,@Ri 0	1	ANL A,Rr 0	1	2	3	4	5	6	7
6	JZ rel	AJMP addr11	XRL dir,A	XRL dir,#data	XRL A,#data	XRL A,dir	XRL A,@Ri 0	1	XRL A,Rr 0	1	2	3	4	5	6	7
7	JNZ rel	ACALL addr11	ORL C,bit	JMP @A+DPTR	MOV A,#data	MOV dir,#data	MOV @Ri,#data 0	1	MOV Rr,#data 0	1	2	3	4	5	6	7
8	SJMP rel	AJMP addr11	ANL C,bit	MOVC A,@A+PC	DIV AB	MOV dir,dir	MOV dir,@Ri 0	1	MOV dir,Rr 0	1	2	3	4	5	6	7
9	MOV DPTR, #data	ACALL addr11	MOV bit,C	MOVC A,@A+DPTR	SUBB A,#data	SUBB A,dir	SUBB A,@Ri 0	1	SUBB A,Rr 0	1	2	3	4	5	6	7
A	ORL C,/bit	AJMP addr11	MOV C,bit	INC DPTR	MUL AB		MOV @Ri,dir 0	1	MOV Rr,dir 0	1	2	3	4	5	6	7
B	ANL C,/bit	ACALL addr11	CPL bit	CPL C	CJNE A, #data,rel	CJNE A,dir,rel	CJNE @Ri,#data,rel 0	1	CJNE Rr,#data,rel 0	1	2	3	4	5	6	7
C	PUSH dir	AJMP addr11	CLR bit	CLR C	SWAP A	XCH A,dir	XCH A,@Ri 0	1	XCH A,Rr 0	1	2	3	4	5	6	7
D	POP dir	ACALL addr11	SETB bit	SETB C	DA A	DJNZ dir,rel	XCHD A,@Ri 0	1	DJNZ Rr,rel 0	1	2	3	4	5	6	7
E	MOVX A,@DPTR	AJMP addr11	MOVX A,@Ri 0	1	CLR A	MOV * A,dir	MOV A,@Ri 0	1	MOV A,Rr 0	1	2	3	4	5	6	7
F	MOVX @DPTR,A	ACALL addr11	MOVX @Ri,A 0	1	CPL A	MOV dir,A	MOV @Ri,A 0	1	MOV Rr,A 0	1	2	3	4	5	6	7

* MOV A,ACC is not a valid instruction.

RATINGS

Limiting values in accordance with the Absolute Maximum System (IEC 134)

parameter	symbol	min.	max.	unit
Input voltage on any pin with respect to ground (V_{SS})	V_I	−0,5	+ 6,5	V
→ Input, output current on any single pin	I_I, I_O	–	± 5	mA
→ Input, output current on any two pins	I_I, I_O	–	± 10	mA
Total power dissipation	P_{tot}	–	1	W
Storage temperature range	T_{stg}	−65	+ 150	°C
Operating ambient temperature range				
PCB83C652 versions	T_{amb}	0	+ 70	°C
→ PCF83C652 versions	T_{amb}	−40	+ 85	°C
→ PCA83C652 versions	T_{amb}	−40	+ 125	°C

DC CHARACTERISTICS

V_{DD} = 5 V (± 10%); V_{SS} = 0 V; T_{amb} = 0 to 70 °C (PCB83C652); −40 to 85 °C (PCF83C652); −40 to 125 °C (PCA83C652). All voltages with respect to V_{SS} unless otherwise specified.

parameter	conditions	symbol	min.	max.	unit	
Supply voltage		V_{DD}	4.5	5.5	V	
Supply current						
operating	(notes 1 and 2)	I_{DD}	–	24	mA	
	(notes 1 and 3)	I_{DD}	–	27	mA	←
idle mode	(notes 2 and 4)	I_{ID}	–	5.0	mA	←
	(notes 3 and 4)	I_{ID}	–	6.0	mA	←
Power-down current	(notes 2, 5 and 6)	I_{PD}	–	50	µA	←
	(notes 3, 5 and 6)	I_{PD}	–	120	µA	←
Inputs						←
LOW level input voltage (except $\overline{EA}$, P1.6/SCL, P1.7/SDA)		V_{IL}	−0.5	$0.2V_{DD}-0.1$	V	
LOW level input voltage ($\overline{EA}$)		V_{IL1}	−0.5	$0.2V_{DD}-0.3$	V	
LOW level input voltage (P1.6/SCL, P1.7/SDA)*		V_{IL2}	−0.5	1.5	V	
HIGH level input voltage (except RST, XTAL 1, P1.6/SCL, P1.7/SDA)		V_{IH}	$0.2V_{DD}+0.9$	$V_{DD}+0.5$	V	
HIGH level input voltage (RST and XTAL 1)		V_{IH1}	$0.7V_{DD}$	$V_{DD}+0.5$	V	
HIGH level input voltage (P1.6/SCL, P1.7/SDA)*		V_{IH2}	3.0	6	V	
Input current logic 0 (Ports 1, 2 and 3 except P1.6/SCL, P1.7/SDA)	V_I = 0.45 V	$-I_{IL}$	–	50	µA	
Input current logic 1 to 0 transition (Ports 1, 2, 3 except P1.6/SCL, P1.7/SDA)	V_I = 2.0 V	$-I_{TL}$	–	650	µA	
Input leakage current (Port 0, $\overline{EA}$)	$0.45\ V < V_I < V_{DD}$	$\pm I_{LI1}$	–	10	µA	
Input leakage current (P1.6/SCL, P1.7/SDA)	$0\ V < V_I < 6\ V$; $0\ V < V_{DD} < 5.5\ V$	$\pm I_{LI2}$	–	10	µA	

* The input threshold voltage of P1.6 and P1.7 (SIO1) meets the I^2C specification, so an input voltage below 1.5 V will be recognized as a logic 0 while an input voltage above 3.0 V will be recognized as a logic 1.

DEVELOPMENT DATA

DC CHARACTERISTICS (continued)

parameter	conditions	symbol	min.	max.	unit
Outputs					
LOW level output voltage (Ports 1, 2, 3 except P1.6/SCL, P1.7/SDA)*	I_{OL} = 1.6 mA	V_{OL}	–	0.45	V
LOW level output voltage (Port 0, ALE, $\overline{PSEN}$)*	I_{OL} = 3.2 mA	V_{OL1}	–	0.45	V
LOW level output voltage (P1.6/SCL, P1.7/SDA)	I_{OL} = 3.0 mA	V_{OL2}	–	0.40	V
HIGH level output voltage (Ports 1, 2, 3)	$-I_{OH}$ = 60 μA	V_{OH}	2.4	–	V
	$-I_{OH}$ = 25 μA	V_{OH}	0.75V_{DD}	–	V
	$-I_{OH}$ = 10 μA	V_{OH}	0.9V_{DD}	–	V
HIGH level output voltage (Port 0 in external bus mode, ALE, $\overline{PSEN}$)**	$-I_{OH}$ = 400 μA	V_{OH1}	2.4	–	V
	$-I_{OH}$ = 150 μA	V_{OH1}	0.75V_{DD}	–	V
	$-I_{OH}$ = 40 μA	V_{OH1}	0.9V_{DD}	–	V
RST pull-down resistor		R_{RST}	50	150	kΩ
I/O pin capacitance	test freq. = 1 MHz; T_{amb} = 25 °C	$C_{I/O}$	–	10	pF

Notes to the DC characteristics

1. The operating supply current is measured with all output pins disconnected; XTAL 1 driven with $t_r = t_f$ = 10 ns; $V_{IL} = V_{SS}$ + 0.5 V; $V_{IH} = V_{DD}$ −0.5 V; XTAL 2 not connected; $\overline{EA}$ = RST = Port 0 = P1.6 = P1.7 = V_{DD}; f_{CLK} = 12 MHz.
2. This is a preliminary value and applies to PCB83C652 (0 to 70 °C) and PCF83C652 (−40 to 85 °C).
3. This is a preliminary value and applies to PCA83C652 (−40 to 125 °C).
4. The idle mode supply current is measured with all output pins disconnected; XTAL 1 driven with $t_r = t_f$ = 10 ns; $V_{IL} = V_{SS}$ + 0.5 V; $V_{IH} = V_{DD}$ −0.5 V; XTAL 2 not connected; $\overline{EA}$ = Port 0 = P1.6 = P1.7 = V_{DD}; RST = V_{SS}; f_{CLK} = 12 MHz.
5. The power-down current is measured with all output pins disconnected; XTAL 2 not connected; $\overline{EA}$ = Port 0 = P1.6 = P1.7 = V_{DD}; RST = V_{SS}.
6. 2 V $\leqslant V_{PD} \leqslant V_{DD}$ max..

* Capacitive loading on Port 0 and Port 2 may cause spurious noise pulses to be superimposed on the LOW level output voltage of ALE, Port 1 and Port 3. The noise is due to external bus capacitance discharging into the Port 0 and Port 2 pins when these pins make a 1-to-0 transition during Bus operations. In the most adverse condition (capacitive loading > 100 pF) the noise pulse on ALE line may exceed 0.8 V. In this event it may be required to qualify ALE with a Schmitt trigger, or use an address latch with a Schmitt trigger STROBE input.

** Capacitive loading on Port 0 and Port 2 may cause the HIGH level output voltage on ALE and $\overline{PSEN}$ to momentarily fall below the 0.9 V_{DD} specification when the address bits are stabilizing.

DEVELOPMENT DATA

AC CHARACTERISTICS

V_{DD} = 5 V (± 10%); V_{SS} = 0 V; T_{amb} = 0 to 70 °C (PCB83C652); −40 to 85 °C (PCF83C652); −40 to 125 °C (PCA83C652). C_L = 100 pF (Port 0, ALE and $\overline{PSEN}$); C_L = 80 pF (all other outputs); unless otherwise specified (see waveforms Figs 16, 17 and 18).

parameter	symbol	10 MHz		12 MHz		variable clock		unit
		min.	max.	min.	max.	min.	max.	
Program memory								
ALE pulse duration	t_{LL}	160	–	127	–	$2t_{CK}-40$	–	ns
Address set-up time to ALE	t_{AL}	45	–	28	–	$t_{CK}-55$	–	ns
Address hold time after ALE	t_{LA}	65	–	48	–	$t_{CK}-35$	–	ns
Time from ALE to valid instruction input	t_{LIV}	–	300	–	233	–	$4t_{CK}-100$	ns
Time from ALE to control pulse $\overline{PSEN}$	t_{LC}	60	–	43	–	$t_{CK}-40$	–	ns
Control pulse duration $\overline{PSEN}$	t_{CC}	255	–	205	–	$3t_{CK}-45$	–	ns
Time from $\overline{PSEN}$ to valid instruction input	t_{CIV}	–	195	–	145	–	$3t_{CK}-105$	ns
Input instruction hold time after $\overline{PSEN}$	t_{CI}	0	–	0	–	0	–	ns
Input instruction float delay after $\overline{PSEN}$	t_{CIF}	–	75	–	59	–	$t_{CK}-25$	ns
Address to valid instruction input	t_{AIV}	–	395	–	312	–	$5t_{CK}-105$	ns
Address float delay after $\overline{PSEN}$	t_{AFC}	–	10	–	10	–	10	ns

AC CHARACTERISTICS (continued)

parameter	symbol	10 MHz		12 MHz		variable clock		unit
		min.	max.	min.	max.	min.	max.	
External data memory								
$\overline{RD}$ pulse duration	t_{RR}	500	–	400	–	$6t_{CK}-100$	–	ns
$\overline{WR}$ pulse duration	t_{WW}	500	–	400	–	$6t_{CK}-100$	–	ns
Address set-up time to ALE	t_{AL}	45	–	28	–	$t_{CK}-55$	–	ns
Address hold time after ALE	t_{LA}	65	–	48	–	$t_{CK}-35$	–	ns
$\overline{RD}$ to valid data input	t_{RD}	–	335	–	252	–	$5t_{CK}-165$	ns
Data hold time after $\overline{RD}$	t_{DR}	0	–	0	–	0	–	ns
Data float delay after $\overline{RD}$	t_{DFR}	–	130	–	97	–	$2t_{CK}-70$	ns
Time from ALE to valid data input	t_{LD}	–	650	–	517	–	$8t_{CK}-150$	ns
Address to valid data input	t_{AD}	–	735	–	585	–	$9t_{CK}-165$	ns
Time from ALE to $\overline{RD}$ or $\overline{WR}$	t_{LW}	250	350	200	300	$3t_{CK}-50$	$3t_{CK}+50$	ns
Time from address to $\overline{RD}$ or $\overline{WR}$	t_{AW}	270	–	203	–	$4t_{CK}-130$	–	ns
Time from $\overline{RD}$ or $\overline{WR}$ HIGH to ALE HIGH	t_{WHLH}	60	140	43	123	$t_{CK}-40$	$t_{CK}+40$	ns
Data valid to $\overline{WR}$ transition	t_{DWX}	40	–	23	–	$t_{CK}-60$	–	ns
Data set-up time before $\overline{WR}$	t_{DW}	550	–	433	–	$7t_{CK}-150$	–	ns
Data hold time after $\overline{WR}$	t_{WD}	50	–	33	–	$t_{CK}-50$	–	ns
Address float delay after $\overline{RD}$	t_{AFR}	–	0	–	0	–	0	ns

Where:

$1/t_{CK}$ = 1,2 to 12 MHz (see Fig. 15 and Table 6)

t_{CY} = 12 t_{CK} (see Fig. 16 and DC characteristics)

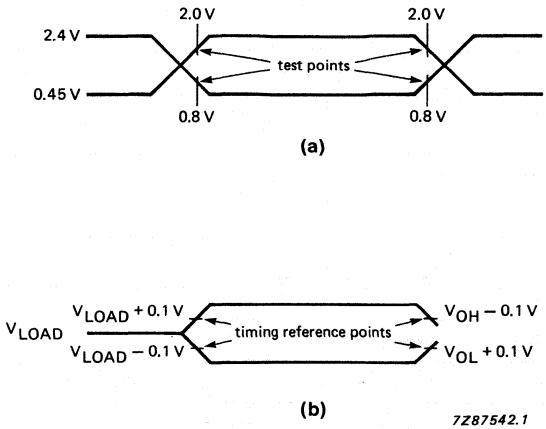


Fig. 14 AC testing input, output waveform (a) and float waveform (b).

AC testing inputs are driven at 2.4 V for a logic 1 and 0.45 V for a logic 0. Timing measurements are taken at 2.0 V for a logic 1 and 0.8 V for logic 0. The float state is defined as the point at which a Port 0 pin sinks 3.2 mA or sources 400 μA at the voltage test levels.

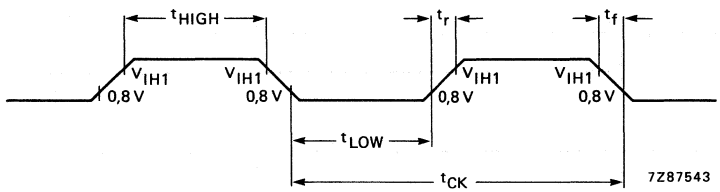


Fig. 15 External clock drive XTAL 1 (see Table 6).

Table 6 External clock drive XTAL 1 (see Fig. 15)

parameter	symbol	variable clock (f = 1.2 to 12 MHz)		unit
		min.	max.	
oscillator clock period	t_{CK}	83	833	ns
HIGH time	t_{HIGH}	20	$t_{CK}-t_{LOW}$	ns
LOW time	t_{LOW}	20	$t_{CK}-t_{HIGH}$	ns
rise time	t_r	–	20	ns
fall time	t_f	–	20	ns
cycle time	$t_{CY} = 12\ t_{CK}$	1	10	μs

DEVELOPMENT DATA

AC CHARACTERISTICS (continued)

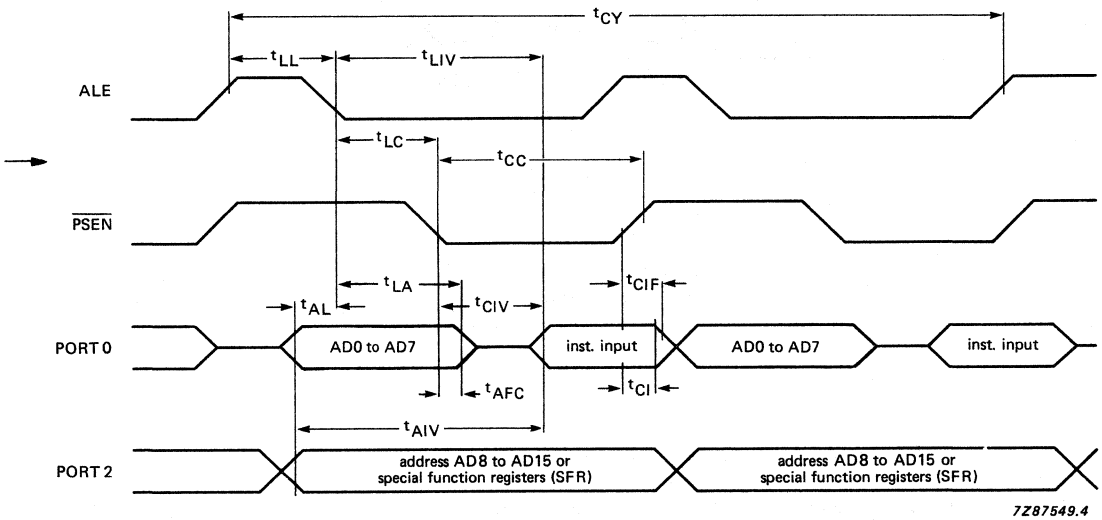


Fig. 16 Read from program memory.

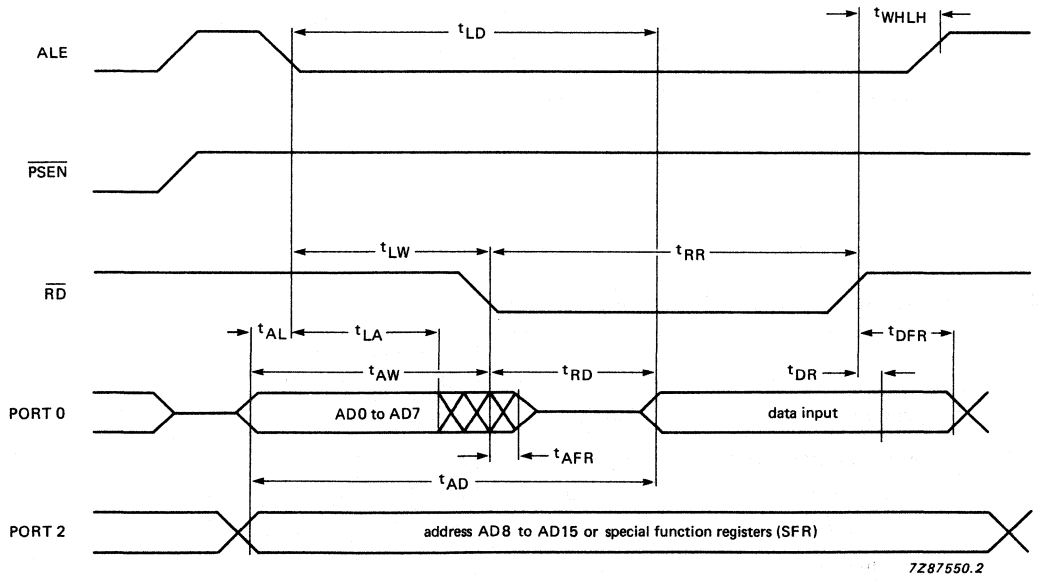


Fig. 17 Read from data memory.

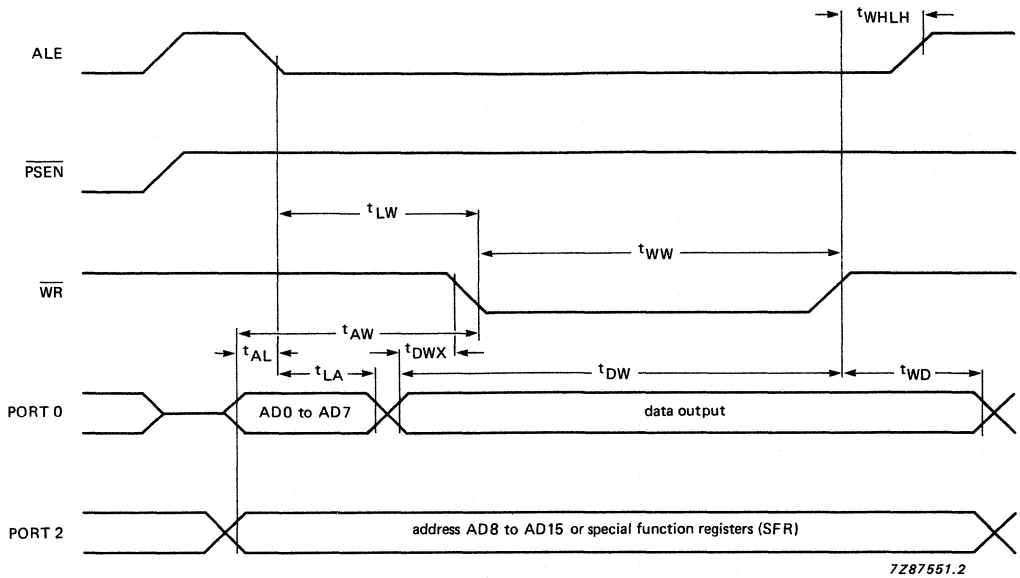


Fig. 18 Write to data memory.

DEVELOPMENT DATA

Purchase of Philips' I²C components conveys a license under the Philips' I²C patent to use the components in the I²C-system provided the system conforms to the I²C specifications defined by Philips.

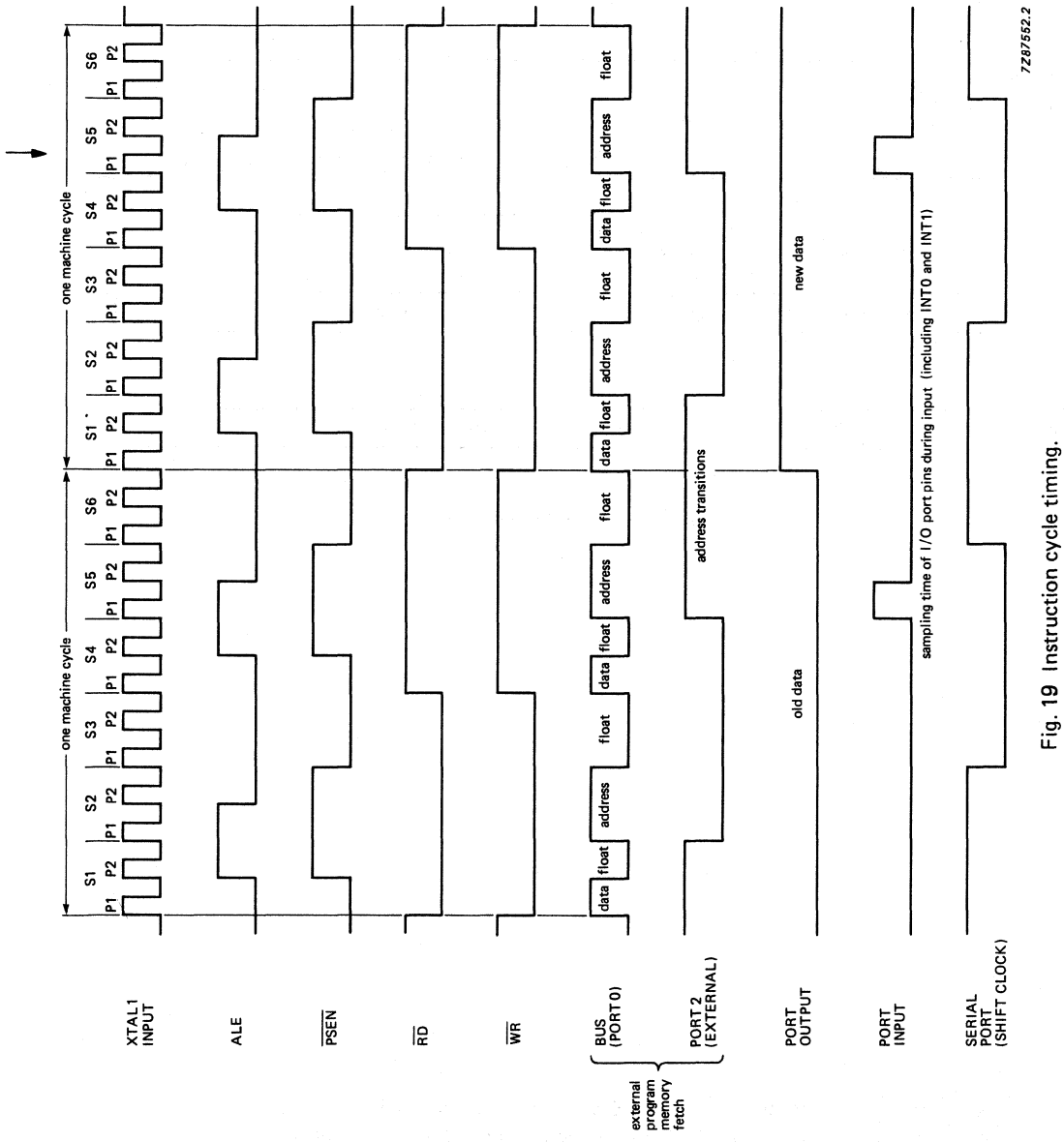


Fig. 19 Instruction cycle timing.

SINGLE-CHIP 8-BIT MICROCONTROLLER

GENERAL DESCRIPTION

The PCB83C662 single-chip 8-bit microcontroller is manufactured in an advanced CMOS process and is a derivative of the PCB80C51 microcontroller family. The PCB83C662 has the same instruction set as the PCB80C51. Two versions of the derivative exist although the generic term "PCB83C662" is used to refer to both family members:

- PCB83C662: 8 K bytes mask-programmable ROM, 256 bytes RAM
- PCB80C662: ROM-less version of the PCB83C662

This device provides architectural enhancements that make it applicable in a variety of applications in general control systems.
The PCB83C662 contains a non-volatile 8 K x 8 read-only program memory, a volatile 256 x 8 read/write data memory, four 8-bit I/O ports, two 16-bit timer/event counters (identical to the timers of the 80C51), a multi-source, two-priority-level, nested interrupt structure, UART and on-chip oscillator and timing circuits. For systems that require extra capability, the PCB83C662 can be expanded using standard TTL compatible memories and logic.
The device also functions as an arithmetic processor having facilities for both binary and BCD arithmetic plus bit-handling capabilities. The instruction set consists of over 100 instructions: 49 one-byte, 45 two-byte and 17 three-byte. With a 12 MHz crystal, 58% of the instructions are executed in 1 μs and 40% in 2 μs. Multiply and divide instructions require 4 μs.

Features

- 80C51 central processing unit
- 8 K x 8 ROM, expandable externally to 64 K bytes
- 256 x 8 RAM, expandable externally to 64 K bytes
- Two standard 16-bit timer/counters
- Four 8-bit I/O ports
- Full-duplex UART facilities
- Three temperature "ranges" 6 available
  0 to + 70 °C; PCB83C662 versions
  −40 to + 85 °C; PCF83C662 versions
  −40 to + 110 °C; PCA83C662 versions
- Extended frequency range; 1,2 MHz to 12 MHz

CHANGE IN ARCHITECTURE UNDERWAY, NEW ISSUE IN PREPARATION

PACKAGE OUTLINES

PCA/PCB/PCF83C662P; PCA/PCB/PCF80C662P: 40-lead DIL; plastic (SOT129).
PCA/PCB/PCF83C662WP; PCA/PCB/PCF80C662WP: 44-lead plastic leaded-chip-carrier (PLCC) (SOT187 pedestal or SOT187AA pocket versions, these are interchangeable).
PCA/PCB/PCF83C662H; PCA/PCB/PCF80C662H: 44-lead quad flat-pack (QFP); plastic (SOTXXX).

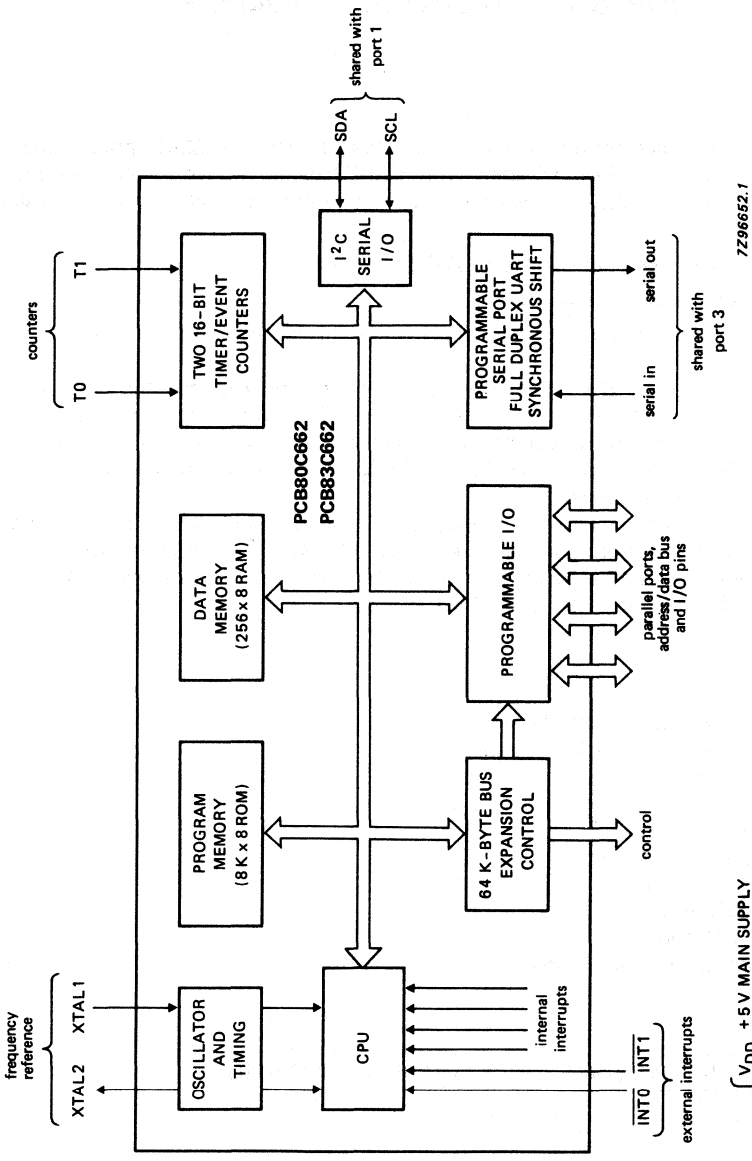


Fig. 1 Block diagram.

DEVELOPMENT DATA

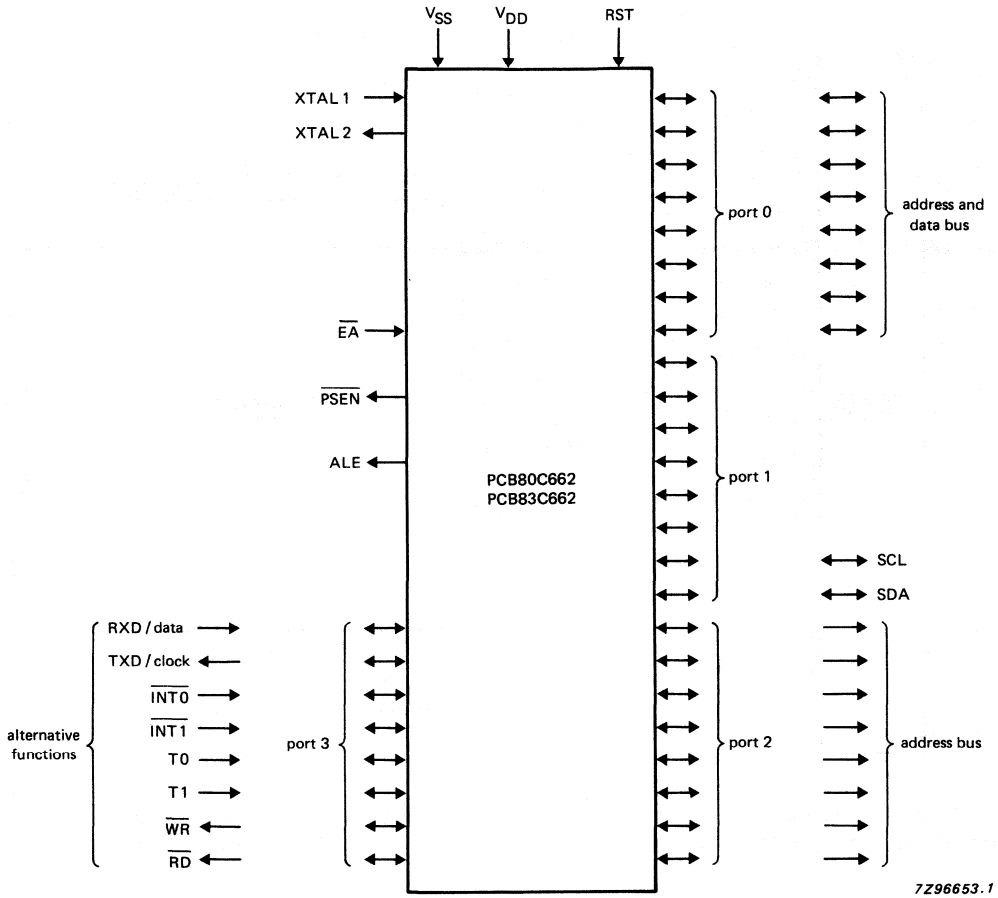


Fig. 2 Functional diagram.

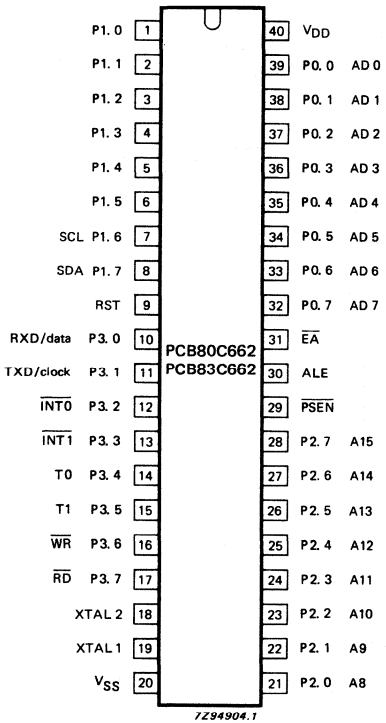


Fig. 3a Pinning diagram for 40-lead DIL package (SOT129).

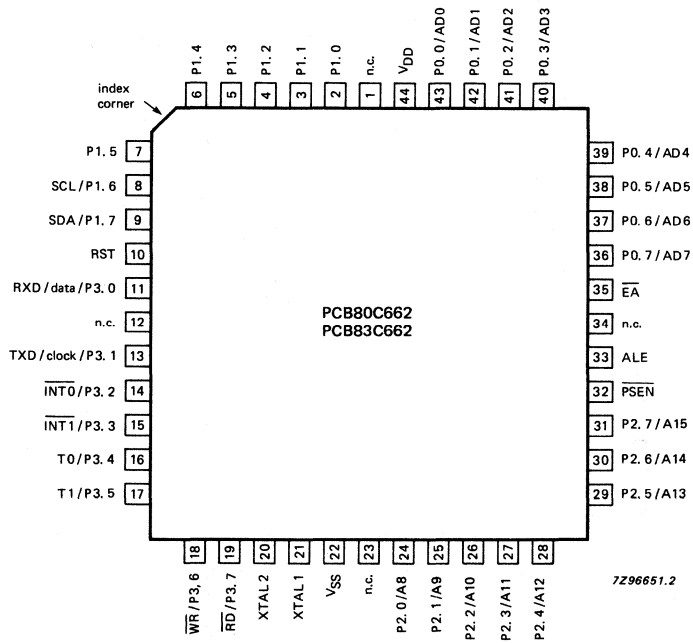


Fig. 3b Pinning diagram for 44-lead PLCC package (SOT187).

PINNING (PCB83C662P, PCB80C662P)

1-8	P1.0-P1.7	**Port 1:** 8-bit quasi-bidirectional I/O port. Port 1 can sink/source one TTL (= 4 LSTTL) input. It can drive CMOS inputs without external pull-ups, except P1.6 and P1.7 which have open drain outputs.
9	RST	**RESET:** a high level on this pin for two machine cycles while the oscillator is running resets the device. An internal pull-down resistor permits Power-On reset using only a capacitor connected to V_{DD}.
10-17	P3.0-P3.7	**Port 3:** 8-bit quasi-bidirectional I/O port with internal pull-ups. It also serves the following alternative functions:

Port pin	Alternative function
P3.0	RXD/data: serial port receiver data input (asynchronous) or data input/output (synchronous)
P3.1	TXD/clock: serial port transmitter data output (asynchronous) or clock output (synchronous)
P3.2	$\overline{INT0}$: external interrupt 0 or gate control input for timer/event counter 0
P3.3	$\overline{INT1}$: external interrupt 1 or gate control input for timer/event counter 1
P3.4	T0: external input for timer/event counter 0
P3.5	T1: external input for timer/event counter 1
P3.6	$\overline{WR}$: external data memory write strobe
P3.7	$\overline{RD}$: external data memory read strobe

The generation or use of a Port 3 pin as an alternative function is carried out automatically by the PCB83C662 provided the associated Special Function Register bit is set high. Port 3 can sink/source one TTL (= 4 LSTTL) input. It can drive CMOS inputs without external pull-ups.

18	XTAL 2	**Crystal input 2:** output of the inverting amplifier that forms the oscillator. Left open-circuit when an external oscillator clock is used (see Figs 10 and 11).
19	XTAL 1	**Crystal input 1:** input to the inverting amplifier that forms the oscillator, and input to the internal clock generator. Receives the external oscillator clock signal when an external oscillator is used (see Figs 10 and 11).
20	V_{SS}	**Ground:** circuit ground potential.
21-28	P2.0-P2.7	**Port 2:** 8-bit quasi-bidirectional I/O port with internal pull-ups. During access to external memories (RAM/ROM) that use 16-bit addresses (MOVX @DPTR) Port 2 emits the high order address byte. When external RAM is accessed with an 8-bit address (MOVX @Ri) Port 2 emits the contents of the P2 special function register. Port 2 can sink/source one TTL (= 4 LSTTL) input. It can drive CMOS inputs without external pull-ups.

DEVELOPMENT DATA

PINNING (continued)

29	$\overline{PSEN}$	**Program Store Enable output:** read strobe to the external program memory via port 0 and 2. It is activated twice each machine cycle during fetches from external program memory. When executing out of external program memory two activations of $\overline{PSEN}$ are skipped during each access to external data memory. $\overline{PSEN}$ is not activated (remains HIGH) during no fetches from external program memory. $\overline{PSEN}$ can sink/source 8 LSTTL inputs. It can drive CMOS inputs without an external pull-up.
30	ALE	**Address Latch Enable output:** latches the low byte of the address during accesses to external memory in normal operation. It is activated every six oscillator periods except during an external data memory access. ALE can sink/source 8 LSTTL inputs. It can drive CMOS inputs without an external pull-up.
31	$\overline{EA}$	**External Access input:** When $\overline{EA}$ is held at a TTL high level the CPU executes out of the internal program ROM provided the program counter is less than 8192. When $\overline{EA}$ is held at a TTL low level, the CPU executes out of external program memory via Port 0 and Port 2. $\overline{EA}$ is not allowed to float.
32-39	P0.7-P0.0	**Port 0:** 8-bit open drain bidirectional I/O port. It is also the multiplexed low-order address and data bus during accesses to external memory (during these accesses it activates internal pull-ups). Port 0 can sink/source eight LSTTL inputs.
40	V_{DD}	**Power supply:** + 5 V power supply pin during normal operation, Idle mode and Power-down mode.

To avoid a 'latch-up' effect at power-on, the voltage on any pin at any time must not be higher or lower than V_{DD} + 0,5 V or V_{SS} − 0,5 V respectively.

SUPERSEDES DATA OF JUNE 1988

SINGLE-CHIP 8-BIT MICROCONTROLLER

GENERAL DESCRIPTION

The PCB83C654 single-chip 8-bit microcontroller is manufactured in an advanced CMOS process and is a derivative of the PCB80C51 microcontroller family. The PCB83C654 has the same instruction set as the PCB80C51. The ROM-less PCB80C652 should be used for development purposes.

This device provides architectural enhancements that make it applicable in a variety of applications in general control systems.

The PCB83C654 contains a non-volatile 16 K x 8 read-only program memory, a volatile 256 x 8 read/write data memory, four 8-bit I/O ports, two 16-bit timer/event counters (identical to the timers of the 80C51), a multi-source, two-priority-level, nested interrupt structure, an I^2C interface, UART and on-chip oscillator and timing circuits. For systems that require extra capability, the PCB83C654 can be expanded using standard TTL compatible memories and logic.

The device also functions as an arithmetic processor having facilities for both binary and BCD arithmetic plus bit-handling capabilities. The instruction set consists of over 100 instructions: 49 one-byte, 45 two-byte and 17 three-byte. With a 12 MHz crystal, 58% of the instructions are executed in 1 μs and 40% in 2 μs. Multiply and divide instructions require 4 μs.

Features

- 80C51 central processing unit
- 16 K x 8 ROM, expandable externally to 64 K bytes
- 256 x 8 RAM, expandable externally to 64 K bytes
- Two standard 16-bit timer/counters
- Four 8-bit I/O ports
- I^2C-bus serial I/O port with byte orientated master and slave functions
- Full-duplex UART facilities

- **A version for extended temperature range is in preparation**

PACKAGE OUTLINES

PCB83C654P : 40-lead DIL; plastic (SOT129).
PCB83C654WP: 44-lead plastic leaded chip-carrier (PLCC); (SOT187 pedestal or SOT187AA pocket version depending on source, versions are interchangeable).
PCB83C654H : 44-lead quad flat-pack; plastic (SOT205A) in preparation.

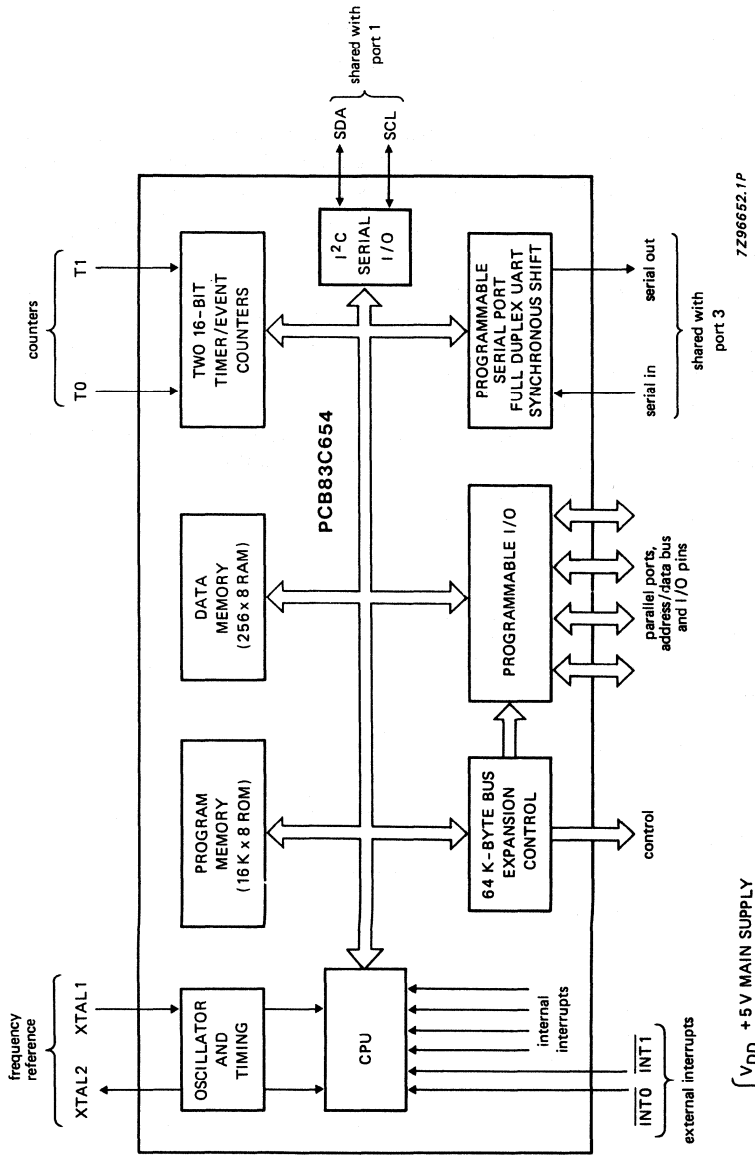


Fig. 1 Block diagram.

DEVELOPMENT DATA

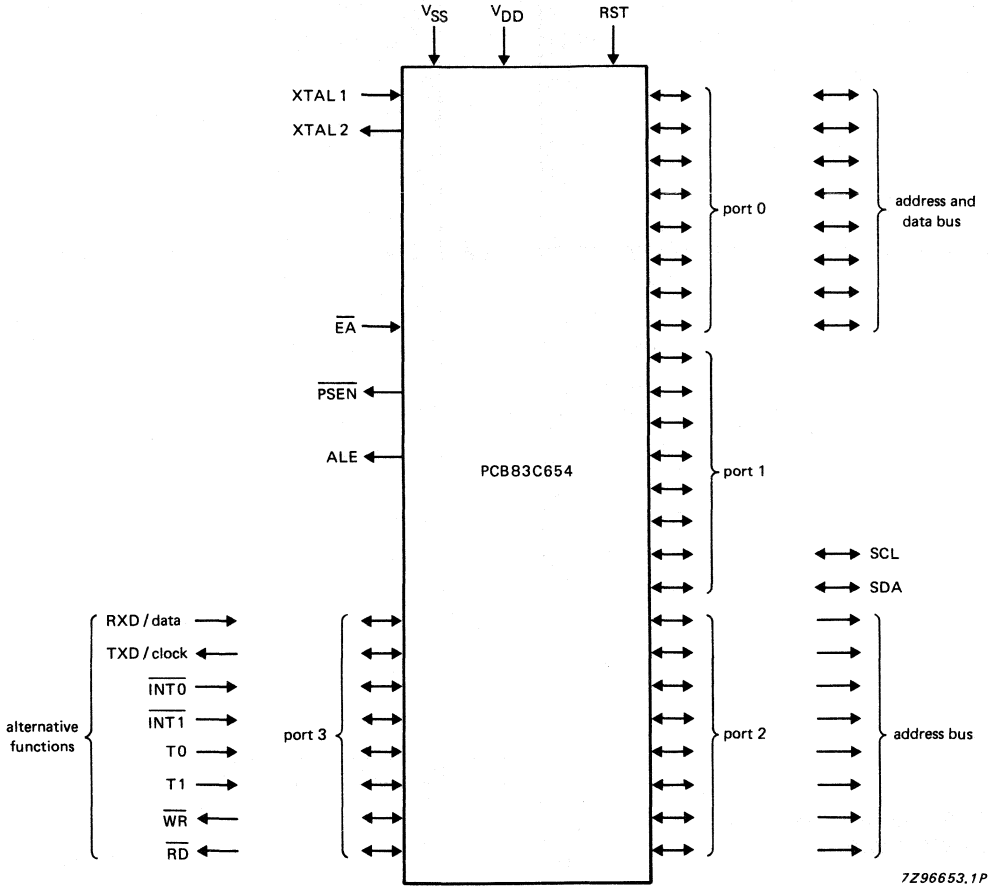


Fig. 2 Functional diagram.

Fig. 3a Pinning diagram for PCB83C654P.

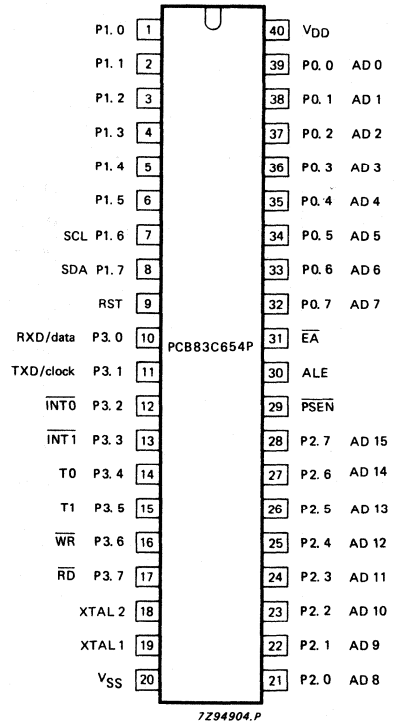
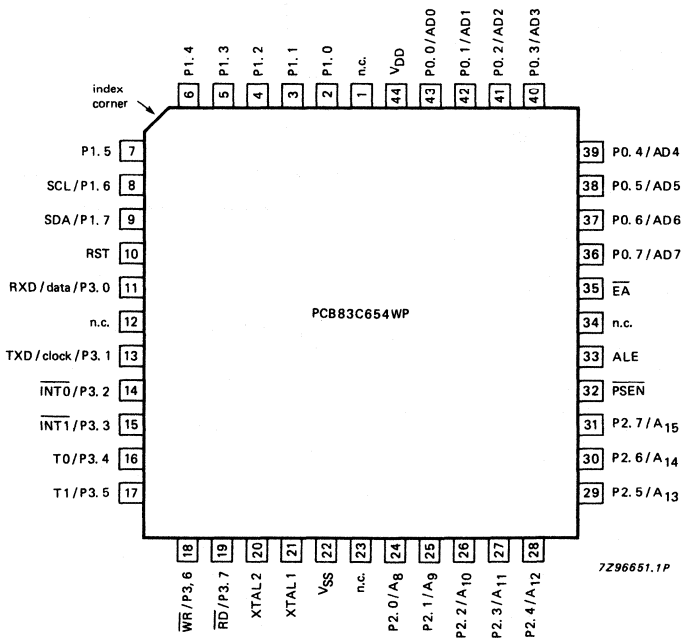


Fig. 3b Pinning diagram for PCB83C654WP.



PINNING (PCB83C654P, PCB80C654P)

1-8	P1.0-P1.7	**Port 1:** 8-bit quasi-bidirectional I/O port. Port 1 can sink/source one TTL (= 4 LSTTL) input. It can drive CMOS inputs without external pull-ups, except P1.6 and P1.7 which have open drain outputs.

Port pin	Alternative function
P1.6	SCL: I^2C-bus serial port clock line
P1.7	SDA: I^2C-bus serial port data line

9	RST	**RESET:** a high level on this pin for two machine cycles while the oscillator is running resets the device. An internal pull-down resistor permits Power-On reset using only a capacitor connected to V_{DD}.
10-17	P3.0-P3.7	**Port 3:** 8-bit quasi-bidirectional I/O port with internal pull-ups. It also serves the following alternative functions:

Port pin	Alternative function
P3.0	RXD/data: serial port receiver data input (asynchronous) or data input/output (synchronous)
P3.1	TXD/clock: serial port transmitter data output (asynchronous) or clock output (synchronous)
P3.2	$\overline{INT0}$: external interrupt 0 or gate control input for timer/event counter 0
P3.3	$\overline{INT1}$: external interrupt 1 or gate control input for timer/event counter 1
P3.4	T0: external input for timer/event counter 0
P3.5	T1: external input for timer/event counter 1
P3.6	$\overline{WR}$: external data memory write strobe
P3.7	$\overline{RD}$: external data memory read strobe

The generation or use of a Port 3 pin as an alternative function is carried out automatically by the PCB83C654 provided the associated Special Function Register bit is set high. Port 3 can sink/source one TTL (= 4 LSTTL) input. It can drive CMOS inputs without external pull-ups.

18	XTAL 2	**Crystal input 2:** output of the inverting amplifier that forms the oscillator. Left open-circuit when an external oscillator clock is used (see Figs 10 and 11).
19	XTAL 1	**Crystal input 1:** input to the inverting amplifier that forms the oscillator, and input to the internal clock generator. Receives the external oscillator clock signal when an external oscillator is used (see Figs 10 and 11).
20	V_{SS}	**Ground:** circuit ground potential.
21-28	P2.0-P2.7	**Port 2:** 8-bit quasi-bidirectional I/O port with internal pull-ups. During access to external memories (RAM/ROM) that use 16-bit addresses (MOVX @DPTR) Port 2 emits the high order address byte. When external RAM is accessed with an 8-bit address (MOVX @Ri) Port 2 emits the contents of the P2 special function register. Port 2 can sink/source one TTL (= 4 LSTTL) input. It can drive CMOS inputs without external pull-ups.
29	$\overline{PSEN}$	**Program Store Enable output:** read strobe to the external program memory via port 0 and 2. It is activated twice each machine cycle during fetches from external program memory. When executing out of external program memory two activations of $\overline{PSEN}$ are skipped during each access to external data memory. $\overline{PSEN}$ is not activated (remains HIGH) during no fetches from external program memory. $\overline{PSEN}$ can sink/source 8 LSTTL inputs. It can drive CMOS inputs without an external pull-up.
30	ALE	**Address Latch Enable output:** latches the low byte of the address during accesses to external memory in normal operation. It is activated every six oscillator periods except during an external data memory access. ALE can sink/source 8 LSTTL inputs. It can drive CMOS inputs without an external pull-up.

PINNING (continued)

31	$\overline{EA}$	**External Access input:** When $\overline{EA}$ is held at a TTL high level the CPU executes out of the internal program ROM provided the program counter is less than 8192. When $\overline{EA}$ is held at a TTL low level, the CPU executes out of external program memory via port 0 and port 2. $\overline{EA}$ is not allowed to float.
32-39	P0.7-P0.0	**Port 0:** 8-bit open drain bidirectional I/O port. It is also the multiplexed low-order address and data bus during accesses to external memory (during these accesses it activates internal pull-ups). Port 0 can sink/source eight LSTTL inputs.
40	V_{DD}	**Power supply:** + 5 V power supply pin during normal operation, Idle mode and Power-down mode.

To avoid a 'latch-up' effect at power-on, the voltage on any pin at any time must not be higher or lower than V_{DD} + 0,5 V or V_{SS} − 0,5 V respectively.

FUNCTIONAL DESCRIPTION

General

The PCB83C654 is a stand-alone high-performance microcontroller designed for use in real-time applications such as instrumentation and industrial control.

The device provides, in addition to the 80C51 standard functions, a serial I^2C-bus interface. As well as the parallel bus, functions may also be expanded using the I^2C-bus utilizing the complete line of the I^2C clips family.

The PCB83C654 is a control-oriented CPU with on-chip program and data memory. It can be extended with external program memory up to 64 K bytes. It can also access up to 64 K bytes of external data memory. For systems requiring extra capability, the PCB83C654 can be expanded using standard memories and peripherals.

The PCB83C654 has two software selectable modes of reduced activity for further power reduction – Idle and Power-down. The Idle mode freezes the CPU while allowing the RAM, timers, serial ports and interrupt system to continue functioning. The Power-down mode saves the RAM contents but freezes the oscillator causing all other chip functions to be inoperative.

Memory organization

The central processing unit (CPU) manipulates operands in three memory spaces; these are the 64 K-byte external data memory, 256-byte internal data memory and the 64 K-byte internal and external program memory. The internal data memory address space is sub-divided into the 256-byte internal data RAM and 128-byte Special Function Register (SFR) address spaces, as shown in Fig. 4. Figure 5 shows the Special Function Register memory map. Internal RAM locations 0-127 are directly and indirectly addressable. Internal RAM locations 128-255 are only indirectly addressable. The special function register locations 128-255 are only directly addressable.

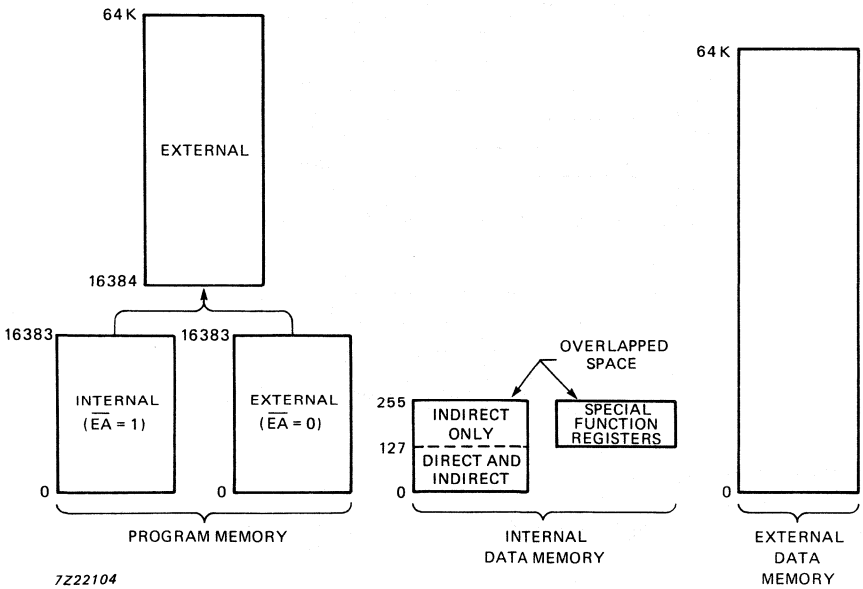
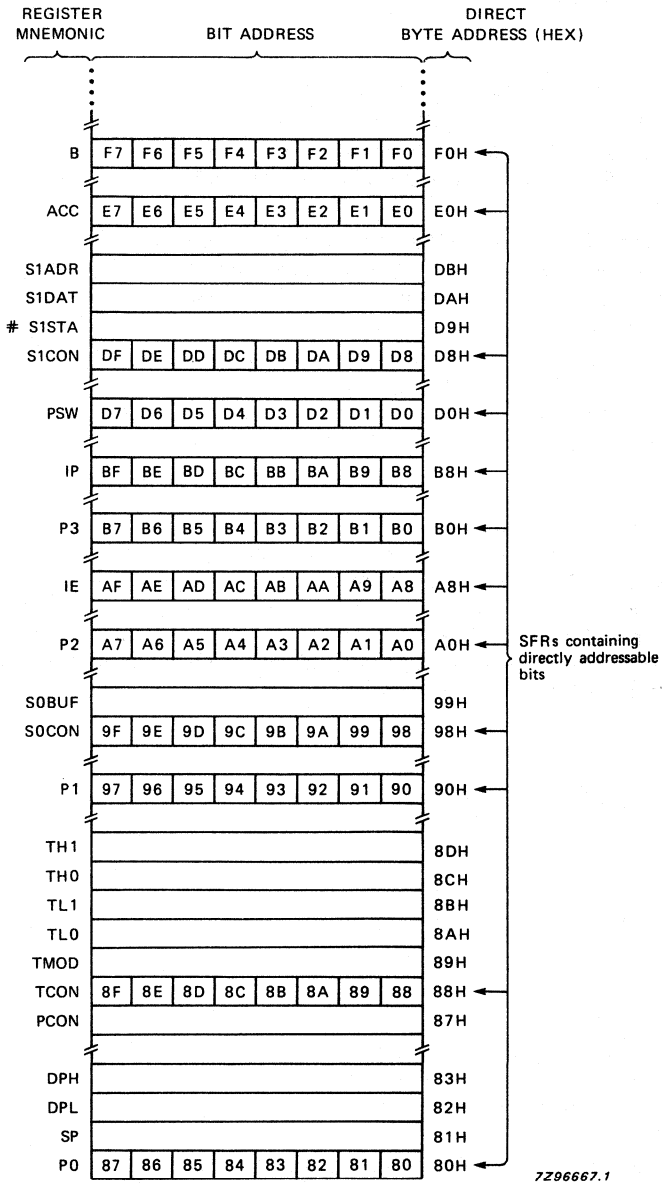


Fig. 4 Memory map.



S1STA is a read-only register.

Fig. 5 Special function registers memory map.

FUNCTIONAL DESCRIPTION (continued)

The internal data RAM contains four register banks (each with eight registers), 128 addressable bits, a scratchpad area and the stack. The stack depth is limited by the available internal data RAM and its location is determined by the 8-bit stack pointer. All registers except the program counter and the four 8-register banks reside in the special function register address space. These memory mapped registers include arithmetic registers, pointers, I/O ports, interrupt system registers, timers and serial port registers. There are 128 addressable bit locations in the SFR address space.

The PCB83C654 contains 256 bytes of internal data RAM and 25 special function registers. It provides a non-paged program memory address space to accommodate relocatable code. Conditional branches are performed relative to the program counter. The register-indirect jump permits branching relative to a 16-bit base register with an offset provided by an 8-bit index register. 16-bit jumps and calls permit branching to any location in the contiguous 64 K program memory address space.

Addressing

The PCB83C654 has five methods for addressing source operands:

- Register
- Direct
- Register-Indirect
- Immediate
- Base-Register plus Index-Register-Indirect

The first three methods can be used for addressing destination operands. Most instructions have a "destination/source" field that specifies the data type, addressing methods and operands involved. For operations other than MOVs, the destination operand is also a source operand.

Access to memory addressing is as follows:

- Registers in one of the four 8-register banks through Register, Direct, or Register-Indirect.
- 256 bytes of internal data RAM through Direct or Register-Indirect. Bytes 0-127 may be addressed directly/indirectly. Bytes 128-255 share their address locations with the SFR registers and so may only be addressed indirectly as data RAM.
- Special function registers through Direct at address locations 128-255.
- External data memory through Register-Indirect.
- Program memory look-up tables through Base-Register plus Index-Register-Indirect.

The PCB83C654 is classified as an 8-bit device since the internal ROM, RAM, Special Function Registers (SFR), Arithmetic Logic Unit (ALU) and external data bus are each 8-bits wide. It performs operations on bit, nibble, byte and double-byte data types.

Facilities are available for byte transfer, logic and integer arithmetic operations. Data transfer, logic and conditional branch operations can be performed directly on Boolean variables to provide excellent bit handling.

Instruction set

The PCB83C654 uses the powerful instruction set of the PCB80C51. Additional special function registers are incorporated to control the on-chip peripherals. The instruction set consists of 49 single-byte, 45 two-byte and 17 three-byte instructions. When using a 12 MHz oscillator, 64 instructions execute in 1 μs and 45 instructions execute in 2 μs. Multiply and divide instructions execute in 4 μs.

FUNCTIONAL DESCRIPTION (continued)

I/O facilities

The PCB83C654 has four 8-bit ports. Ports 0-3 are the same as in the 80C51, with the exception of the additional functions of port 1. Port lines P1.7 and P1.6 may be selected as the SDA and SCL lines of serial port SIO1 (I^2C). Because the I^2C bus may be active while the device is disconnected from V_{DD}, these pins are provided with open drain drivers.

N.B. Therefore pins P1.7 and P1.6 do not have pull-up devices when used as ports.

Ports 0, 1, 2 and 3 perform the following alternative functions:

- Port 0: provides the multiplexed low-order address and data bus used for expanding the PCB83C654 with standard memories and peripherals.
- Port 1: Port 1 is partly used for the I^2C bus functions;
  - SCL and SDA for the I^2C interface, P1.6 and P1.7 respectively.

Pins whose alternate function is not used may be used as normal bidirectional I/O pins.

- Port 2: provides the high-order address bus when expanding the PCB83C654 with external program memory and/or external data memory.
- Port 3: pins can be configured individually to provide:
  - external interrupt request inputs
  - counter inputs
  - serial port receiver input and transmitter output
  - control signals to READ and WRITE external data memory

A Port 3 pin automatically assumes its alternative function when the associated Special Function Register bit is set high.

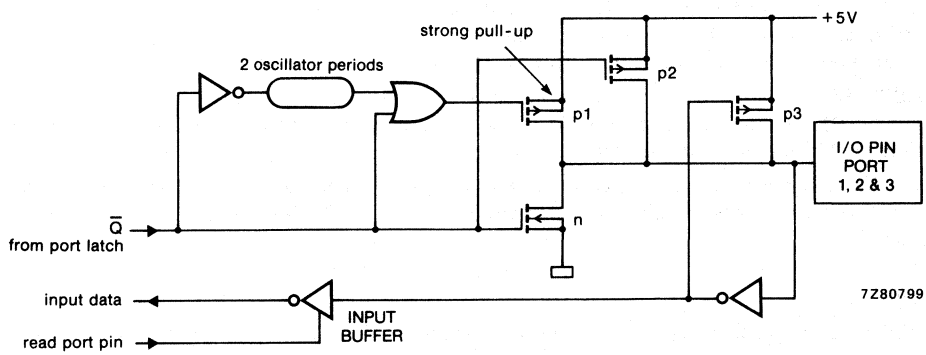


Fig. 6 I/O buffers in the PCB83C654 (Ports 2, 3, and P1.0 to P1.5).

Timer/event counters

The PCB83C654 contains two 16-bit timer/event counters: Timer 0 and Timer 1. Timer 0 and Timer 1 may be programmed to carry out the following functions:

- Measure time intervals and pulse durations
- Count events
- Generate interrupt requests.

Timer 0 and Timer 1 can be programmed independently to operate in three modes:

- Mode 0: 8-bit timer or 8-bit counter each with divide by 32 prescaler
- Mode 1: 16-bit time-interval or event counter
- Mode 2: 8-bit time-interval or event counter with automatic reload upon overflow.

Timer 0 can be programmed to operate in an additional mode as follows:

- Mode 3: one 8-bit time-interval or event counter and one 8-bit time-interval counter.

When Timer 0 is in Mode 3, Timer 1 can be programmed to operate in Modes 0, 1 or 2 but cannot set an interrupt request flag or generate an interrupt. However the overflow from Timer 1 can be used to pulse the serial port transmission-rate generator.

The frequency handling range of these counters with a 12 MHz crystal is as follows:

- In the timer function, the timer is incremented at a frequency of 1 MHz – a division by 12 of the oscillator frequency
- 0 Hz to an upper limit of 0,5 MHz when programmed for external inputs.

Both internal and external inputs can be gated to the counter by a second external source for directly measuring pulse durations.

The counters are started and stopped under software control. Each one sets its interrupt request flag when it overflows from all logic 1s to all logic 0s (or automatic reload value), with the exception of mode 3 as previously described.

Serial I/O (see Fig. 7)

The PCB83C654 is equipped with two independent serial ports. SIO0 is the full duplex UART port and is identical to the serial port of the PCB80C51.

Serial port SIO1 supports the I^2C-bus, the function of which is controlled by the S1CON register. S1STA is the status register whose contents may also be used as a vector to various service routines. S1DAT is the data shift register and S1ADR the slave address register. The least significant bit of S1ADR enables/disables general call address recognition.

FUNCTIONAL DESCRIPTION (continued)

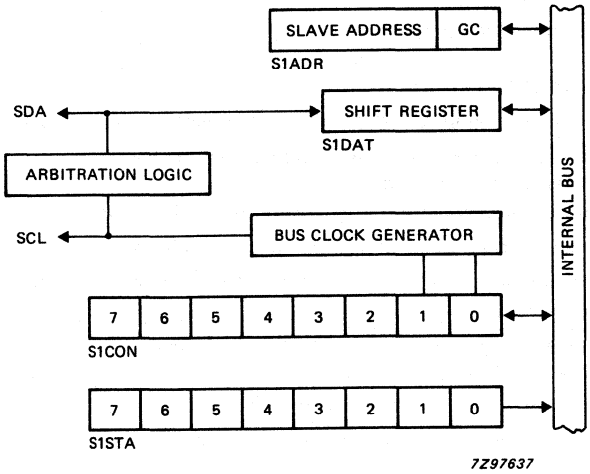


Fig. 7 Block diagram of I^2C serial I/O.

The I^2C serial I/O has complete autonomy in byte handling and operates in 4 modes:

1. Master transmitter
2. Master receiver
3. Slave transmitter
4. Slave receiver

Slave address recognition is performed by on-chip hardware.

The I^2C-bus consists of two lines: a data line (SDA) and a clock line (SCL). These lines also function as the I/O port lines on P1.7 and P1.6. The system is unique because data transport, clock generation, address recognition and bus control arbitration are all controlled by hardware.

Serial control register S1CON

	7	6	5	4	3	2	1	0
S1CON(D8H)	X	ENS1	STA	STO	SI	AA	CR1	CR0

Bits CR1 and CR0 determine the clock frequency that is generated in the master mode of operation. Table 1 displays the clock rate when using a 12 MHz crystal.

Table 1 Clock rate when using a 12 MHz crystal

CR1 / CR0	bit frequency	f_{osc} divided by
0 0	12,5 kHz	960
0 1	100 kHz	120
1 0	200 kHz	60 ($f_{osc} < 6$ MHz meeting I^2C)
1 1	62,5–0,5 kHz	96 x (256–reload value Timer 1) (reload value range: 0-254 in mode 2)

AA

Assert acknowledge bit. When this bit is set, an acknowledge is returned after any one of the following conditions:

- Own slave address is received
- General call address is received (S1ADR.0 = logic 1)
- A data byte is received, while the device is programmed to be a master receiver
- A data byte is received, while the device is a selected slave receiver.

When this bit is reset, no acknowledge is returned. Consequently, no interrupt is requested when the own address or general call address is received.

SI

SIO1 interrupt flag. This flag is set by hardware and an interrupt request is generated after any of the following events occur:

- A START condition is generated in MST mode
- The own slave address has been received during AA = logic 1
- The general call address has been received while S1ADR.0 and AA = logic 1
- A data byte has been received or transmitted in MST mode (even if arbitration is lost)
- A data byte has been received or transmitted as selected slave
- A STOP or START condition is received as selected slave receiver or transmitter.

While the SI flag is set, SCL remains LOW and the serial transfer is suspended.
SI must be reset by software.

STO

STOP flag. When in master mode,and this bit is set a STOP condition is generated. A STOP condition detected on the I^2C-bus clears this bit. This bit may also be set in slave mode in order to recover from an error condition. Then no STOP condition is generated to the I^2C-bus, but the hardware releases the SDA and SCL lines and switches to the not selected slave receiver mode. The STOP flag is cleared by the hardware.

STA

START flag. When this bit is set, the hardware checks the I^2C-bus and generates a START condition if the bus is free. If the device is already programmed as either master/transmitter or master/receiver, it will generate a repeated START condition.

ENS1

0 = Serial I/O Disabled and reset.Output ports P1.6 and P1.7 function with open drain

1 = Serial I/O Enabled. Output ports P1.6 and P1.7 must be set to logic 1.

FUNCTIONAL DESCRIPTION (continued)

Serial status register S1STA (S1STA is a read-only register)

	7	6	5	4	3	2	1	0
S1STA(D9H)	SC4	SC3	SC2	SC1	SC0	0	0	0

S1STA.3 - S1STA.7 hold a status code. S1STA.0 - S1STA.2 are held LOW. The contents of the status register may be used as a vector to a service routine. This optimizes the response time of the software and consequently that of the I^2C-bus.

Abbreviations used:

SLA : 7-bit slave address
R : Read bit
W : Write bit
ACK : Acknowledgement (acknowledge bit = logic 0)
$\overline{ACK}$: Not acknowledgement (acknowledge bit = logic 1)
DATA : 8-bit data byte to or from I^2C bus
MST : Master
SLV : Slave
TRX : Transmitter
REC : Receiver

The following is a list of the status codes:

MST/TRX mode

S1STA value

08H - A START condition has been transmitted
10H - A repeated START condition has been transmitted
18H - SLA and W have been transmitted, ACK has been received
20H - SLA and W have been transmitted, $\overline{ACK}$ received
28H - DATA of S1DAT has been transmitted, ACK received
30H - DATA of S1DAT has been transmitted, $\overline{ACK}$ received
38H - Arbitration lost in SLA, R/W or DATA

MST/REC mode

S1STA value

38H - Arbitration lost while returning $\overline{ACK}$
40H - SLA and R have been transmitted, ACK received
48H - SLA and R have been transmitted, $\overline{ACK}$ received
50H - DATA has been received, ACK returned
58H - DATA has been received, $\overline{ACK}$ returned

SLV/REC mode

S1STA value

60H - Own SLA and W have been received, ACK returned
68H - Arbitration lost in SLA, R/W as MST. Own SLA and W have been received, ACK returned
70H - General CALL has been received, ACK returned
78H - Arbitration lost in SLA, R/W as MST. General call has been received
80H - Previously addressed with own SLA. DATA byte received, ACK returned
88H - Previously addressed with own SLA. DATA byte received, $\overline{\text{ACK}}$ returned
90H - Previously addressed with general call. DATA byte has been received, ACK has been returned
98H - Previously addressed with general call. DATA byte has been received, $\overline{\text{ACK}}$ has been returned
A0H - A STOP condition or repeated START condition has been received while still addressed as SLV/REC or SLV/TRX.

SLV/TRX mode

S1STA value

A8H - Own SLA and R have been received, ACK returned
B0H - Arbitration lost in SLA, R/W as MST. Own SLA and R have been received, ACK returned
B8H - DATA byte has been transmitted, ACK received
C0H - DATA byte has been transmitted, $\overline{\text{ACK}}$ received
C8H - Last DATA byte has been transmitted (AA = logic 0), ACK received.

Miscellaneous

S1STA value

00H - Bus error during MST mode or selected SLV mode, due to an erroneous START or STOP condition

The data shift register S1DAT

S1DAT(DAH)	7	6	5	4	3	2	1	0

This register contains the serial data to be transmitted or data which has just been received. Bit 7 is transmitted or received first; i.e. data is shifted from right to left.

Address register S1ADR

	slave address							GC
S1ADR (DBH)	7	6	5	4	3	2	1	0

S1ADR.0, GC : 0 = general call address is not recognized
1 = general call address is recognized

S1ADR.7-1 : own slave address

This 8-bit register may be loaded with the 7-bit slave address to which the controller will respond when programmed as a slave receiver/transmitter. The LSB (GC) is used to determine whether the general call address is recognized.

FUNCTIONAL DESCRIPTION (continued)

Idle and Power-down operation (see Fig. 8)

Idle mode operation permits the interrupt, serial ports and timer blocks to continue to function while the CPU is halted.

The following functions remain active during Idle mode. These functions may generate an interrupt or reset and thus end the Idle mode:

Timer 0, Timer 1
SIO0, SIO1
External interrupt

The Power-down operation freezes the oscillator. The Power-down mode can only be activated by setting the PD bit in the PCON register.

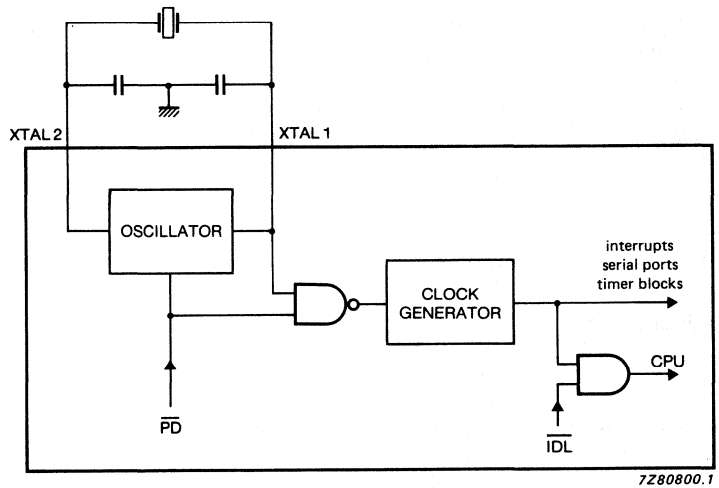


Fig. 8 Internal Idle and Power-down clock configuration.

Power control register

These special modes are activated by software via the Special Function Register PCON. Its hardware address is 87H. PCON is not bit addressable.

	7	6	5	4	3	2	1	0
PCON(87H)	SMOD	–	–	–	GF1	GF0	PD	IDL

Bit	Symbol	Function
PCON.7	SMOD	Double Baud rate bit. When set to logic 1 the baud rate is doubled when the serial port SIO0 is being used in modes 1, 2 or 3
PCON.6	–	(reserved)
PCON.5	–	(reserved)
PCON.4	–	(reserved)
PCON.3	GF1	General-purpose flag bit
PCON.2	GF0	General-purpose flag bit
PCON.1	PD	Power-down bit. Setting this bit activates Power-down mode
PCON.0	IDL	Idle mode bit. Setting this bit activates the Idle mode.

If logic 1s are written to PD and IDL at the same time, PD takes precedence. The reset value of PCON is 0XXX0000.

Idle mode

The instruction that sets PCON.0 is the last instruction executed in the normal operating mode before Idle mode is activated. Once in the Idle mode, the CPU status is preserved in its entirety: the Stack Pointer, Program Counter, Program Status Word, Accumulator, RAM and all other registers maintain their data during Idle mode. The status of the external pins during Idle mode is shown in Table 2.

There are two ways to terminate the Idle mode:

Activation of any enabled interrupt will cause PCON.0 to be cleared by hardware terminating Idle mode. The interrupt is serviced, and following return from interrupt instruction RETI, the next instruction to be executed will be the one which follows the instruction that wrote a logic 1 to PCON.0.

The flag bits GF0 and GF1 may be used to determine whether the interrupt was received during normal execution or during the Idle mode. For example, the instruction that writes to PCON.0 can also set or clear one or both flag bits. When Idle mode is terminated by an interrupt, the service routine can examine the status of the flags bits.

The second way of terminating the Idle mode is with an external hardware reset. Since the oscillator is still running, the hardware reset is required to be active for two machine cycles (24 oscillator periods) to complete the reset operation.

Power-down mode

The instruction that sets PCON.1 is the last executed prior to going into the Power-down mode. Once in Power-down mode, the oscillator is stopped. Only the contents of the on-chip RAM are preserved. The Special Function Registers are not saved. A hardware reset is the only way of exiting the Power-down mode.

In the Power-down mode, V_{DD} may be reduced to minimize circuit power consumption. The voltage must not be reduced until the Power-down mode is entered, but must be restored before the hardware reset is applied which will free the oscillator. Reset should not be released until the oscillator has restarted and stabilized.

The status of the external pins during Power-down mode is shown in Table 2. If the Power-down mode is activated while in external program memory, the port data that is held in the Special Function Register P2 is restored to port 2. If the data is a logic 1, the port pin is held HIGH during the Power-down mode by the strong pull-up transistor p1 (see Fig. 6).

FUNCTIONAL DESCRIPTION (continued)

Table 2 Status of the external pins during Idle and Power-down modes.

mode	memory	ALE	$\overline{PSEN}$	Port 0	Port 1	Port 2	Port 3
Idle	internal	1	1	port data	port data	port data	port data
Idle	external	1	1	floating	port data	address	port data
Power-down	internal	0	0	port data	port data	port data	port data
Power-down	external	0	0	floating	port data	port data	port data

Note: Ports 1.7 and 1.6 if selected, function as SDA and SCL respectively in the Idle mode.

Interrupt system (see Fig. 9)

External events and the real-time-driven on-chip peripherals require service by the CPU asynchronous to the execution of any particular section of code. To tie the asynchronous activities of these functions to normal program execution a multiple-source, two-priority-level, nested interrupt system is provided. Interrupt response latency is from 3 μs to 8 μs when using a 12 MHz crystal. The PCB83C654 acknowledges interrupt requests from six sources as follows:

- $\overline{INT0}$ and $\overline{INT1}$: two external interrupts
- Timer 0 and Timer 1: two internal counters
- I^2C serial I/O interrupt
- UART serial I/O port interrupt

Each interrupt vectors to a separate location in program memory for its service program. Each source can be individually enabled or disabled by a corresponding bit in the IE register, moreover each interrupt may be programmed to a high or low priority level using a corresponding bit in the IP register. Also all enabled sources can be globally disabled or enabled. Both external interrupts can be programmed to be level-activated or transition-activated; an active LOW level allows "wire-ORing" of several interrupt sources to the input pin.

DEVELOPMENT DATA

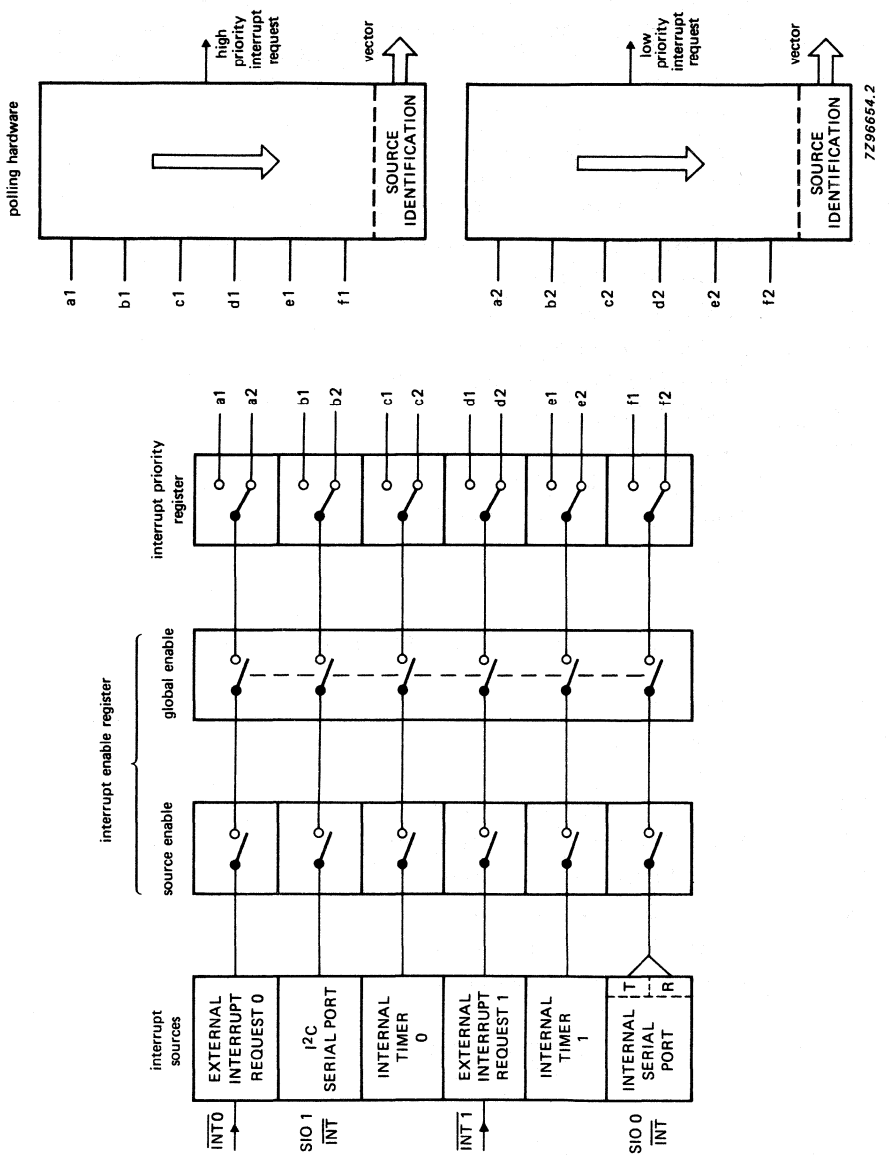


Fig. 9 Interrupt system.

FUNCTIONAL DESCRIPTION (continued)

Interrupt enable register

	7	6	5	4	3	2	1	0
IE(A8H)	EA	–	ES1	ES0	ET1	EX1	ET0	EX0

Bit	Symbol	Function
IE.7	EA	General enable/disable control 0 = No interrupt is enabled 1 = Any individually enabled interrupt will be accepted
IE.6	–	Unused
IE.5	ES1	Enable SIO1 (I^2C) interrupt
IE.4	ES0	Enable SIO0 (UART) interrupt
IE.3	ET1	Enable Timer 1 interrupt
IE.2	EX1	Enable External 1 interrupt
IE.1	ET0	Enable Timer 0 interrupt
IE.0	EX0	Enable External 0 interrupt
		where "0" = interrupt disabled and "1" = interrupt enabled

Interrupt priority register

	7	6	5	4	3	2	1	0
IP(B8H)	–	–	PS1	PS0	PT1	PX1	PT0	PX0

Bit	Symbol	Function
IP.7	–	Unused
IP.6	–	Unused
IP.5	PS1	SIO1 (I^2C) interrupt priority level
IP.4	PS0	SIO0 (UART) interrupt priority level
IP.3	PT1	Timer 1 interrupt priority level
IP.2	PX1	External interrupt 1 priority level
IP.1	PT0	Timer 0 interrupt priority level
IP.0	PX0	External interrupt 0 priority level

Interrupt priority levels are as follows: "0" – low priority
"1" – high priority

Table 3 shows the interrupt vectors. The vector indicates the ROM location where the appropriate interrupt service routine starts.

Table 3 Interrupt vectors

Source		Vector
External 0	X0	0003H
Timer 0 overflow	T0	000BH
External 1	X1	0013H
Timer 1 overflow	T1	001BH
Serial I/O 0 (UART)	S0	0023H
Serial I/O 1 (I^2C)	S1	002BH

Interrupt priority

Each interrupt source can be either high priority or low priority. If both priorities are requested simultaneously, the processor will branch to the high priority vector. If there are simultaneous requests from sources of the same priority, then interrupts will be serviced in the following order:

X0, S1, T0, X1, T1, S0

A low priority interrupt routine can only be interrupted by high priority interrupt. A high priority interrupt routine can not be interrupted.

Oscillator circuitry

The oscillator circuitry of the PCB83C654 is a single-stage inverting amplifier in a Pierce oscillator configuration. The circuitry between XTAL 1 and XTAL 2 is basically an inverter biased to the transfer point. Either a crystal or ceramic resonator can be used as the feedback element to complete the oscillator circuitry. Both are operated in parallel resonance. XTAL 1 is the high gain amplifier input, and XTAL 2 is the output (see Fig. 10). To drive the PCB83C654 externally, XTAL 1 is driven from an external source and XTAL 2 left open-circuit (see Fig. 11).

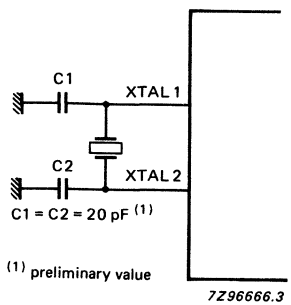


Fig. 10 PCB83C654 oscillator circuit.

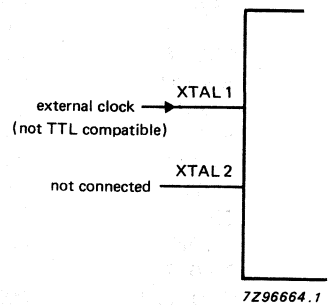


Fig. 11 Driving the PCB83C654 from an external source.

FUNCTIONAL DESCRIPTION (continued)

Reset circuitry (see Fig. 12)

The reset circuitry for the PCB83C654 is connected to the reset pin RST. A Schmitt trigger is used at the input for noise rejection. The output of the Schmitt trigger is sampled by the reset circuitry every machine cycle.

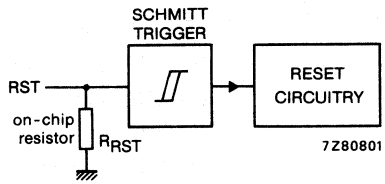


Fig. 12 On chip reset configuration.

A reset is accomplished by holding the RST pin HIGH for at least two machine cycles (24 oscillator periods). The CPU responds by executing an internal reset. During reset ALE and $\overline{PSEN}$ output a HIGH level. In order to perform a correct reset, this level must not be affected by external elements.

The internal reset is executed during the second cycle in which RST is HIGH and is repeated every cycle until RST goes LOW. It leaves the internal registers as follows:

Register	Content
ACC	0000 0000
B	0000 0000
DPL	0000 0000
DPH	0000 0000
IE	0x00 0000
IP	xx00 0000
PCH	0000 0000
PCL	0000 0000
PCON	0xxx 0000
PSW	0000 0000
P0 - P3	1111 1111
S0BUF	xxxx xxxx
S0CON	0000 0000
S1ADR	0000 0000
S1CON	x000 0000
S1DAT	0000 0000
S1STA	1111 1000
SP	0000 0111
TCON	0000 0000
TH0, TH1	0000 0000
TL0, TL1	0000 0000
TMOD	0000 0000

The internal RAM is not affected by reset. When V_{DD} is turned on, the RAM content is indeterminate.

Power-on reset (see Fig. 13)

When V_{DD} is turned on an automatic reset can be obtained by connecting the RST pin to V_{DD} via a 2,2 μF capacitor. When the power is switched on, the voltage on the RST pin is equal to V_{DD} minus the capacitor voltage, and decreases from V_{DD} as the capacitor charges through the internal resistor (R_{RST}) to ground. The larger the capacitor, the more slowly V_{RST} decreases. V_{RST} must remain above the lower threshold of the Schmitt trigger long enough to effect a complete reset. The time required is the oscillator start-up time, plus 2 machine cycles.

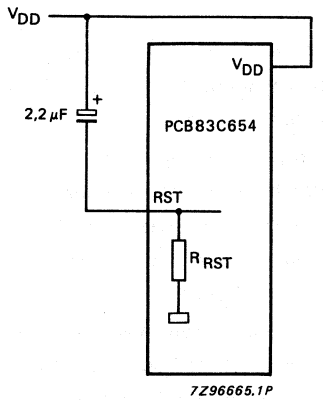


Fig. 13 Power-on reset.

INSTRUCTION SET

Table 4 Instruction set description

mnemonic		description	bytes/ cycles		opcode (hex.)
Arithmetic operations					
ADD	A,Rr	Add register to A	1	1	2*
ADD	A,direct	Add direct byte to A	2	1	25
ADD	A,@Ri	Add indirect RAM to A	1	1	26, 27
ADD	A,#data	Add immediate data to A	2	1	24
ADDC	A,Rr	Add register to A with carry flag	1	1	3*
ADDC	A,direct	Add direct byte to A with carry flag	2	1	35
ADDC	A,@Ri	Add indirect RAM to A with carry flag	1	1	36, 37
ADDC	A,#data	Add immediate data to A with carry flag	2	1	34
SUBB	A,Rr	Subtract register from A with borrow	1	1	9*
SUBB	A,direct	Subtract direct byte from A with borrow	2	1	95
SUBB	A,@Ri	Subtract indirect RAM from A with borrow	1	1	96, 97
SUBB	A,#data	Subtract immediate data from A with borrow	2	1	94
INC	A	Increment A	1	1	04
INC	Rr	Increment register	1	1	0*
INC	direct	Increment direct byte	2	1	05
INC	@Ri	Increment indirect RAM	1	1	06, 07
DEC	A	Decrement A	1	1	14
DEC	Rr	Decrement register	1	1	1*
DEC	direct	Decrement direct byte	2	1	15
DEC	@Ri	Decrement indirect RAM	1	1	16, 17
INC	DPTR	Increment data pointer	1	2	A3
MUL	AB	Multiply A & B	1	4	A4
DIV	AB	Divide A by B	1	4	84
DA	A	Decimal adjust A	1	1	D4

mnemonic		description	bytes/ cycles		opcode (hex.)
Logic operations					
ANL	A,Rr	AND register to A	1	1	5*
ANL	A,direct	AND direct byte to A	2	1	55
ANL	A,@Ri	AND indirect RAM to A	1	1	56, 57
ANL	A,#data	AND immediate data to A	2	1	54
ANL	direct,A	AND A to direct byte	2	1	52
ANL	direct,#data	AND immediate data to direct byte	3	2	53
ORL	A,Rr	OR register to A	1	1	4*
ORL	A,direct	OR direct byte to A	2	1	45
ORL	A,@Ri	OR indirect RAM to A	1	1	46, 47
ORL	A,#data	OR immediate data to A	2	1	44
ORL	direct,A	OR A to direct byte	2	1	42
ORL	direct,#data	OR immediate data to direct byte	3	2	43
XRL	A,Rr	Exclusive-OR register to A	1	1	6*
XRL	A,direct	Exclusive-OR direct byte to A	2	1	65
XRL	A,@Ri	Exclusive-OR indirect RAM to A	1	1	66, 67
XRL	A,#data	Exclusive-OR immediate data to A	2	1	64
XRL	direct,A	Exclusive-OR A to direct byte	2	1	62
XRL	direct,#data	Exclusive-OR immediate data to direct byte	3	2	63
CLR	A	Clear A	1	1	E4
CPL	A	Complement A	1	1	F4
RL	A	Rotate A left	1	1	23
RLC	A	Rotate A left through the carry flag	1	1	33
RR	A	Rotate A right	1	1	03
RRC	A	Rotate A right through the carry flag	1	1	13
SWAP	A	Swap nibbles within A	1	1	C4

INSTRUCTION SET (continued)

mnemonic	description	bytes/ cycles		opcode (hex.)
Data transfer				
MOV A,Rr	Move register to A	1	1	E*
MOV A,direct (**)	Move direct byte to A	2	1	E5
MOV A,@Ri	Move indirect RAM to A	1	1	E6, E7
MOV A,#data	Move immediate data to A	2	1	74
MOV Rr,A	Move A to register	1	1	F*
MOV Rr,direct	Move direct byte to register	2	2	A*
MOV Rr,#data	Move immediate data to register	2	1	7*
MOV direct,A	Move A to direct byte	2	1	F5
MOV direct,Rr	Move register to direct byte	2	2	8*
MOV direct,direct	Move direct byte to direct byte	3	2	85
MOV direct,@Ri	Move indirect RAM to direct byte	2	2	86, 87
MOV direct,#data	Move immediate data to direct byte	3	2	75
MOV @Ri,A	Move A to indirect RAM	1	1	F6, F7
MOV @Ri,direct	Move direct byte to indirect RAM	2	2	A6, A7
MOV @Ri,#data	Move immediate data to indirect RAM	2	1	76, 77
MOV DPTR,#data 16	Load data pointer with a 16-bit constant	3	2	90
MOVC A,@A+DPTR	Move code byte relative to DPTR to A	1	2	93
MOVC A,@A+PC	Move code byte relative to PC to A	1	2	83
MOVX A,@Ri	Move external RAM (8-bit address) to A	1	2	E2, E3
MOVX A,@DPTR	Move external RAM (16-bit address) to A	1	2	E0
MOVX @Ri,A	Move A to external RAM (8-bit address)	1	2	F2, F3
MOVX @DPTR,A	Move A to external RAM (16-bit address)	1	2	F0
PUSH direct	Push direct byte onto stack	2	2	C0
POP direct	Pop direct byte from stack	2	2	D0
XCH A,Rr	Exchange register with A	1	1	C*
XCH A,direct	Exchange direct byte with A	2	1	C5
XCH A,@Ri	Exchange indirect RAM with A	1	1	C6, C7
XCHD A,@Ri	Exchange LOW-order nibble indirect RAM with A	1	1	D6, D7

** MOV A,ACC is not permitted.

mnemonic		description	bytes/ cycles		opcode (hex.)
Boolean variable manipulation					
CLR	C	Clear carry flag	1	1	C3
CLR	bit	Clear direct bit	2	1	C2
SETB	C	Set carry flag	1	1	D3
SETB	bit	Set direct bit	2	1	D2
CPL	C	Complement carry flag	1	1	B3
CPL	bit	Complement direct bit	2	1	B2
ANL	C,bit	AND direct bit to carry flag	2	2	82
ANL	C,/bit	AND complement of direct bit to carry flag	2	2	B0
ORL	C,bit	OR direct bit to carry flag	2	2	72
ORL	C,/bit	OR complement of direct bit to carry flag	2	2	A0
MOV	C,bit	Move direct bit to carry flag	2	1	A2
MOV	bit,C	Move carry flag to direct bit	2	2	92
Program and machine control					
ACALL	addr11	Absolute subroutine call	2	2	●1addr
LCALL	addr16	Long subroutine call	3	2	12
RET		Return from subroutine	1	2	22
RETI		Return from interrupt	1	2	32
AJMP	addr11	Absolute jump	2	2	▲1addr
LJMP	addr16	Long jump	3	2	02
SJMP	rel	Short jump (relative address)	2	2	80
JMP	@A+DPTR	Jump indirect relative to the DPTR	1	2	73
JZ	rel	Jump if A is zero	2	2	60
JNZ	rel	Jump if A is not zero	2	2	70
JC	rel	Jump if carry flag is set	2	2	40
JNC	rel	Jump if carry flag is not set	2	2	50
JB	bit,rel	Jump if direct bit is set	3	2	20
JNB	bit,rel	Jump if direct bit is not set	3	2	30
JBC	bit,rel	Jump if direct bit is set and clear bit	3	2	10
CJNE	A,direct,rel	Compare direct to A and jump if not equal	3	2	B5
CJNE	A,#data,rel	Compare immediate to A and jump if not equal	3	2	B4
CJNE	Rr,#data,rel	Compare immed. to reg. and jump if not equal	3	2	B*
CJNE	@Ri,#data,rel	Compare immed. to ind. and jump if not equal	3	2	B6, B7
DJNZ	Rr,rel	Decrement register and jump if not zero	2	2	D*
DJNZ	direct,rel	Decrement direct and jump if not zero	3	2	D5
NOP		No operation	1	1	00

Notes to Table 4

Data addressing modes

Rr	Working register R0-R7.
direct	128 internal RAM locations and any special function register (SFR).
@Ri	Indirect internal RAM location addressed by register R0 or R1 of the actual register bank.
#data	8-bit constant included in instruction.
#data16	16-bit constant included as bytes 2 and 3 of instruction.
bit	direct addressed bit in internal RAM or SFR.
addr16	16-bit destination address. Used by LCALL and LJMP. The branch will be anywhere within the 64 K-byte program memory address space.
addr11	11-bit destination address. Used by ACALL and AJMP. The branch will be within the same 2 K-byte page of program memory as the first byte of the following instruction.
rel	Signed (two's complement) 8-bit offset byte. Used by SJMP and all conditional jumps. Range is −128 to + 127 bytes relative to first byte of the following instruction.

Hexadecimal opcode cross-reference to Table 5

* : 8, 9, A, B, C, D, E, F.
● : 11, 31, 51, 71, 91, B1, D1, F1.
▲ : 01, 21, 41, 61, 81, A1, C1, E1.

Table 5 Instruction map

first hexadecimal chacter of opcode ↓ / second hexadecimal character of opcode →

	0	1	2	3	4	5	6 | 7	8 | 9 | A	B	C	D	E	F
0	NOP	AJMP addr11	LJMP addr16	RR A	INC A	INC dir	INC @Ri 0 | 1	INC Rr 0 | 1 | 2	3	4	5	6	7
1	JBC bit,rel	ACALL addr11	LCALL addr16	RRC A	DEC A	DEC dir	DEC @Ri 0 | 1	DEC Rr 0 | 1 | 2	3	4	5	6	7
2	JB bit,rel	AJMP addr11	RET	RL A	ADD A,#data	ADD A,dir	ADD A,@Ri 0 | 1	ADD A,Rr 0 | 1 | 2	3	4	5	6	7
3	JNB bit,rel	ACALL addr11	RETI	RLC A	ADDC A,#data	ADDC A,dir	ADDC A,@Ri 0 | 1	ADDC A,Rr 0 | 1 | 2	3	4	5	6	7
4	JC rel	AJMP addr11	ORL dir,A	ORL dir,#data	ORL A,#data	ORL A,dir	ORL A,@Ri 0 | 1	ORL A,Rr 0 | 1 | 2	3	4	5	6	7
5	JNC rel	ACALL addr11	ANL dir,A	ANL dir,#data	ANL A,#data	ANL A,dir	ANL A,@Ri 0 | 1	ANL A,Rr 0 | 1 | 2	3	4	5	6	7
6	JZ rel	AJMP addr11	XRL dir,A	XRL dir,#data	XRL A,#data	XRL A,dir	XRL A,@Ri 0 | 1	XRL A,Rr 0 | 1 | 2	3	4	5	6	7
7	JNZ rel	ACALL addr11	ORL C,bit	JMP @A+DPTR	MOV A,#data	MOV dir,#data	MOV @Ri,#data 0 | 1	MOV Rr,#data 0 | 1 | 2	3	4	5	6	7
8	SJMP rel	AJMP addr11	ANL C,bit	MOVC A,@A+PC	DIV AB	MOV dir,dir	MOV dir,@Ri 0 | 1	MOV dir,Rr 0 | 1 | 2	3	4	5	6	7
9	MOV DPTR, #data	ACALL addr11	MOV bit,C	MOVC A,@A+DPTR	SUBB A,#data	SUBB A,dir	SUBB A,@Ri 0 | 1	SUBB A,Rr 0 | 1 | 2	3	4	5	6	7
A	ORL C,/bit	AJMP addr11	MOV C,bit	INC DPTR	MUL AB		MOV @Ri,dir 0 | 1	MOV Rr,dir 0 | 1 | 2	3	4	5	6	7
B	ANL C,/bit	ACALL addr11	CPL bit	CPL C	CJNE A, #data,rel	CJNE A,dir,rel	CJNE @Ri,#data,rel 0 | 1	CJNE Rr,#data,rel 0 | 1 | 2	3	4	5	6	7
C	PUSH dir	AJMP addr11	CLR bit	CLR C	SWAP A	XCH A,dir	XCH A,@Ri 0 | 1	XCH A,Rr 0 | 1 | 2	3	4	5	6	7
D	POP dir	ACALL addr11	SETB bit	SETB C	DA A	DJNZ dir,rel	XCHD A,@Ri 0 | 1	DJNZ Rr,rel 0 | 1 | 2	3	4	5	6	7
E	MOVX A,@DPTR	AJMP addr11	MOVX A,@Ri 0 | 1		CLR A	MOV * A,dir	MOV A,@Ri 0 | 1	MOV A,Rr 0 | 1 | 2	3	4	5	6	7
F	MOVX @DPTR,A	ACALL addr11	MOVX @Ri,A 0 | 1		CPL A	MOV dir,A	MOV @Ri,A 0 | 1	MOV Rr,A 0 | 1 | 2	3	4	5	6	7

* MOV A,ACC is not a valid instruction.

RATINGS

Limiting values in accordance with the Absolute Maximum System (IEC 134)

parameter	symbol	min.	max.	unit
Input voltage on any pin with respect to ground (V_{SS})	V_I	−0,5	+ 6,5	V
Input, output current	I_I, I_O	−	± 5,0	mA
Input, output current on any single pin	I_I, I_O	−	± 10	mA
Total power dissipation	P_{tot}	−	1,0	W
Storage temperature range	T_{stg}	−65	+ 150	°C
Operating ambient temperature range				
PCB83C654	T_{amb}	0	+ 70	°C
PCF83C654	T_{amb}	−40	+ 85	°C
PCA83C654	T_{amb}	−40	+ 125	°C

DEVELOPMENT DATA

DC CHARACTERISTICS

V_{DD} = 5 V (± 10%); V_{SS} = 0 V; T_{amb} = 0 to 70 °C (PCB83C654), −40 to + 85 °C (PCF83C654).
All voltages measured with respect to V_{SS} unless otherwise specified.

parameter	conditions	symbol	min.	max.	unit
Supply voltage		V_{DD}	4,5	5,5	V
Supply current operating	(note 1) f_{CLK} = 12 MHz; V_{DD} = 5,5 V	I_{DD}	–	tbf	mA
Idle mode	(note 2) f_{CLK} = 12 MHz; V_{DD} = 5 V ± 10%	I_{ID}	–	tbf	mA
Power-down current	2 V < V_{PD} < V_{DD} max. (note 3)	I_{PD}	–	tbf	μA
Inputs					
LOW level input voltage (except $\overline{EA}$, P1.6/SCL, P1.7/SDA)		V_{IL}	−0,5	$0{,}2V_{DD}-0{,}1$	V
LOW level input voltage ($\overline{EA}$)		V_{IL1}	−0,5	$0{,}2V_{DD}-0{,}3$	V
LOW level input voltage (P1.6/SCL, P1.7/SDA)	(note 6)	V_{IL2}	−0,5	$0{,}3V_{DD}$	V
HIGH level input voltage (except RST, XTAL1, P1.6/SCL, P1.7/SDA)		V_{IH}	$0{,}2V_{DD}+0{,}9$	$V_{DD}+0{,}5$	V
HIGH level input voltage (RST and XTAL1)		V_{IH1}	$0{,}7V_{DD}$	$V_{DD}+0{,}5$	V
HIGH level input voltage (P1.6/SCL, P1.7/SDA)	(note 6)	V_{IH2}	$0{,}7V_{DD}$	6,0	V
Input current logic 0 (Ports 1, 2 and 3; except P1.6/SDA, P1.7/SCL)	V_I = 0,45 V	$-I_{IL}$	–	50	μA
Input current logic 1 to 0 transition (Ports 1, 2, 3; except P1.6/SCL, P1.7/SDA)	V_I = 2,0 V	$-I_{TL}$	–	650	μA
Input leakage current (Port 0, $\overline{EA}$)	V_I = 0,45 V	$\pm I_{LI1}$	–	10	μA
Input leakage current (P1.6/SCL, P1.7/SDA)	0 V < V_I < 6,0 V 0 V < V_{DD} < 5,5 V	$\pm I_{LI2}$	–	10	μA

DC CHARACTERISTICS (continued)

parameter	conditions	symbol	min.	max.	unit
Outputs					
LOW level output voltage (Ports 1, 2 and 3; except P1.6/SCL, P1.7/SDA)	(note 4) I_{OL} = 1,6 mA	V_{OL}	–	0,45	V
LOW level output voltage (Port 0, ALE, $\overline{PSEN}$)	(note 4) I_{OL} = 3,2 mA	V_{OL1}	–	0,45	V
LOW level output voltage (P1.6/SCL, P1.7/SDA)	I_{OL} = 3,0 mA	V_{OL2}	–	0,40	V
HIGH level output voltage (Ports 1, 2 and 3)	V_{DD} = 5 V ± 10% $-I_{OH}$ = 60 μA $-I_{OH}$ = 25 μA $-I_{OH}$ = 10 μA	V_{OH} V_{OH} V_{OH}	2,4 $0,75V_{DD}$ $0,9V_{DD}$	– – –	V V V
HIGH level output voltage (Port 0 in external bus mode, ALE, $\overline{PSEN}$)	(note 5) V_{DD} = 5 V ± 10% $-I_{OH}$ = 400 μA $-I_{OH}$ = 150 μA $-I_{OH}$ = 40 μA	V_{OH1} V_{OH1} V_{OH1}	2,4 $0,75V_{DD}$ $0,9V_{DD}$	– – –	V V V
RST pull-down resistor		R_{RST}	50	150	kΩ
Capacitance of I/O buffer	Test freq. = 1 MHz T_{amb} = 25 °C	$C_{I/O}$	–	10	pF

DEVELOPMENT DATA

DC CHARACTERISTICS

AV_{DD} = 5,0 V ± 10%; AV_{SS} = 0 V; V_{REF} = 5,0 V; T_{amb} = −40 °C to + 125 °C (PCA83C654).
All voltages measured with respect to V_{SS} unless otherwise specified.

parameter	conditions	symbol	min.	max.	unit
Supply voltage		V_{DD}	4,5	5,5	V
Supply current operating	(note 1) f_{CLK} = 12 MHz; V_{DD} = 5,5 V	I_{DD}	–	tbf	mA
Idle mode	(note 2) f_{CLK} = 12 MHz; V_{DD} = 5 V ± 10%	I_{ID}	–	tbf	mA
Power-down current	2 V < V_{PD} < V_{DD} max. (note 3)	I_{PD}	–	tbf	µA
Inputs					
LOW level input voltage (except $\overline{EA}$, P1.6/SCL, P1.7/SDA)		V_{IL}	−0,5	$0{,}2V_{DD}-0{,}25$	V
LOW level input voltage ($\overline{EA}$)		V_{IL1}	−0,5	$0{,}2V_{DD}-0{,}45$	V
LOW level input voltage (P1.6/SCL, P1.7/SDA)	(note 6)	V_{IL2}	−0,5	$0{,}3V_{DD}$	V
HIGH level input voltage (except RST, XTAL1, P1.6/SCL, P1.7/SDA)		V_{IH}	$0{,}2V_{DD}+1{,}0$	$V_{DD}+0{,}5$	V
HIGH level input voltage (RST and XTAL1)		V_{IH1}	$0{,}7V_{DD}+0{,}1$	$V_{DD}+0{,}5$	V
HIGH level input voltage (P1.6/SCL, P1.7/SDA)	(note 6)	V_{IH2}	$0{,}7V_{DD}$	6,0	V
Input current logic 0 (Ports 1, 2 and 3; except P1.6/SDA, P1.7/SCL)	V_I = 2,0 V	$-I_{IL}$	–	75	µA
Input current logic 1 to 0 transition (Ports 1, 2 and 3; except P1.6/SCL, P1.7/SDA)	V_I = 2,0 V	$-I_{TL}$	–	750	µA
Input leakage current (Port 0, $\overline{EA}$)	0,45 < V_I < V_{DD}	$\pm I_{LI1}$	–	10	µA
Input leakage current (P1.6/SCL, P1.7/SDA)	0 V < V_I < 6,0 V 0 V < V_{DD} < 5,5 V	$\pm I_{LI2}$	–	10	µA

DC CHARACTERISTICS (continued)

parameter	conditions	symbol	min.	max.	unit
Outputs					
LOW level output voltage (Ports 1, 2 and 3; except P1.6/SCL, P1.7/SDA)	(note 4) I_{OL} = 1,6 mA	V_{OL}	–	0,45	V
LOW level output voltage (Port 0, ALE, $\overline{PSEN}$)	(note 4) I_{OL} = 3,2 mA	V_{OL1}	–	0,45	V
LOW level output voltage (P1.6/SCL, P1.7/SDA)	I_{OL} = 3,0 mA	V_{OL2}	–	0,40	V
HIGH level output voltage (Ports 1, 2 and 3)	V_{DD} = 5 V ± 10% $-I_{OH}$ = 60 μA $-I_{OH}$ = 25 μA $-I_{OH}$ = 10 μA	V_{OH} V_{OH} V_{OH}	2,4 $0{,}75V_{DD}$ $0{,}9V_{DD}$	– – –	V V V
HIGH level output voltage (Port 0 in external bus mode, ALE, $\overline{PSEN}$)	(note 5) V_{DD} = 5 V ± 10% $-I_{OH}$ = 400 μA $-I_{OH}$ = 150 μA $-I_{OH}$ = 40 μA	V_{OH1} V_{OH1} V_{OH1}	2,4 $0{,}75V_{DD}$ $0{,}9V_{DD}$	– – –	V V V
RST pull-down resistor		R_{RST}	50	150	kΩ
Capacitance of I/O buffer	Test freq. = 1 MHz T_{amb} = 25 °C	$C_{I/O}$	–	10	pF

Notes to the DC characteristics

1. The operating supply current is measured with all output pins disconnected; XTAL1 driven with $t_r = t_f$ = 10 ns; $V_{IL} = V_{SS}$ + 0,5 V; $V_{IH} = V_{DD}$ −0,5 V; XTAL2 not connected; $\overline{EA}$ = RST = Port 0 = V_{DD}.
2. The Idle mode supply current is measured with all output pins disconnected; XTAL1 driven with $t_r = t_f$ = 10 ns; $V_{IL} = V_{SS}$ + 0,5 V; $V_{IH} = V_{DD}$ −0,5 V; XTAL2 not connected; $\overline{EA}$ = Port 0 = V_{DD}; RST = V_{SS}.
3. The power-down current is measured with all output pins disconnected; XTAL2 not connected; $\overline{EA}$ = Port 0 = V_{DD}; RST = V_{SS}.
4. Capacitive loading on Port 0 and Port 2 may cause spurious noise pulses to be superimposed on the LOW level output voltage of ALE, Port 1 and Port 3.
   The noise is due to external bus capacitance discharging into the Port 0 and Port 2 pins when these pins make a 1-to-0 transition during bus operations. In the most adverse condition (capacitive loading > 100 pF) the noise pulse on ALE line may exceed 0,8 V. In this event it may be required to qualify ALE with a Schmitt trigger, or use an address latch with a Schmitt trigger STROBE input.

5. Capacitive loading on Port 0 and Port 2 may cause the HIGH level output voltage on ALE and $\overline{PSEN}$ to momentarily fall below the 0,9 V_{DD} specification when the address bits are stabilizing.
6. The input threshold voltage of P1.6 and P1.7 (SI01) meets the I^2C-bus specification. A voltage below 0,3 V_{DD} will therefore be recognised as a logic 0, while an input voltage above 0,7 V_{DD} will be recognised as a logic 1.

AC CHARACTERISTICS

V_{DD} = 5 V ± 20% (PCB83C654), V_{SS} = 0 V, $t_{CKmin} = 1/f_{max}$ (maximum operating frequency)

V_{DD} = 5 V ± 20% (PCF83C654), V_{SS} = 0 V, $t_{CKmin} = 1/f_{max}$ (maximum operating frequency)

V_{DD} = 5 V ± 10% (PCA83C654), V_{SS} = 0 V, $t_{CKmin} = 1/f_{max}$ (maximum operating frequency)

T_{amb} = 0 to + 70 °C, t_{CKmin} = 83 ns (PCB83C654 see notes 1 and 2)

T_{amb} = −40 to + 85 °C, t_{CKmin} = 83 ns (PCF83C654 see notes 1 and 2)

T_{amb} = −40 to + 125 °C, t_{CKmin} = 83 ns (PCA83C654 see notes 1 and 2)

C_L = 100 pF for Port 0, ALE and $\overline{PSEN}$; C_L = 80 pF for all other outputs unless otherwise specified. (See waveforms Figs 16, 17 and 18).

parameter	symbol	10 MHz		12 MHz		variable clock		unit
		min.	max.	min.	max.	min.	max.	
Program memory								
ALE pulse duration	t_{LL}	160	–	127	–	$2t_{CK}-40$	–	ns
Address set-up time to ALE	t_{AL}	45	–	28	–	$t_{CK}-55$	–	ns
Address hold time after ALE	t_{LA}	65	–	48	–	$t_{CK}-35$	–	ns
Time from ALE to valid instruction input	t_{LIV}	–	300	–	233	–	$4t_{CK}-100$	ns
Time from ALE to control pulse $\overline{PSEN}$	t_{LC}	60	–	43	–	$t_{CK}-40$	–	ns
Control pulse duration $\overline{PSEN}$	t_{CC}	255	–	205	–	$3t_{CK}-45$	–	ns
Time from $\overline{PSEN}$ to valid instruction input	t_{CIV}	–	195	–	145	–	$3t_{CK}-105$	ns
Input instruction hold time after $\overline{PSEN}$	t_{CI}	0	–	0	–	0	–	ns
Input instruction float delay after $\overline{PSEN}$	t_{CIF}	–	75	–	59	–	$t_{CK}-25$	ns
Address to valid instruction input	t_{AIV}	–	395	–	312	–	$5t_{CK}-105$	ns
Address float delay after $\overline{PSEN}$	t_{AFC}	–	10	–	10	–	10	ns

DEVELOPMENT DATA

AC CHARACTERISTICS

parameter	symbol	10 MHz		12 MHz		variable clock		unit
		min.	max.	min.	max.	min.	max.	
External data memory								
$\overline{RD}$ pulse duration	t_{RR}	500	–	400	–	$6t_{CK}-100$	–	ns
$\overline{WR}$ pulse duration	t_{WW}	500	–	400	–	$6t_{CK}-100$	–	ns
Address hold time after ALE	t_{LA}	–	–	48	–	$t_{CK}-35$	–	ns
$\overline{RD}$ to valid data input	t_{RD}	–	355	–	252	–	$5t_{CK}-165$	ns
Data hold time after $\overline{RD}$	t_{DR}	0	–	0	–	0	–	ns
Data float delay after $\overline{RD}$	t_{DFR}	–	130	–	97	–	$2t_{CK}-70$	ns
Time from ALE to valid data input	t_{LD}	–	650	–	517	–	$8t_{CK}-150$	ns
Address to valid data input	t_{AD}	–	735	–	585	–	$9t_{CK}-165$	ns
Time from ALE to $\overline{RD}$ or $\overline{WR}$	t_{LW}	250	350	200	300	$3t_{CK}-50$	$3t_{CK}+50$	ns
Time from address to $\overline{RD}$ or $\overline{WR}$	t_{AW}	270	–	203	–	$4t_{CK}-130$	–	ns
Time from $\overline{RD}$ or $\overline{WR}$ HIGH to ALE HIGH	t_{WHLH}	60	140	43	123	$t_{CK}-40$	$t_{CK}+40$	ns
Data valid to $\overline{WR}$ transition	t_{DWX}	40	–	23	–	$t_{CK}-60$	–	ns
Data set-up time before $\overline{WR}$	t_{DW}	550	–	433	–	$7t_{CK}-150$	–	ns
Data hold time after $\overline{WR}$	t_{WD}	50	–	33	–	$t_{CK}-50$	–	ns
Address float delay after $\overline{RD}$	t_{AFR}	–	0	–	0	–	0	ns

Notes to AC characteristics:

1. The maximum and minimum operating frequency for all versions is 12 MHz and 1,2 MHz respectively.
2. The minimum operating frequency is limited to 1,2 MHz for all versions.
3. Interfacing the PCB83C654 to devices with float times up to 75 ns is permitted. This limited bus contention will not cause damage to Port 0 drivers.

AC CHARACTERISTICS (continued)

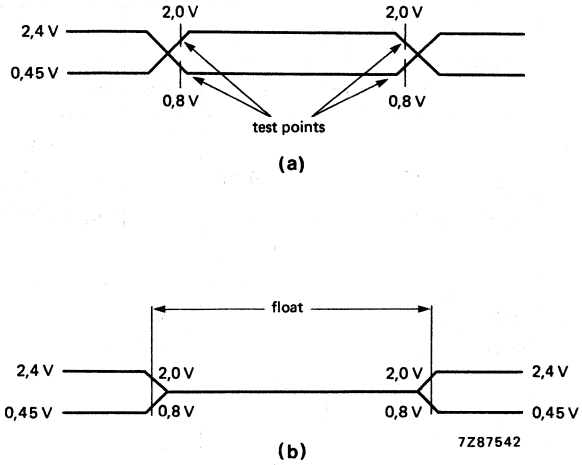


Fig. 14 AC testing input, output waveform (a) and float waveform (b).

AC testing inputs are driven at 2,4 V for a logic 1 and 0,45 V for a logic 0. Timing measurements are taken at 2,0 V for a logic 1 and 0,8 V for logic 0. The float state is defined as the point at which a Port 0 pin sinks 3,2 mA or sources 400 μA at the voltage test levels.

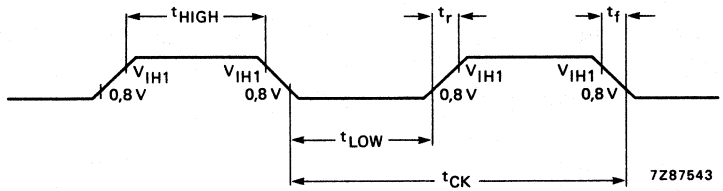


Fig. 15 External clock drive XTAL 1 (see Table 6).

Table 6 External clock drive XTAL 1 (see Fig. 15)

parameter	symbol	variable clock (f = 1,2 to 12 MHz)		unit
		min.	max.	
oscillator clock period	t_{CK}	83,3	833	ns
HIGH time	t_{HIGH}	20	$t_{CK}-t_{LOW}$	ns
LOW time	t_{LOW}	20	$t_{CK}-t_{HIGH}$	ns
rise time	t_r	–	20	ns
fall time	t_f	–	20	ns
cycle time	t_{CY} (1)	1	3,43	μs

(1) $t_{CY} = 12\ t_{CK}$.

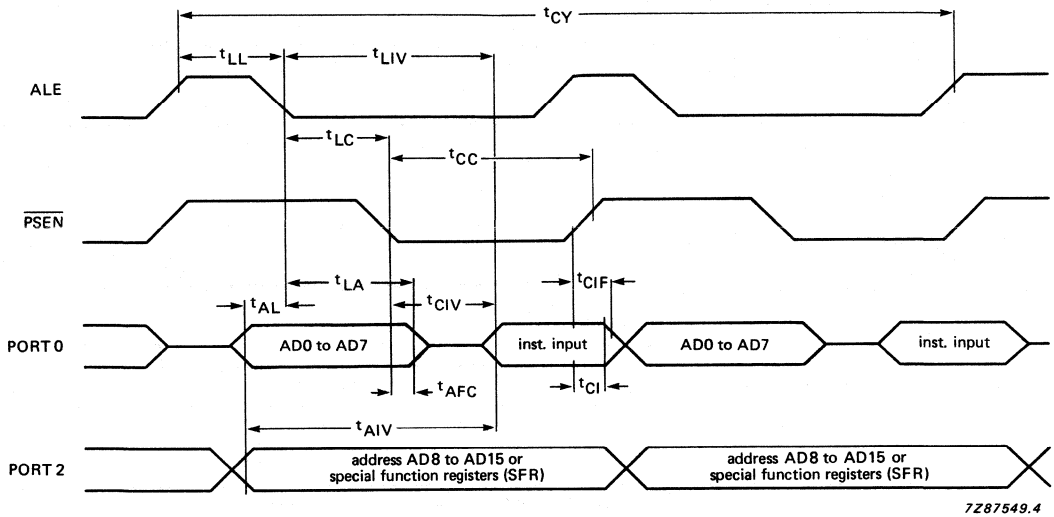


Fig. 16 Read from program memory.

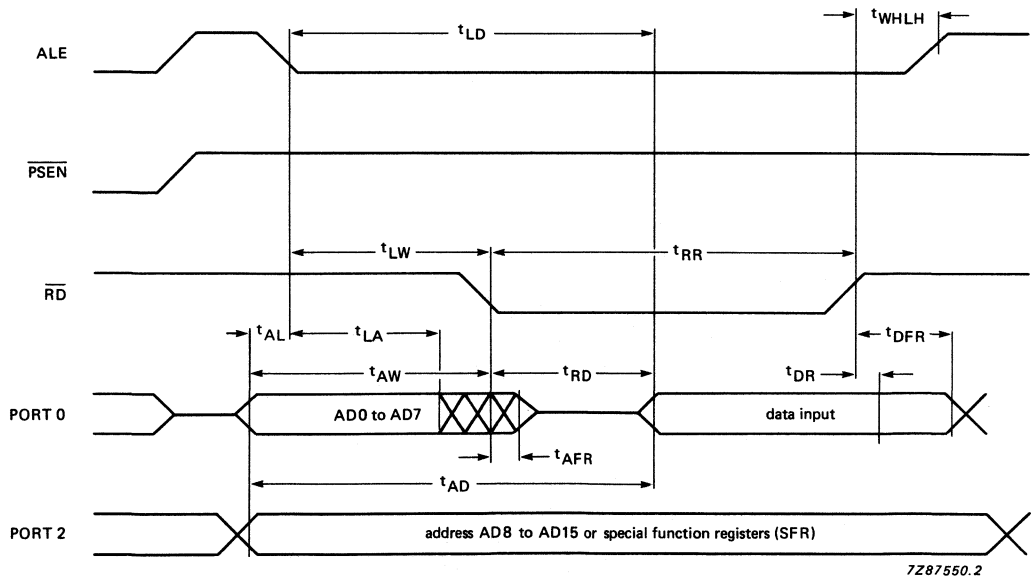


Fig. 17 Read from data memory.

DEVELOPMENT DATA

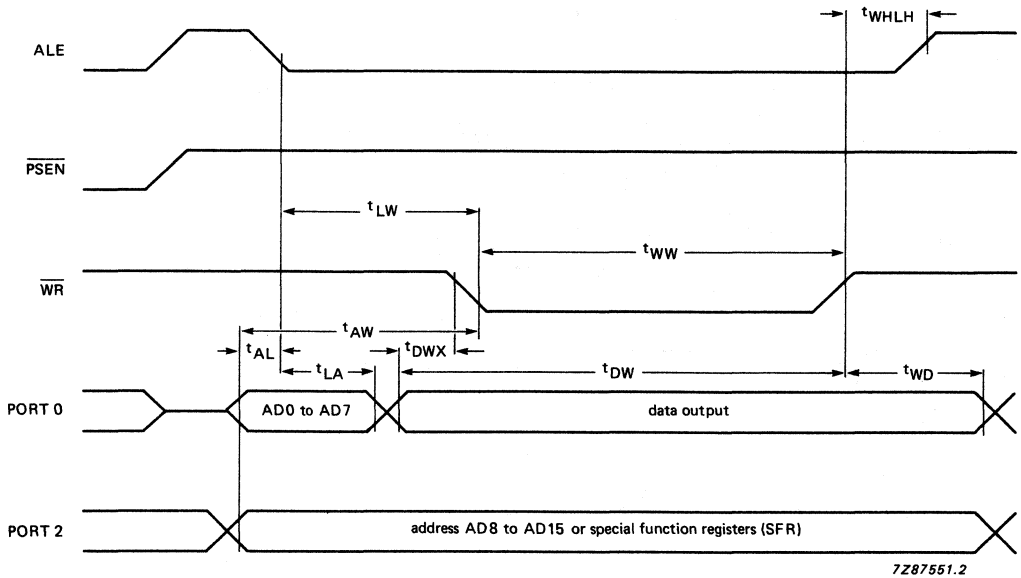
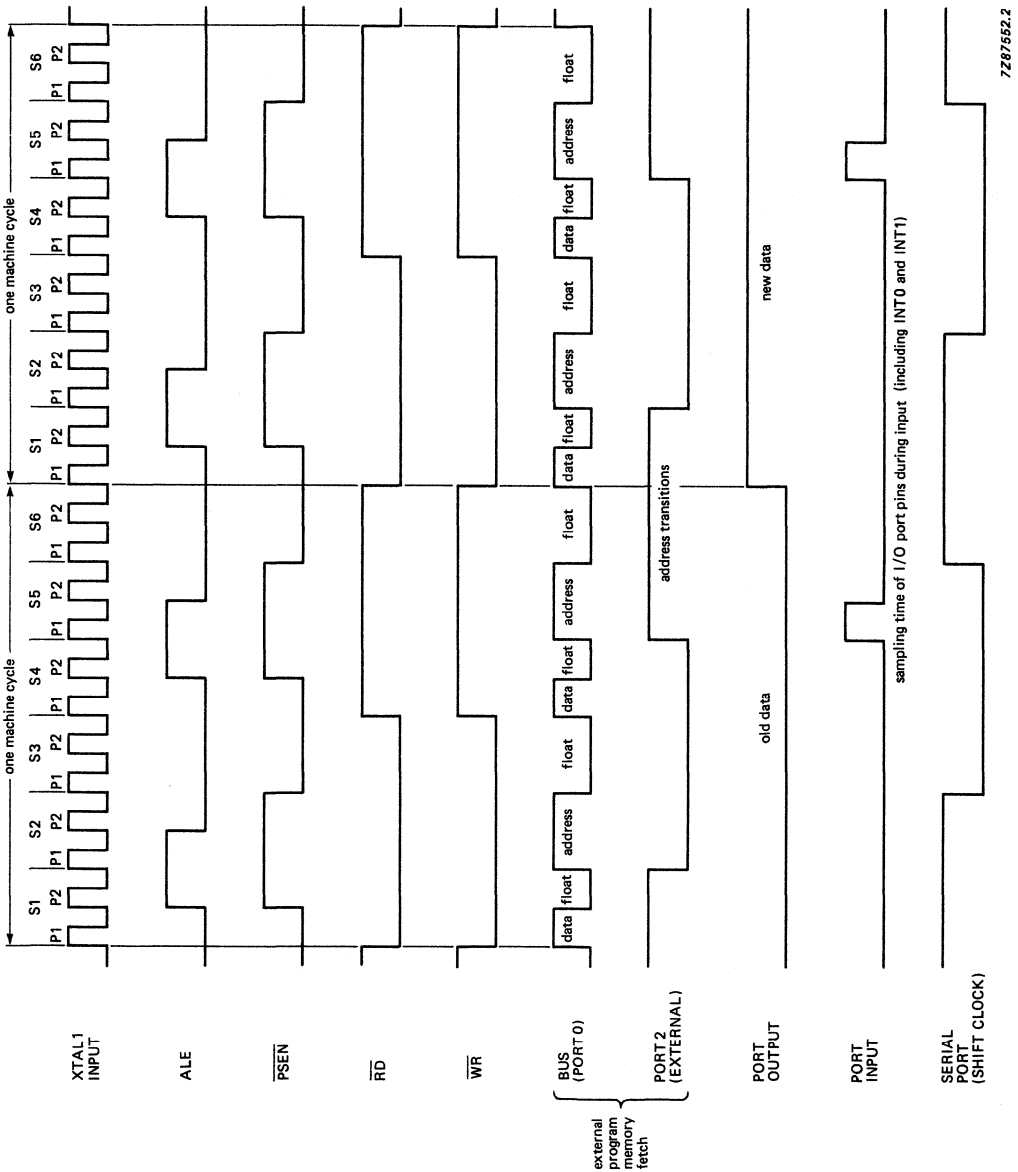


Fig. 18 Write to data memory.

DEVELOPMENT DATA



SINGLE-CHIP 8-BIT MICROCONTROLLER

GENERAL DESCRIPTION

The PCB83C851 single chip microcontroller is manufactured in an advanced CMOS process and is a derivative of the PCB80C51 microcontroller family. The PCB83C851 has the same instruction set as the PCB80C51. Two versions of the derivative exist although the generic term 'PCB83C851' is used to refer to both family members:

- PCB83C851: 4 K bytes mask-programmable ROM, 128 bytes RAM, 256 bytes EEPROM
- PCB80C851: ROM-less version of the PCB83C851

This device provides architectural enhancements that make it suitable for a variety of applications, specifically control systems.

The PCB83C851 contains a non-volatile 4 K x 8 read-only program memory; a volatile 128 x 8 read/write data memory; a 256 byte electrically erasable programmable read only memory (EEPROM); 32 I/O lines; two 16-bit timer/event counters (identical to the timers of the PCB80C51); a seven source, five-vector, two-priority-level, nested interrupt structure; a serial I/O port for either multi-processor communications, I/O expansion or full duplex UART; and on-chip oscillator and timing circuits. For systems that require extra capability, the PCB83C851 can be expanded using standard TTL compatible memories and logic.

The PCB83C851 has two software selectable modes of reduced activity for further power reduction: Idle and Power-down. The Idle mode freezes the CPU while allowing the RAM, timers, serial port and interrupt system to continue functioning. The Power-down mode saves the RAM contents but freezes the oscillator causing all other chip functions to be inoperative.

The device also functions as an arithmetic processor having facilities for both binary and BCD arithmetic plus bit-handling capabilities. The instruction set consists of over 100 instructions: 49 one-byte, 46 two-byte and 16 three-byte. With a 12 MHz crystal, 58% of the instructions are executed in 1 μs and 40% in 2 μs. Multiply and divide instructions require 4 μs.

Features

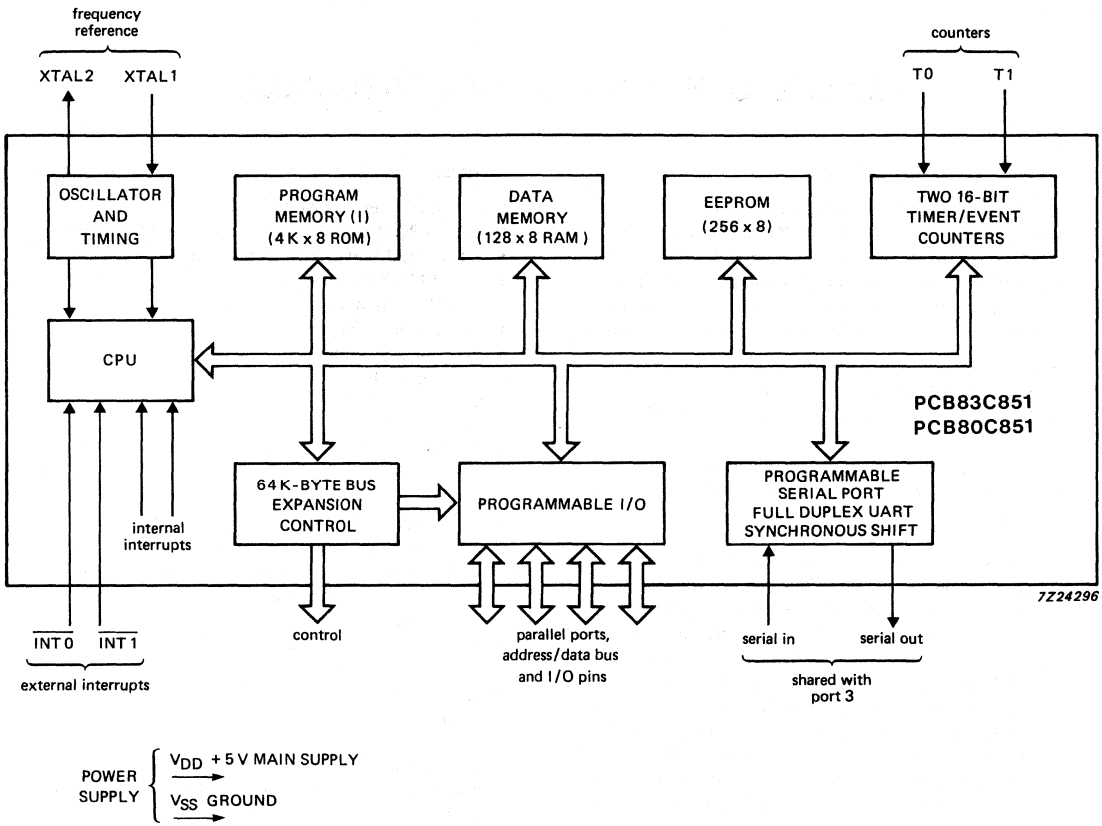
- PCB80C51 central processing unit
- 4 K x 8 ROM, expandable externally to 64 K bytes
- 128 x 8 RAM, expandable externally to 64 K bytes
- Four 8-bit I/O ports, 32 I/O lines
- Two 16-bit timer/event counters
- Full-duplex serial port
- Boolean processing
- On-chip oscillator
- Seven-source, five-vector interrupt structure with two priority levels
- 58% of the instructions are executed in 1 μs; multiply and divide in 4 μs; all others are executed in 2 μs (with a 12 MHz oscillator)
- Enhanced architecture with non-page-oriented-instructions, direct addressing, four 8-byte register banks, stack depth up to 128-bytes, multiply, divide, subtract and compare instructions
- ROM code protection (mask-programmable)
- Security mode, user dependent protection of the EEPROM contents
- Additional interrupt source (EEPROM) 'ORed' with serial interrupt

EEPROM:

- Non-volatile 256 x 8 bit EEPROM (electrically erasable programmable read only memory)
- On-chip voltage multiplier for erase/write
- 10000 erase/write cycles per byte
- 10 years non volatile data retention
- Infinite number of read cycles

PACKAGE OUTLINES

PCD/PCF83C851/80C851P: 40-lead DIL; plastic (SOT 129)
PCB/PCF83C851/80C851WP: 44-lead PLCC; plastic, leaded-chip-carrier (SOT187AA)



Note (1): PCB/PCF83C851 only

Fig. 1 Block diagram.

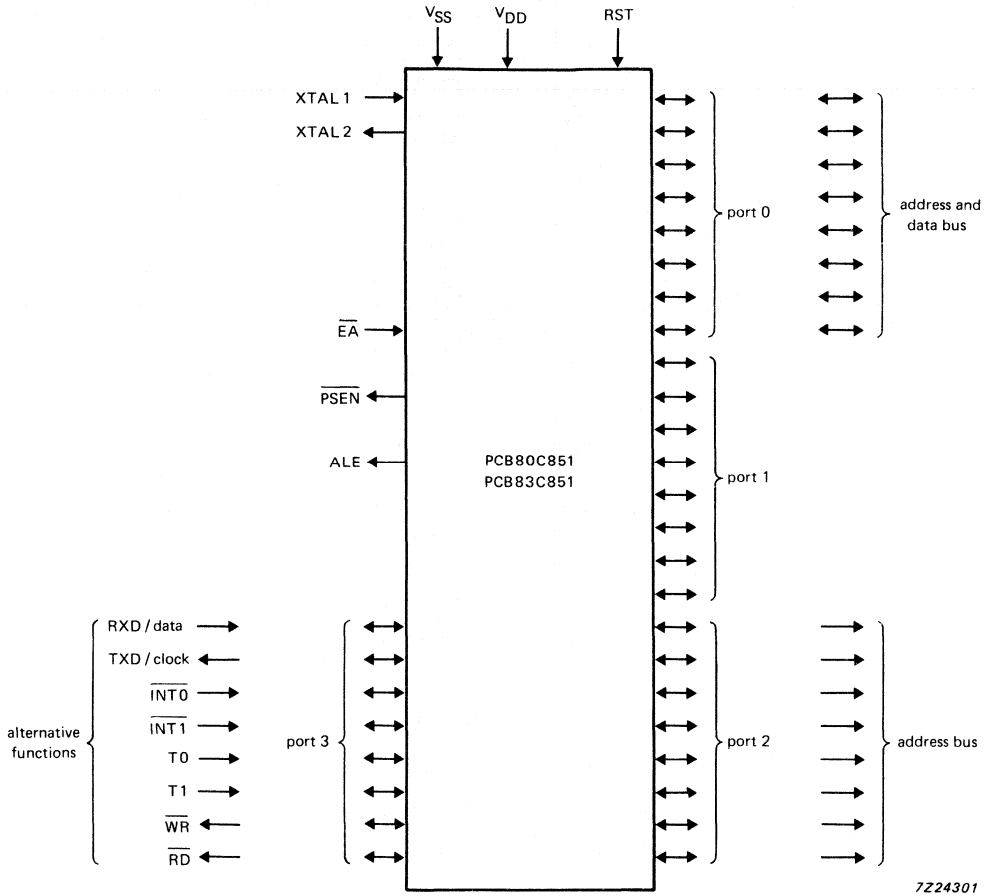


Fig. 2 Functional diagram.

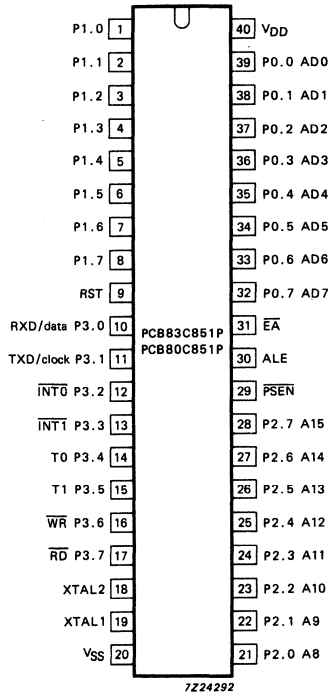


Fig. 3(a) Pinning diagram (DIL versions).

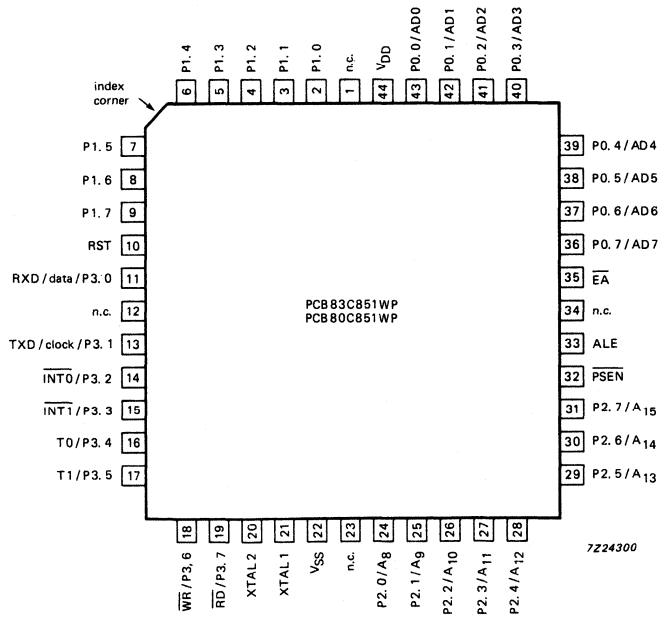


Fig. 3(b) Pinning diagram (PLCC versions).

PINNING (DIL versions)

pin no.	designation	function
1 – 8	P1.0-P1.7	Port 1: 8-bit quasi bidirectional I/O port with internal pull-up. Port 1 can sink/source one TTL (= 4 LSTTL) input. It can drive CMOS inputs without external pull-ups.
9	RST	Reset: a high level on this pin for two machine cycles while the oscillator is running resets the device. An internal pull-down permits Power-on reset using only a capacitor connected to V_{DD}.
10 – 17	P3.0-P3.7	Port 3: 8-bit quasi bidirectional I/O port. Port 3 has the following alternate functions:
10	P3.0/RXD	RXD/data: serial port receiver data input (asynchronous) or data input/output (synchronous).
11	P3.1/TXD	TXD/clock: serial port transmitter data output (asynchronous) or clock output (synchronous).
12	P3.2/$\overline{INT0}$	External interrupt 0 or gate control input for timer/event counter 0.
13	P3.3/$\overline{INT1}$	External interrupt 1 or gate control input for timer/event counter 1.
14	P3.4/T0	External input for timer/event counter 0.
15	P3.5/T1	External input for timer/event counter 1.
16	P3.6/$\overline{WR}$	External data memory write strobe.
17	P3.7/$\overline{RD}$	External data memory read strobe.
		Operation of an alternative function is automatic provided that the associated SFR bit is set HIGH.
18	XTAL2	Crystal input 2: output of the inverting amplifier that forms the oscillator; left open when an external oscillator is used (see Fig. 10).
19	XTAL1	Crystal input 1: input to the inverting amplifier that forms the oscillator, and input from an external clock generator. Receives the external oscillator signal when an external oscillator is used (see Fig. 10).
20	V_{SS}	Ground: circuit ground potential.
21 – 28	P2.0-P2.7	Port 2: 8-bit quasi bidirectional I/O port. Port 2 is used during external memory accesses to output the high order byte of the external memory address if the address is 16 bits wide. If the address is 8 bits wide (data memory accesses only), Port 2 outputs the contents of the Port 2 special function register.
29	$\overline{PSEN}$	Program store enable output: read strobe to the external program memory. It is activated twice each machine cycle during fetches from external program memory. When executing out of external program memory, two activations of $\overline{PSEN}$ are skipped during each access to external data memory. $\overline{PSEN}$ is not activated (remains HIGH) during fetches from internal program memory. $\overline{PSEN}$ can sink/source 8 LSTTL inputs. It can drive CMOS inputs without an external pull-up.
30	ALE	Address latch enable output: latches the low byte of the address during access to external memory in normal operation. It is activated every six oscillator periods except during an external data memory access. ALE can sink/source 8 LSTTL inputs. It can drive CMOS inputs without an external pull-up.
31	$\overline{EA}$	External access input: When $\overline{EA}$ is held at a TTL HIGH level, the CPU executes out of the internal program memory (ROM), provided the program counter is less than 4096. When $\overline{EA}$ is held at a TTL LOW level, the CPU executes out of external program memory. The status of the $\overline{EA}$ pin can only be changed during Reset.
32 – 39	P0.0-P0.7	Port 0: 8-bit bidirectional I/O port. It is also used as the multiplexed low-order address and data bus during access to external memory (during these accesses, it activates internal pull-ups). Port 0 can sink/source 8 TTL inputs.
40	V_{DD}	Power supply: + 5 V power supply pin during normal operation, Idle mode and Power-down mode. In Power-down mode, V_{DD} may be reduced to minimize circuit power consumption.

FUNCTIONAL DESCRIPTION

General

The PCB83C851 is a stand-alone high-performance microcontroller designed for use in real-time applications such as instrumentation, smart-cards, industrial control and intelligent computer peripherals.

The device provides hardware features, architectural enhancements and instructions to function as a controller for applications requiring up to 64 K bytes of program memory and/or up to 64 K bytes of data storage.

The PCB80C851 is a control-oriented CPU without on-chip program memory. It can address 64 K bytes of external program memory in addition to 64 K bytes of external data memory. The PCB83C851 is a PCB80C851 with the lower 4 K bytes of program memory filled with on-chip mask programmable ROM. For systems requiring extra capability, the PCB83C851 can be expanded using standard memories and peripherals.

The two pin-compatible versions of this component reduce development problems to a minimum and provide maximum flexibility. The PCB80C851 can be easily modified and updated in the field.

The PCB83C851 contains a non-volatile 4 K x 8 read-only program memory; a volatile 128 x 8 read/write data memory; a 256 byte electrically erasable programmable read only memory (EEPROM); 32 I/O lines; two 16-bit timer/event counters; a seven source, five-vector, two-priority-level, nested interrupt structure; a serial I/O port for either multi-processor communications, I/O expansion or full duplex UART; and on-chip oscillator and timing circuits.

In addition, some special security features are implemented (for details see the section on security facilities):

- ROM code protection: mask-programmable. When implemented, access to the internal ROM is possible only when executing internal program memory; it is not possible to access the internal ROM when executing external program memory.
- In the security mode (enabled when the security bit is set), the contents of the EEPROM are protected and, when executing external program memory, no read or write operation to the EEPROM is possible except 'Blockerase' ('Blockerase' clears all bytes including the byte containing the security bits).

Note: Using the PCB83C851/80C851 with external program memory only and setting the security bits is senseless.

Memory organization

The central processing unit (CPU) manipulates operands in four memory spaces; these include the 64 k byte external data memory, 256 byte internal data memory and the 64 k byte internal and external program memory. The fourth memory space is the EEPROM data memory of 256 bytes. Communication between the CPU and the EEPROM is achieved via special function registers (see the section on the EEPROM). The internal data memory address space is subdivided into 128 byte internal data RAM and 128 byte special function register (SFR) address spaces, as shown in Fig. 4.

Table 1 shows the special function register memory map. Internal RAM locations 0-127 are directly and indirectly addressable. Special function register locations 128-255 are only directly addressable.

Table 1 Special function register memory map

Register Mnemonic	Bit Address								Direct Byte Address (Hex)
ECNTRL									F6H
ETIM									F5H
EDAT									F4H
EADRH									F3H
EADRL									F2H
B	F7	F6	F5	F4	F3	F2	F1	F0	F0H
ACC	E7	E6	E5	E4	E3	E2	E1	E0	E0H
PSW	D7	D6	D5	D4	D3	D2	D1	D0	D0H
IP	BF	BE	BD	BC	BB	BA	B9	B8	B8H
P3	B7	B6	B5	B4	B3	B2	B1	B0	B0H
IE	AF	AE	AD	AC	AB	AA	A9	A8	A8H

Table 1 (continued)

Register Mnemonic	Bit Address								Direct Byte Address (Hex)
P2	A7	A6	A5	A4	A3	A2	A1	A0	A0H
SBUF									99H
SCON	9F	9E	9D	9C	9B	9A	99	98	98H
P1	97	96	95	94	93	92	91	90	90H
TH1									8DH
TH0									8CH
TL1									8BH
TL0									8AH
TMOD									89H
TCON	8F	8E	8D	8C	8B	8A	89	88	88H
PCON									87H
DPH									83H
DPL									82H
SP									81H
P0	87	86	85	84	83	82	81	80	80H

The internal data RAM contains four register banks (each with eight registers), 128 addressable bits, and the stack. The stack depth is limited by the available internal data RAM and its location is determined by the 8 bit stack pointer. All registers except the program counter and the four 8 register banks reside in the special function register address space. These memory mapped registers include arithmetic registers, pointers, I/O ports, interrupt system registers, timers, serial port and EEPROM registers. There are 128 addressable bit locations in the SFR address space. The PCB83C851 contains 128 bytes of internal data RAM and 25 special function registers. It provides a non-paged program memory address space to accommodate relocatable code. Conditional branches are performed relative to the programm counter. The register-indirect jump permits branching relative to a 16 bit base register with an offset provided by an 8-bit index register. 16-bit jumps and calls permit branching to any location in the continguous 64 K program memory address space.

Addressing

The PCB83C851 has five methods of addressing source operands:

- Register
- Direct
- Register-Indirect
- Immediate
- Base-Register plus Index-Register-Indirect

The first three methods can be used for addressing destination operands. Most instructions have a 'destination/source' field that specifies the data type, addressing methods and operands involved. For operations other than MOVs, the destination operands is also a source operand.

Access to memory addressing is as follows:

- Registers in the four 8-register banks through Register, Direct, or Register-Indirect
- 128 bytes of internal data RAM through Direct or Register-Indirect
- Special function registers through Direct
- External data memory through Register-Indirect
- Program memory look-up tables through Base-Register plus Index-Register-Indirect

The PCB83C851 is classified as an 8-bit device since the internal ROM, special function registers (SFR), Arithmetic Logic Unit (ALU), and external bus are each 8-bit wide. It performs operations on bit, nibble, byte and double-byte data types. Facilities are available for byte transfer, logic, and integer arithmetic operations, data transfer, logic, and conditional branch operations can be performed directly on Boolean variables to provide excellent bit handling.

I/O facilities

The PCB83C851 has 32 I/O lines. These I/O lines are controlled by 32 individually addressable bits in four special function registers. Port 0,1,2 and 3 perform the following alternate functions:

- Port 0: provides the multiplexed low-order address and data bus for expanding the PCB83C851 with standard memories and peripherals.
- Port 2: provides the high-order address when expanding the PCB83C851 with external program memory or more than 256 bytes of external data memory.
- Port 3: pins can be configured individually (by writing a logic 1 to the associated P3 SFR bits) to provide:
  - external interrupt request inputs
  - counter inputs
  - serial port receiver and transmitter output
  - control signals to READ and WRITE to external data memory

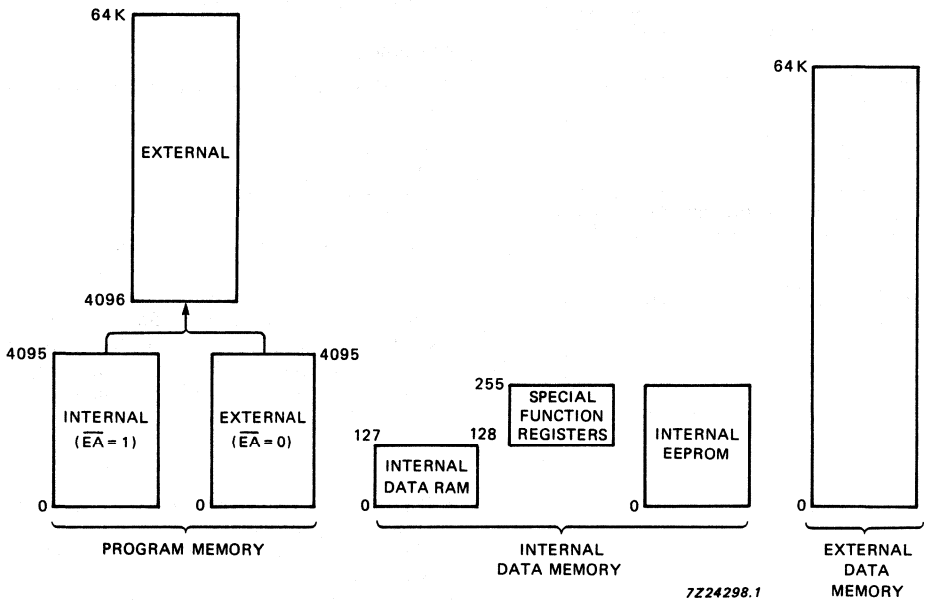


Fig. 4 Memory Map.

Timer/event counters

The PCB83C851 contains two 16-bit registers, Timer 0 and Timer 1, that can be used as timers or event counters to carry out the following functions:

- Measure time intervals and pulse durations
- Count events
- Generate interrupt requests

Each timer/event counter can be programmed independently to operate in three modes:

- Mode 0: 8-bit timer or 8-bit counter each with divide by 32 prescaler
- Mode 1: 16-bit time-interval or event counter
- Mode 2: 8-bit time-interval or event counter with automatic reload upon overflow

Timer 0 can be programmed to operate in an additional mode as follows:

- Mode 3: one 8-bit time-interval or event counter and one 8-bit time/interval counter

When Timer 0 is in Mode 3, Timer 1 can be programmed to operate in Modes 0, 1 or 2 but cannot set an interrupt request flag or generate an interrupt. However the overflow from Timer 1 can be used to pulse the serial port transmission-rate generator.

The frequency handling range of these counters with 12 MHz crystal is as follows:

- Up to 1 MHz when programmed for an internal input (the oscillator frequency divided by 12)
- 0 Hz to an upper limit of 0.5 MHz when programmed for external inputs

Both internal and external inputs can be gated to the counter by a second external source for directly measuring pulse durations.

The counters are started and stopped under software control. Each one sets its interrupt request flag when it overflows from all 1's to all 0's (or automatic reload value), with the exception of Mode 3.

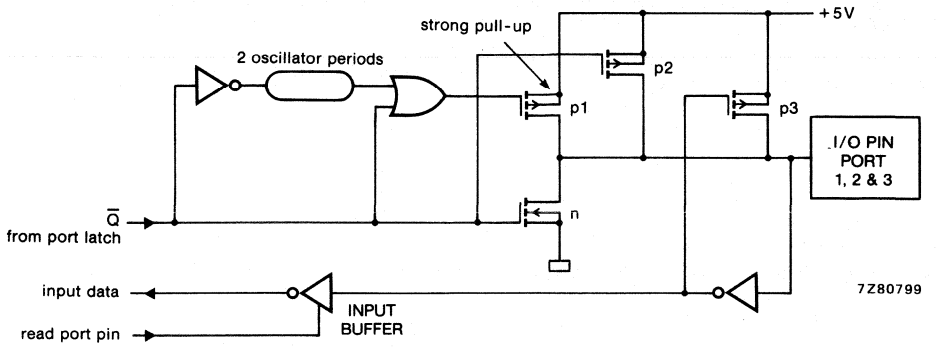


Fig 5 I/O buffers in the PCB83C851 (Ports 1,2 and 3).

On-chip peripherals functions

In addition to the CPU and memories, an interrupt system, extensive I/O facilities, and several peripherals functions are integrated on-chip to relieve the CPU of repetitious, complicated or time- critical tasks and to permit stringent real-time control of external system interface. The I/O facilities include the I/O pins, parallel ports, bidirectional address/data bus and the serial port for I/O expansion. The CPU peripheral functions integrated on-chip are the two 16-bit timer/event counters and the serial port.

EEPROM

Communication between the CPU and the EEPROM is accomplished via 5 Special Function Registers: 2 address registers (high and low byte), 1 data register for read and write operations, 1 control register, and 1 timer register to adapt the erase/write time to the clock frequency. All registers can be read and written. Fig. 6 shows the block diagram of the 80C51 CPU, the EEPROM and the interface.

Register and functional description

Address register (EADRH, EADRL):

The lower byte contains the address of one of the 256 bytes. The higher byte (EADRH) is for future extensions and for addressing the security bits (see the section on security facilities). The special function register addresses are shown in Table 1.

Data register (EDAT)

This register is required for read and write operations and also for row/block erase. In write mode its contents are written to the addressed byte (for 'row erase' and 'block erase' the contents are don't care). The write pulse starts all operations, except read. In read mode EDAT contains the data of the addressed byte.

Timer register (ETIM)

The timer register is required to adapt the erase/write time to the oscillator frequency. The user has to ensure that the erase or write (program) time is neither too short or too long.

Table 2 gives the values which must be written to the ETIM register by software for various oscillator frequencies. (the default value is 08H after RESET).

Table 2 Values for the Timer register ETIM

f_{XTAL1}	Values for ETIM		
	Binary	hexadecimal	decimal
1.0 MHz	00001000	08	8
2.0 MHz	00010011	13	19
3.0 MHz	00011101	1D	29
4.0 MHz	00101000	28	40
5.0 MHz	00110010	32	50
6.0 MHz	00111100	3C	60
7.0 MHz	01000111	47	71
8.0 MHz	01010001	51	81
9.0 MHz	01011100	5C	92
10.0 MHz	01100110	66	102
11.0 MHz	01110001	71	113
12.0 MHz	01111011	7B	123

The general formula is:

$$\text{Value (decimal)} = \frac{f_{XTAL1}\,(\text{kHz})}{96} - 2$$

The adjustment range is 0 to 255 corresponding to frequencies of 192 kHz to 24.7 MHz.

Control register (ECNTRL)

ECNTRL:

7	6	5	4	3	2	1	0
IFE	EEINT	EWP	–	ECNTRL3	ECNTRL2	ECNTRL1	ECNTRL0

Bit	Symbol	Function
ECNTRL.7	IFE	Active HIGH EEPROM interrupt flag set by the sequencer or by software; reset by software. When set and enabled, this flag forces an interrupt to the same vector as the serial port interrupt (see the interrupt section).
ECNTRL.6	EEINT	EEPROM interrupt enable set and reset by software (active HIGH)
ECNTRL.5	EWP	Erase/write in progress flag set and reset by the sequencer (active HIGH). When EWP is set, access to the EEPROM is not possible. EWP cannot be set or reset by software.

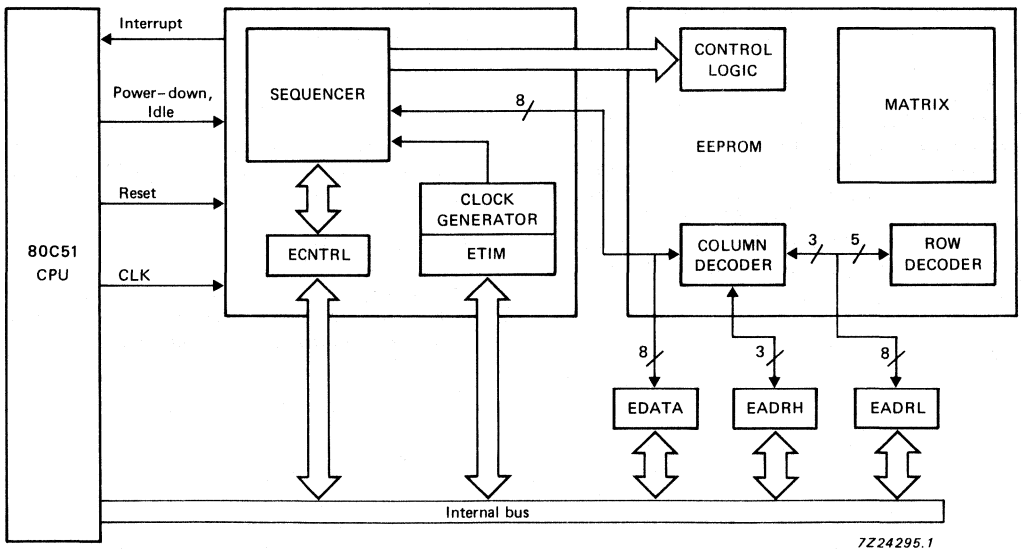


Fig. 6 Block diagram of the EEPROM interface.

ECNTRL.4 – (reserved)

ECNTRL.3
ECNTRL.2
ECNTRL.1
ECNTRL.0
} These four bits are used as follows:

Operation	ECNTRL.3	ECNTRL.2	ECNTRL.1	ECNTRL.0
Byte mode	0	0	0	0
Row erase	1	1	0	0
Page write*	–	–	–	–
Page erase/ write*	–	–	–	–
Block erase	1	0	1	0

* Future products.

Byte mode: Normal EEPROM mode, default mode after reset. In this mode, data can be read and written to one byte at a time.

Read mode: This is the default mode when byte mode is selected. This means that the contents of the addressed byte are available in the data register.

Write mode: This mode is activated by writing to the data register. The address register must be loaded first. Since the old contents are read first (by default), this allows the sequencer to decide whether an erase/write or write cycle only (data = 00H) is required.

Row erase: In this mode, the addressed row is cleared. The three LSBs of EADRL are not significant, i.e. the 8 bytes addressed by EADRL are cleared in the same time normally needed to clear one byte ($t_{rowerase} = t_e = t_w$). For the following write modes, only the write and not the erase/ write cycle is required. For example, using the row erase mode, programming 8 bytes takes $t_{total} = t_e + 8 \times t_w$ compared to $t_{total} = 8 \times t_e + 8 \times t_w$ ($t_e = t_{erase}$, $t_w = t_{write}$).

Page write: for future products.

Page erase/write: for future products.

Block erase: In this mode all 256 bytes are cleared; the byte containing the security bits is also cleared. $t_{blockerase} = t_e$. The contents of EADRH, EADRL and EDAT are insignificant.

Program sequences and register contents after Reset

The contents of the EEPROM registers after a Reset are the default values:

EADRH	= 80H	(security bit address)
EADRL	= 00H	(security bit address)
ETIM	= 08H	(minimum erase time with the lowest permissable oscillator frequency)
ECNTRL	= 00H	(Byte mode, read)
EDAT	= xxH	(security bit)

```
Initialize:   MOV ETIM, ..
              MOV EADRH, ..

Read:         MOV EADRL, ..
              MOV .., EDAT

Write:        MOV EADRL, ..
              MOV EDAT, ..

Erase row:    MOV EADRL, ..      Row address, 3 LSBs
                                 don't care
              MOV ECNTRL, 0CH    Erase row mode
              MOV EDAT, ..       (EDAT) don't care

Erase block:  MOV ECNTRL, 0AH    Erase block mode
              MOV EDAT, ..       (EDAT) don't care
```

If the security bit is to be altered, the program generally starts as follows:

```
MOV EADRH, #80H
MOV EADRL, #00H
```

Security facilities

ROM code protection

Since the external access mode can only be selected by pulling the $\overline{EA}$ pin LOW during Reset, it is not possible to read the internal program memory using the MOVC instruction while executing external program memory. Furthermore, it is not possible to change this mode to internal access within the MOVC cycle.

Additionally, a mask-programmable ROM code protection facility is available. When the program memory passes the 4 K boundary using both the internal and external ROMs, it is not possible to make an access to the internal ROM from the external program memory if the mask-programmable ROM security bit is set. An access to the lower 4 K bytes of program memory using the MOVC instruction is only possible while executing internal program memory.

Also the verification mode (test-mode which writes the ROM contents to a port for comparison with a reference code) is not implemented for security reasons. A different test-mode is implemented for test purpose. This mode allows every bit to be tested. However, the internal code can not be accessed via a port.

EEPROM protection

The EEPROM is protected using four security bits. These bits are contained in an extra EEPROM byte at address 8000H (EADRH/EADRL). They can be set or cleared by software.

To activate the EEPROM protection, the program sequence in byte mode must be as follows:

```
MOV EADRH, #80H
MOV EADRL, #00H
MOV EDAT,  #FFH
```

If two or more of these bits are reset, SB = 0, the security mode is disabled and the EEPROM is not protected. If three or four bits are set, SB = 1 and the $\overline{EA}$ mode differs from the internal access mode.

In this case access to the EEPROM is only possible in one mode regardless of how the external access mode is reached (by pulling the $\overline{EA}$ pin LOW or by passing the 4 K boundary). For SB = 1 and 'external access' only, the 'block erase' mode is enabled. The program sequence has to be as follows:

```
MOV EADRH,  #80H  (security byte address)
MOV EADRL,  #00H  (security byte address)
MOV ECNTRL, #0AH  (block erase mode)
MOV EDAT,   #xxH  (start block erase)
```

All 256 data bytes, the security bits, and SB will be cleared after completing this mode (EWP = 0). SB will also be affected in byte mode when writing to the security byte (not for SB = 1 and 'external access'). Fig. 7 illustrates the access to SB.

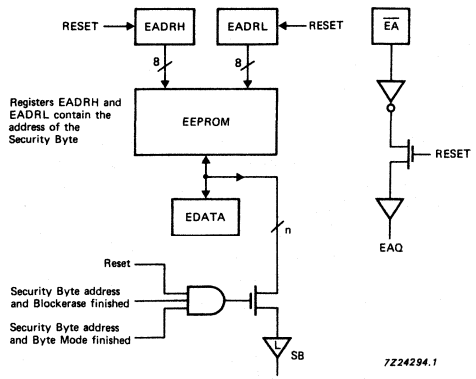


Fig. 7a Protection of the EEPROM (functional).

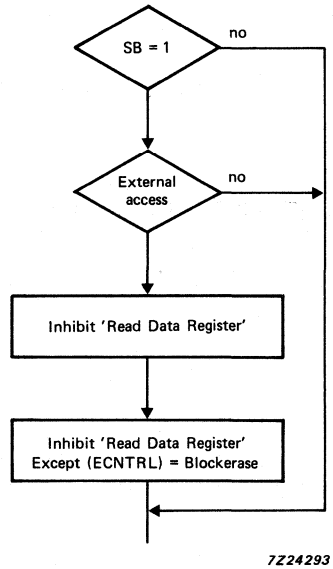


Fig. 7b EEPROM protection (flowchart).

Idle and Power-down operation (see Fig. 8)

Entering the Power-down mode freezes the oscillator. In the Idle mode, the interrupt, serial port and timer blocks continue to function while the clock to the CPU is halted.

Power control register (PCON)

These special modes are activated by software via the PCON special function register. Its hardware address is 87H. PCON is not bit addressable.

	MSB							LSB
PCON (87H)	SMOD	–	–	–	GF1	GF0	PD	IDL

bit	symbol	function
PCON.7	SMOD	Double baud rate bit, when set to logic one baud rate is doubled when the serial port is being used in modes 1, 2 or 3
PCON.6	–	(reserved)
PCON.5	–	(reserved)
PCON.4	–	(reserved)
PCON.3	GF1	General-purpose flag bit
PCON.2	GF2	General-purpose flag bit
PCON.1	PD	Power-down bit. Setting this bit activates the Power-down mode.
PCON.0	IDL	Idle mode bit. Setting this bit activates the Idle mode.

Note: Before entering the Idle or Power-down modes, the user has to ensure that there is no EEPROM erase/write cycle in progress (i.e. the EWP bit has to be reset before activating the Idle or Power-down modes; otherwise EEPROM accesses will be aborted).

Idle mode

The instruction that sets PCON.0 is the last instruction executed in the normal operating mode before Idle mode is activated. Once in the Idle mode, the CPU status is preserved in its entirety: the Stack Pointer, Program Counter, Program Status Word, Accumulator, RAM and all other registers maintain their data during Idle mode. The status of the external pins during Idle mode is shown in Table 3.

There are two ways to terminate the Idle mode:

Activation of an enabled interrupt will cause PCON.0 to be cleared by hardware terminating Idle mode. The interrupt is serviced, and following return from interrupt instruction RETI, the next instruction to be executed will be the one which follows the instruction that wrote a logic 1 to PCON.0. The flag bits GF0 and GF1 may be used to determine whether the interrupt was received during normal execution or during the Idle mode. For example, the instruction that writes to PCON.0 can also set or clear one or both flag bits. When Idle mode is terminated by an interrupt, the service routine can examine the status of the flag bits.

The second way of terminating the Idle mode is with a hardware reset*. Since the oscillator is still running, the hardware reset is required to be active for two machine cycles (24 oscillator periods) to complete the reset operation.

Power-down mode

The instruction that sets PCON.1 is the last executed prior to entering the Power-down mode. The oscillator is stopped. Only the contents of the on-chip RAM and EEPROM are preserved. The special function registers are not saved. A hardware reset is the only way of exiting the Power-down mode*.

In the Power-down mode, V_{DD} may be reduced to minimize circuit power consumption. The voltage must not be reduced until the Power-down mode is entered, and must be restored before the hardware reset is applied which will free the oscillator. Reset should not be released until the oscillator has restarted and stabilized.

The status of the external pins during Power-down mode is shown in Table 3. If the Power-down mode is activated while in external program memory, the port data that is held in the P2 special function register is restored to Port 2. If the data is a logic 1, the port pin is held HIGH during the Power-down mode by the strong pull-up transistor p1 (see Fig. 5).

Interrupt system

External events and the real-time-driven on-chip peripherals require service by the CPU asynchronous to the execution of any particular section of code. To tie the asynchronous activities of these functions to normal program execution a multiple-source, two-priority-level, nested interrupt system is provided. Interrupt response latency is from 3 µs to 7 µs when using a 12 MHz crystal. The PCB83C851 acknowledges interrupt requests from 7 sources as follows:

- $\overline{INT0}$ and $\overline{INT1}$: externally via pins 12 and 13 respectively
- Timer 0 and Timer 1: from the two internal counters
- Serial Port: from the internal serial I/O port or EEPROM (1 vector)

Each interrupt vectors to a separate location in program memory for its service program. Each source can be individually enabled (the EEPROM interrupt can only be enabled when the serial port interrupt is enabled) or disabled and can be programmed to a high or low priority level. All enabled sources can also be globally disabled or enabled. Both external interrupts can be programmed to be level-activated and are active LOW to allow 'wire-0Ring' of several interrupt sources to one input pin.

Note: The serial port and EEPROM interrupt flags must be cleared by software; all other flags are cleared by hardware.

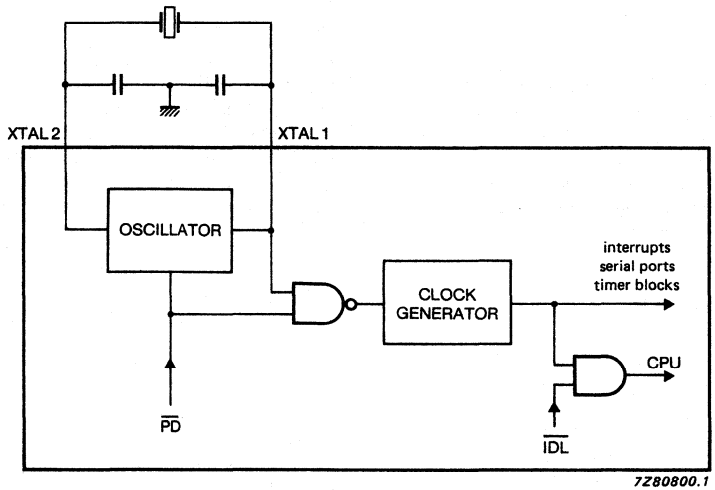


Fig. 8 Internal Idle and Power-down clock configuration.

Table 3 Status of the external pins during Idle and Power-down Mode.

mode	memory	ALE	PSEN	Port 0	Port 1	Port 2	Port 3
Idle	internal	1	1	port data	port data	port data	port data
Idle	external	1	1	floating	port data	address	port data
Power-down	internal	0	0	port data	port data	port data	port data
Power-down	external	0	0	floating	port data	port data	port data

* Terminating Idle or Power-down mode with a hardware RESET will cause a RAM and EEPROM clear if $\overline{EA} = 0$ and the security bit is set.

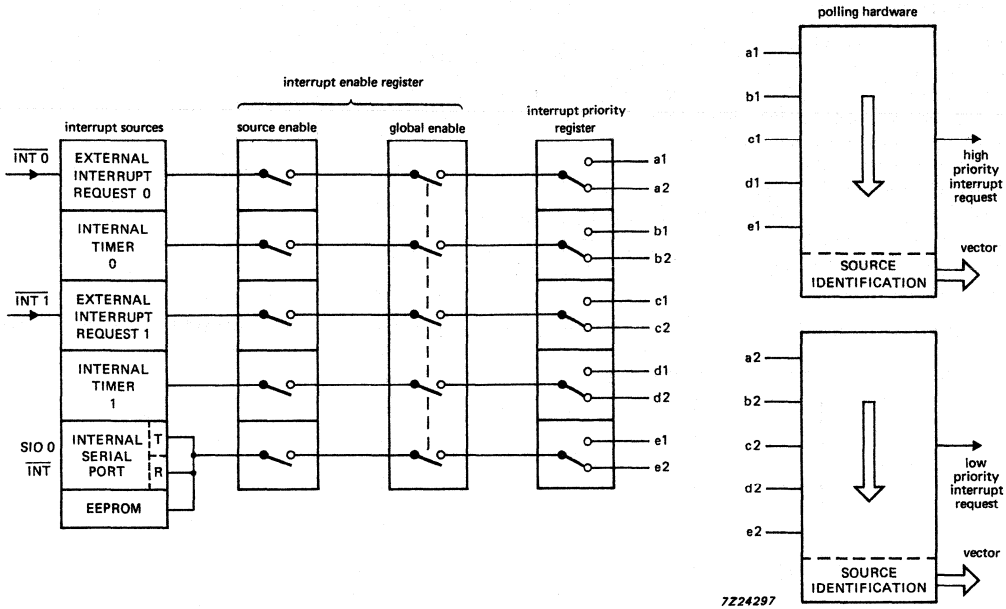


Fig. 9 Interrupt System.

Oscillator circuitry

The oscillator circuitry of the PCB83C851 is a single stage inverting amplifier in a Pierce oscillator configuration. The circuitry between XTAL1 and XTAL2 is basically an invertor biased to the transfer point. Either a crystal or ceramic resonator can be used as the feedback element to complete the oscillator circuitry. Both are operated in parallel resonance. XTAL1 is the high gain amplifier input, and XTAL2 is the output (see Fig. 10).

To drive the PCB83C851 externally, XTAL1 is driven from an external source and XTAL2 left open circuit (see Fig. 10).

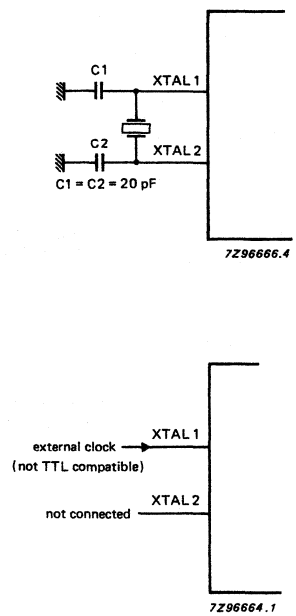


Fig. 10 PCB83C851 Oscillator circuit.

Power-on reset (see Fig. 11)

When V_{DD} is turned on, and provided its rise-time does not exceed 10 ms, an automatic reset can be obtained by connecting the RST pin to V_{DD} via a 2.2 µF capacitor. When the power is switched on, the voltage on the RST pin is equal to V_{DD} minus the capacitor voltage, and decreases from V_{DD} as the capacitor charges through the internal resistor (R_{RST}) to ground. The larger the capacitor, the more slowly V_{RST} decreases. V_{RST} must remain above the lower threshold of the Schmitt trigger long enough to effect a complete reset. The time required is the oscillator start-up time, plus 2 machine cycles.

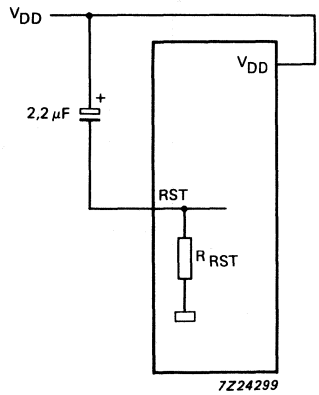


Fig. 11 Power-on reset.

To ensure a correct start-up at power-on, the V_{DD} and RST must come up at the same time; if not, the EEPROM might be effected.

The internal reset is executed during the second cycle in which RST is HIGH and is repeated every cycle until RST goes LOW. It leaves the internal registers as follows:

Register	Content
PC	000H
ACC	00H
B	00H
PSW	00H
SP	07H
DPTR	0000H
P0 – P3	0FFH
IP	(XXX00000)
IE	(0XX00000)
TMOD	00H
TCON	00H
TH0	00H
TL0	00H
TH1	00H
TL1	00H
SCON	00H
SBUF	Intermediate
PCON	(0XXX0000)
EADRH	80H
EADRL	00H
EDAT	XXH
ETIM	08H (for f_{CLK} = 1 MHz)
ECNTRL	00H

X = undefined

The internal RAM is not affected by the reset. When V_{DD} is turned on, the RAM content is indeterminate.

Reset circuitry

This reset circuitry for the PCB83C851 is connected to the reset pin, RST, as shown in Fig. 12. A Schmitt trigger is used as the input for noise rejection. The output of the Schmitt trigger is sampled by the reset circuitry every machine cycle.

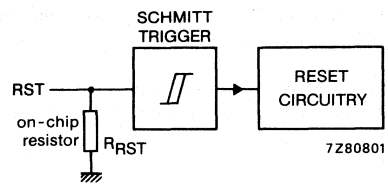


Fig. 12 Reset configuration at RST.

A reset is accomplished by holding the RST pin HIGH for at least two machine cycle (24 oscillator periods), while the oscillator is running. The CPU responds by executing an internal reset. It also configures the ALE and $\overline{PSEN}$ pins as inputs (they are quasi-bidirectional).

INSTRUCTION SET

The PCB83C851 uses a powerful instruction set to allow expansion of on-chip CPU peripherals and to optimize byte efficiency and execution speed. Assigned opcodes add new high-power operations and permit new addressing modes.

The instruction set consists of 49 single-byte, 46 two-byte and 16 three-byte instructions. When using a 12 MHz oscillator, 64 instructions execute in 1 μs and 45 instructions execute in 2 μs. Multiply and divide instructions execute in 4 μs.

Table 4 Instruction description

mnemonic		description	bytes/ cycles	opcode (hex.)
Arithmetic operation				
ADD	A,Rr	Add register to A	1 1	2*
ADD	A,direct	Add direct byte to A	2 1	25
ADD	A,@Ri	Add indirect RAM to A	1 1	26, 27
ADD	A,#data	Add immediate data to A	2 1	24
ADDC	A,Rr	Add register to A with carry flag	1 1	3*
ADDC	A,direct	Add direct byte to A with carry flag	2 1	35
ADDC	A,@Ri	Add indirect RAM to A with carry flag	1 1	36, 37
ADDC	A,#data	Add immediate data to A with carry flag	2 1	34
SUBB	A,Rr	Subtract register from A with borrow	1 1	9*
SUBB	A,direct	Subtract direct byte from A with borrow	2 1	95
SUBB	A,@Ri	Subtract indirect RAM from A with borrow	1 1	96, 97
SUBB	A,#data	Subtract immediate data from A with borrow	2 1	94
INC	A	Increment A	1 1	04
INC	Rr	Increment register	1 1	0*
INC	direct	Increment direct byte	2 1	05
INC	@Ri	Increment indirect RAM	1 1	06, 07
DEC	A	Decrement A	1 1	14
DEC	Rr	Decrement register	1 1	1*
DEC	direct	Decrement direct byte	2 1	15
DEC	@Ri	Decrement indirect RAM	1 1	16, 17
INC	DPTR	Increment data pointer	1 2	A3
MUL	AB	Multiply A & B	1 4	A4
DIV	AB	Divide A by B	1 4	84
DA	A	Decimal adjust A	1 1	D4

INSTRUCTION SET (continued)

mnemonic		description	bytes/ cycles	opcode (hex.)
Logic operations				
ANL	A,Rr	AND register to A	1 1	5*
ANL	A,direct	AND direct byte to A	2 1	55
ANL	A,@Ri	AND indirect RAM to A	1 1	56, 57
ANL	A,#data	AND immediate data to A	2 1	54
ANL	direct,A	AND A to direct byte	2 1	52
ANL	direct,#data	AND immediate data to direct byte	3 2	53
ORL	A,Rr	OR register to A	1 1	4*
ORL	A,direct	OR direct byte to A	2 1	45
ORL	A,@Ri	OR indirect RAM to A	1 1	46, 47
ORL	A,#data	OR immediate data to A	2 1	44
ORL	direct,A	OR A to direct byte	2 1	42
ORL	direct,#data	OR immediate data to direct byte	3 2	43
XRL	A,Rr	Exclusive-OR register to A	1 1	6*
XRL	A,direct	Exclusive-OR direct byte to A	2 1	65
XRL	A,@Ri	Exclusive-OR indirect RAM to A	1 1	66, 67
XRL	A,#data	Exclusive-OR immediate data to A	2 1	64
XRL	direct, A	Exclusive-OR A to direct byte	2 1	62
XRL	direct,#data	Exclusive-OR immediate data to direct byte	3 2	63
CLR	A	Clear A	1 1	E4
CPL	A	Complement A	1 1	F4
RL	A	Rotate A left	1 1	23
RLC	A	Rotate A left through the carry flag	1 1	33
RR	A	Rotate A right	1 1	03
RRC	A	Rotate A right through the carry flag	1 1	13
SWAP	A	Swap nibbles within A	1 1	C4

mnemonic		description	bytes/ cycles	opcode (hex.)
Data transfer				
MOV	A,Rr	Move register to A	1 1	E*
MOV	A,direct **	Move direct byte to A	2 1	E5
MOV	A,@Ri	Move indirect RAM to A	1 1	E6, E7
MOV	A,#data	Move immediate data to A	2 1	74
MOV	Rr,A	Move A to register	1 1	F*
MOV	Rr,direct	Move direct byte to register	2 2	A*
MOV	Rr,#data	Move immediate data to register	2 1	7*
MOV	direct,A	Move A to direct byte	2 1	F5
MOV	direct,Rr	Move register to direct byte	2 2	8*
MOV	direct,direct	Move direct byte to direct byte	3 2	85
MOV	direct,@Ri	Move indirect RAM to direct byte	2 2	86, 87
MOV	direct,#data	Move immediate data to direct byte	3 2	75
MOV	@Ri,A	Move A to indirect RAM	1 1	F6, F7
MOV	@Ri,direct	Move direct byte to indirect RAM	2 2	A6, A7
MOV	@Ri,#data	Move immediate data to indirect RAM	2 1	76, 77
MOV	DPTR,#data16	Load data pointer with a 16-bit constant	3 2	90
MOVC	A,@A+DPTR	Move code byte relative to DPTR to A	1 2	93
MOVC	A,@A+PC	Move code byte relative to PC to A	1 2	83
MOVX	A,@Ri	Move external RAM (8-bit address) to A	1 2	E2, E3
MOVX	A,@DPTR	Move external RAM (16-bit address) to A	1 2	E0
MOVX	@Ri,A	Move A to external RAM (8-bit address)	1 2	F2, F3
MOVX	@DPTR,A	Move A to external RAM (16-bit address)	1 2	F0
PUSH	direct	Push direct byte onto stack	2 2	C0
POP	direct	Pop direct byte from stack	2 2	D0
XCH	A,Rr	Exchange register with A	1 1	C*
XCH	A,direct	Exchange direct byte with A	2 1	C5
XCH	A,@Ri	Exchange indirect RAM with A	1 1	C6, C7
XCHD	A,@Ri	Exchange LOW-order digit indirect RAM with A	1 1	D6, D7

** MOV A, ACC is not a valid instruction.

INSTRUCTION SET (continued)

mnemonic		description	bytes/ cycles	opcode (hex.)
Boolean variable manipulation				
CLR	C	Clear carry flag	1 1	C3
CLR	bit	Clear direct bit	2 1	C2
SETB	C	Set carry flag	1 1	D3
SETB	bit	Set direct bit	2 1	D2
CPL	C	Complement carry flag	1 1	B3
CPL	bit	Complement direct bit	2 1	B2
ANL	C,bit	AND direct bit to carry flag	2 2	82
ANL	C,/bit	AND complement of direct bit to carry flag	2 2	B0
ORL	C,bit	OR direct bit to carry flag	2 2	72
ORL	C,/bit	OR complement of direct bit to carry flag	2 2	A0
MOV	C,bit	Move direct bit to carry flag	2 1	A2
MOV	bit,C	Move carry flag to direct bit	2 2	92
Program and machine control				
ACALL	addr11	Absolute subroutine call	2 2	●1addr
LCALL	addr16	Long subroutine call	3 2	12
RET		Return from subroutine	1 2	22
RETI		Return from interrupt	1 2	32
AJMP	addr11	Absolute jump	2 2	▲1addr
LJMP	addr16	Long jump	3 2	02
SJMP	rel	Short jump (relative address)	2 2	80
JMP	@A+DPTR	Jump indirect relative to the DPTR	1 2	73
JZ	rel	Jump if A is zero	2 2	60
JNZ	rel	Jump if A is not zero	2 2	70
JC	rel	Jump if carry flag is set	2 2	40
JNC	rel	Jump if carry flag is not set	2 2	50
JB	bit,rel	Jump if direct bit is set	3 2	20
JNB	bit,rel	Jump if direct bit is not set	3 2	30
JBC	bit,rel	Jump if direct bit is set and clear bit	3 2	10
CJNE	A,direct,rel	Compare direct to A and jump if not equal	3 2	B5
CJNE	A,#data,rel	Compare immediate to A and jump if not equal	3 2	B4
CJNE	Rr,#data,rel	Compare immed. to reg. and jump if not equal	3 2	B*
CJNE	@Ri,#data,rel	Compare immed. to ind. and jump if not equal	3 2	B6, B7
DJNZ	Rr,rel	Decrement register and jump if not zero	2 2	D*
DJNZ	direct,rel	Decrement direct and jump if not zero	3 2	D5
NOP		No operation	1 1	00

Notes to Table 4

Data addressing modes

Rr	Working register R0-R7.
direct	128 internal RAM locations and any special function register (SFR).
@Ri	Indirect internal RAM location addressed by register R0 or R1 of the actual register bank.
#data	8-bit constant included in instruction.
#data16	16-bit constant included as bytes 2 and 3 of instruction.
bit	direct addressed bit in internal RAM or SFR.
addr16	16-bit destination address. Used by LCALL and LJMP. The branch will be anywhere within the 64 K-byte program memory address space.
addr11	11-bit destination address. Used by ACALL and AJMP. The branch will be within the same 2 K-byte page of program memory as the first byte of the following instruction.
rel	Signed (two's complement) 8-bit offset byte. Used by SJMP and all conditional jumps. Range is −128 to + 127 bytes relative to first byte of the following instruction.

Hexadecimal opcode cross-reference to Table 3

★ : 8, 9, A, B, C, D, E, F.

● : 11, 31, 51, 71, 91, B1, D1, F1.

▲ : 01, 21, 41, 61, 81, A1, C1, E1.

Table 5 Instruction map

first hexadecimal character of opcode ↓ — second hexadecimal character of opcode →

	0	1	2	3	4	5	6 | 7	8 | 9 | A	B	C	D	E	F
0	NOP	AJMP addr11	LJMP addr16	RR A	INC A	INC dir	INC @Ri 0 | 1	INC Rr 0 | 1 | 2	3	4	5	6	7
1	JBC bit,rel	ACALL addr11	LCALL addr16	RRC A	DEC A	DEC dir	DEC @Ri 0 | 1	DEC Rr 0 | 1 | 2	3	4	5	6	7
2	JB bit,rel	AJMP addr11	RET	RL A	ADD A,#data	ADD A,dir	ADD A,@Ri 0 | 1	ADD A,Rr 0 | 1 | 2	3	4	5	6	7
3	JNB bit,rel	ACALL addr11	RETI	RLC A	ADDC A,#data	ADDC A,dir	ADDC A,@Ri 0 | 1	ADDC A,Rr 0 | 1 | 2	3	4	5	6	7
4	JC rel	AJMP addr11	ORL dir,A	ORL dir,#data	ORL A,#data	ORL A,dir	ORL A,@Ri 0 | 1	ORL A,Rr 0 | 1 | 2	3	4	5	6	7
5	JNC rel	ACALL addr11	ANL dir,A	ANL dir,#data	ANL A,#data	ANL A,dir	ANL A,@Ri 0 | 1	ANL A,Rr 0 | 1 | 2	3	4	5	6	7
6	JZ rel	AJMP addr11	XRL dir,A	XRL dir,#data	XRL A,#data	XRL A,dir	XRL A,@Ri 0 | 1	XRL A,Rr 0 | 1 | 2	3	4	5	6	7
7	JNZ rel	ACALL addr11	ORL C,bit	JMP @A+DPTR	MOV A,#data	MOV dir,#data	MOV @Ri,#data 0 | 1	MOV Rr,#data 0 | 1 | 2	3	4	5	6	7
8	SJMP rel	AJMP addr11	ANL C,bit	MOVC A,@A+PC	DIV AB	MOV dir,dir	MOV dir,@Ri 0 | 1	MOV dir,Rr 0 | 1 | 2	3	4	5	6	7
9	MOV DPTR, #data	ACALL addr11	MOV bit,C	MOVC A,@A+DPTR	SUBB A,#data	SUBB A,dir	SUBB A,@Ri 0 | 1	SUBB A,Rr 0 | 1 | 2	3	4	5	6	7
A	ORL C,/bit	AJMP addr11	MOV C,bit	INC DPTR	MUL AB		MOV @Ri,dir 0 | 1	MOV Rr,dir 0 | 1 | 2	3	4	5	6	7
B	ANL C,/bit	ACALL addr11	CPL bit	CPL C	CJNE A, #data,rel	CJNE A,dir,rel	CJNE @Ri,#data,rel 0 | 1	CJNE Rr,#data,rel 0 | 1 | 2	3	4	5	6	7
C	PUSH dir	AJMP addr11	CLR bit	CLR C	SWAP A	XCH A,dir	XCH A,@Ri 0 | 1	XCH A,Rr 0 | 1 | 2	3	4	5	6	7
D	POP dir	ACALL addr11	SETB bit	SETB C	DA A	DJNZ dir,rel	XCHD A,@Ri 0 | 1	DJNZ Rr,rel 0 | 1 | 2	3	4	5	6	7
E	MOVX A,@DPTR	AJMP addr11	MOVX A,@Ri 0 | 1		CLR A	MOV * A,dir	MOV A,@Ri 0 | 1	MOV A,Rr 0 | 1 | 2	3	4	5	6	7
F	MOVX @DPTR,A	ACALL addr11	MOVX @Ri,A 0 | 1		CPL A	MOV dir,A	MOV @Ri,A 0 | 1	MOV Rr,A 0 | 1 | 2	3	4	5	6	7

* MOV A,ACC is not a valid instruction.

RATINGS

Limiting values in accordance with the Absolute Maximum System (IEC134)

Input voltage on any pin with respect to ground (V_{SS})	V_I	−0.5 to + 6.5 V	
Input, output current	$\pm I_I, I_O$	max.	5 mA
Input, output current on any single pin	$\pm I_I, I_O$	max.	10 mA
Total power dissipation	P_{tot}	max.	1 W
Storage temperature range	T_{stg}	−65 to + 150°C	
Operating ambient temperature range	T_{amb} (PCB)	0 to + 70°C	
	(PCF)	−40 to + 85°C	

DC CHARACTERISTICS

V_{DD} = 5 V ± 10%, V_{SS} = 0 V, T_{amb} = 0 to + 70°C (PCB), −40 to + 85°C (PCF); All voltages with respect to V_{SS} unless otherwise specified.

parameter	symbol	min.	max.	unit	conditions
Supply voltage	V_{DD}	4.5	5.5	V	
Supply current					
Operating (note 1)	I_{DD}	–	32	mA	f_{CLK} = 12 MHz
idle mode (note 2)	I_{DD}		12	mA	f_{CLK} = 12 MHz
Power-down current	I_{PD}		100	µA	$2V<V_{DD}<V_{DD}$max (note 3)
Inputs					
LOW level input voltage (except $\overline{EA}$)	V_{IL}	−0.5	$0.2V_{DD}$−0.1	V	
LOW level input voltage ($\overline{EA}$)	V_{IL1}	−0.5	$0.2V_{DD}$−0.3	V	
HIGH level input voltage (except RST and XTAL1)	V_{IH}	$0.2V_{DD}$+0.9	V_{DD}+0.5	V	
HIGH level input voltage (RST and XTAL1)	V_{IH1}	$0.7V_{DD}$	V_{DD}+0.5	V	
Input curr. logic 0 (Port 1, 2 and 3)	$-I_{LI}$	–	50	µA	V_I = 0.45 V
Input curr. logic 1 to 0 transition (Port 1, 2 and 3)	$-I_{TL}$	–	650	µA	V_I = 2.00 V
Input leakage curr. (Port 0, $\overline{EA}$)	$\pm I_{LI}$	–	10	µA	$0.45 < V_I < V_{DD}$
Outputs					
LOW level output voltage (note 4) (Ports 1,2,3)	V_{OL}	–	0.45	V	I_{OI} = 1.6 mA
LOW level output voltage (note 4) (Port 0, ALE, $\overline{PSEN}$)	V_{OL1}	–	0.45	V	I_{OI} = 3.2 V
HIGH level output voltage (Ports 1,2,3)	V_{OH}	2.4	–	V	$-I_{OH}$ = 60 µA V_{DD} = 5 V ± 10%
		$0.75V_{DD}$	–	V	$-I_{OH}$ = 25 µA
		$0.9V_{DD}$	–	V	$-I_{OH}$ = 10 µA

DC CHARACTERISTICS (continued)

parameter	symbol	min.	max.	unit	conditions
HIGH level output voltage (note 5) (Port 0 in external Bus mode, ALE, $\overline{PSEN}$	V_{OH1}	2.4 $0.75V_{DD}$ $0.9V_{DD}$	– – –	V V V	$-I_{OH}$ = 400 μA V_{DD} = 5 V ± 10% $-I_{OH}$ = 150 μA $-I_{OH}$ = 40 μA
RST pull-down resistor	R_{RST}	50	150	kΩ	
Capacitance of I/O Buffer	$C_{I/O}$	–	10	pF	frequency = 1 MHz T_{amb} = 25°C
Erase/write timer constant (note 6) Erase/write cycle time Erase time Write time Data retention time	 $t_{e/w}$ t_e t_w t_s	 20 10 10 10	 100 100 100	 ms ms ms years	 T_{amb} = +55°C

Notes to the DC characteristics

1. The operating supply current is measured with all output pins disconnected; XTAL1 driven with $t_r = t_f = 10$ ns; $V_{IL} = V_{SS} + 0.5V$; $V_{IH} = V_{DD} - 0.5$; XTAL2 not connected; $\overline{EA}$ = RST = Port 0 = V_{DD}.
2. The idle mode supply current is measured with all output pins disconnected; XTAL1 driven with $t_r = t_f = 10$ ns; $V_{IL} = V_{SS} + 0.5V$; $V_{IH} = V_{DD} - 0.5$; XTAL2 not connected; $\overline{EA}$ = RST = Port 0 = V_{DD}.
3. The power-down current is measured with all output pins disconnected; XTAL2 not connected; $\overline{EA}$ = Port 0 = V_{DD}; RST = V_{SS}.
4. Capacitive loading on Port 0 and Port 2 may cause spurious noise pulses to be superimposed on the LOW level output voltage of ALE, Port 1 and Port 3. The noise is due to external bus capacitance discharging into the Port 0 and Port 2 pins when these pins make a 1-to-0 transition during bus operations. In the most adverse condition (capacitive loading > 100 pF), the noise pulse on ALE line may exceed 0.8 V. In this event it may be required to qualify ALE with a Schmitt-trigger, or use an address latch with a Schmitt trigger STROBE input.
5. Capacitive loading on Port 0 and Port 2 may cause the HIGH level output voltage on ALE and $\overline{PSEN}$ to momentarily fall below the $0.9V_{DD}$ specification when the address bits are stabilizing.
6. Endurance (number of erase/write cycles), NE/W, is 10000 E/W cycles.

AC CHARACTERISTICS

V_{DD} = 5 V ± 10%, V_{SS} = 0 V, t_{CK} min. = $1/f_{MAX}$ (maximum operating frequency). T_{amb} = 0 to + 70°C, t_{CK} min. = 83 ns for PCB83C851/80C851. T_{amb} = −40 to + 85°C , t_{CK} min. = 83 ns for PCF83C851/80C851; Cl = 100 pF for Port 0, ALE and $\overline{PSEN}$; Cl = 80 pF for all other outputs; unless otherwise specified.

Parameter	Symbol	MHz		12 MHz		Variable clock		Unit
		min	max	min	max	min	max	
ALE pulse duration	t_{LL}		–	127	–	$2t_{CK}-40$	–	ns
Address set-up time to ALE	t_{AL}		–	28	–	$t_{CK}-55$	–	ns
Address hold time after ALE	t_{LA}		–	48	–	$t_{CK}-35$	–	ns
Time from ALE to valid instruction input	t_{LIV}	–		–	233	–	$4t_{CK}-100$	ns
Time from ALE to control pulse $\overline{PSEN}$	t_{LC}		–	43	–	$t_{CK}-40$	–	ns
Control pulse duration $\overline{PSEN}$	t_{CC}		–	205	–	$3t_{CK}-45$	–	ns
Time from $\overline{PSEN}$ to valid instruction input	t_{CIV}	–		–	145	–	$3t_{CK}-105$	ns
Input instruction hold time after $\overline{PSEN}$	t_{CI}		–	0	–	0	–	ns
Input instruction float delay after $\overline{PSEN}$ (Note 3)	t_{CIF}	–		–	59	–	$t_{CK}-25$	ns
Address to valid instruction input	t_{AIV}	–		–	312	–	$5t_{CK}-105$	ns
Address float time to $\overline{PSEN}$	t_{AFC}	–		–	10	–		ns
$\overline{RD}$ pulse duration	t_{RR}		–	400	–	$6t_{CK}-100$	–	ns
$\overline{WR}$ pulse duration	t_{WW}		–	400	–	$6t_{CK}-100$	–	ns
Address hold time after ALE	t_{LA}		–	48	–	$t_{CK}-35$	–	ns
$\overline{RD}$ to valid data input	t_{RD}	–		–	252	–	$5t_{CK}-165$	ns
Data hold time after $\overline{RD}$	t_{DR}		–	0	–	0	–	ns

AC CHARACTERISTICS (continued)

Parameter	Symbol	MHz		12 MHz		Variable clock		Unit
		min	max	min	max	min	max	
Data float delay after $\overline{RD}$	t_{DFR}	–		–	97	–	$2t_{CK}-70$	ns
Time from ALE to valid data input	t_{LD}	–		–	517	–	$8t_{CK}-150$	ns
Address to valid data input	t_{AD}	–		–	585	–	$9t_{CK}-165$	ns
Time from ALE to $\overline{RD}$ or $\overline{WR}$	t_{LW}			200	300	$3t_{CK}-50$	$3t_{CK}+50$	ns
Time from address to $\overline{RD}$ or $\overline{WR}$	t_{AW}		–	203	–	$4t_{CK}-130$	–	ns
Time from $\overline{RD}$ or $\overline{WR}$ high to ALE high	t_{WHLH}			43	123	$t_{CK}-40$	$t_{CK}+40$	ns
Data valid to $\overline{WR}$ transition	t_{DWX}		–	23	–	$t_{CK}-60$	–	ns
Data set-up time before $\overline{WR}$	t_{DW}		–	433	–	$7t_{CK}-150$	–	ns
Data hold time after $\overline{WR}$	t_{WD}		–	33	–	$t_{CK}-50$	–	ns
Address float delay after $\overline{RD}$	t_{AFR}	–		–	0	–	0	ns

Notes to the AC Characteristics

1. The maximum operating frequency is limited to 12 MHz for all versions.
2. The minimum operating frequency is limited to 1.2 MHz for all versions.
3. Interfacing the PCB83C851 to devices with float times up to 75 ns is permitted. This limited bus contention will not cause damage to Port 0 drivers.

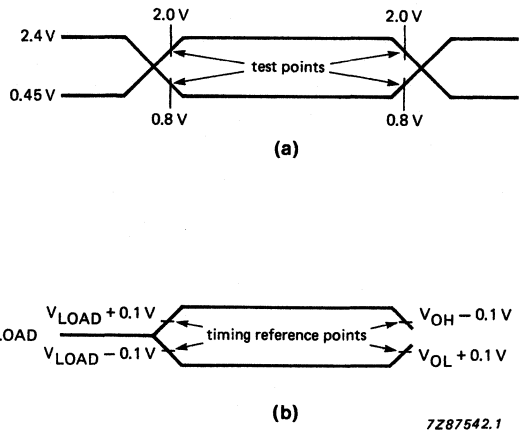


Fig. 13 AC testing input/output waveform (a) and float waveform (b).

AC testing inputs are driven at 2.4 V for a logic 1 and 0.45 V for a logic 0. Timing measurements are taken at 2.0 V for a logic 1 and 0,8 V for a logic 0. The float state is defined as the point at which a Port 0 pin sinks 3.2 mA or sources 400 μA at the test voltage levels.

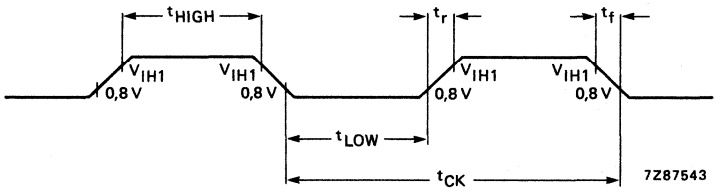


Fig. 14 External clock drive XTAL 1 (see Table 6).

Table 6 External clock drive XTAL1 (see Fig.14)

parameter	symbol	variable clock (f_{XTAL} = 1.2 to 12 MHz)		unit
		min.	max.	
oscillator clock period	t_{CK}	63	833.3	ns
HIGH time	t_{HIGH}	20	$t_{CK}-t_{LOW}$	ns
LOW time	t_{LOW}	20	$t_{CK}-t_{HIGH}$	ns
rise time	t_r	–	20	ns
fall time	t_f	–	20	ns

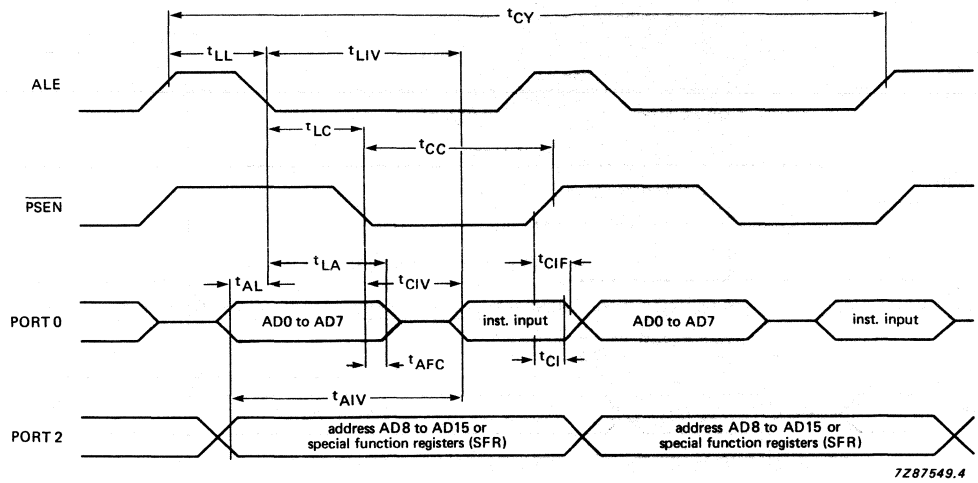


Fig. 15 Read from program memory.

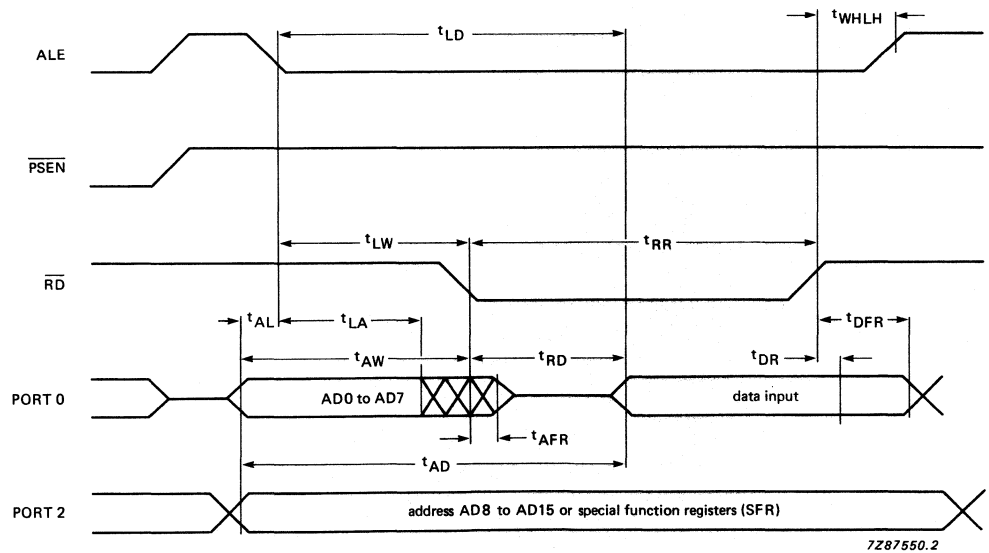


Fig. 16 Read from data memory.

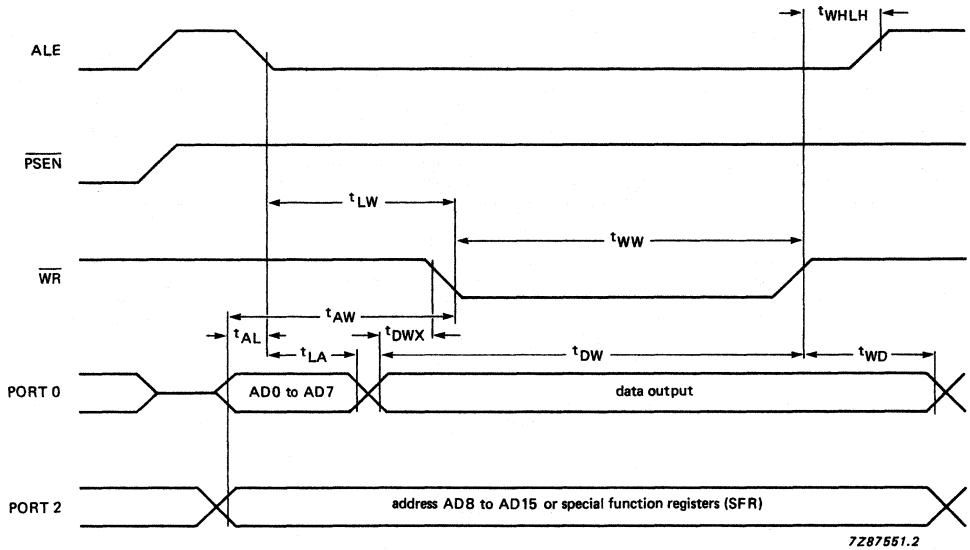


Fig. 17 Write to data memory.

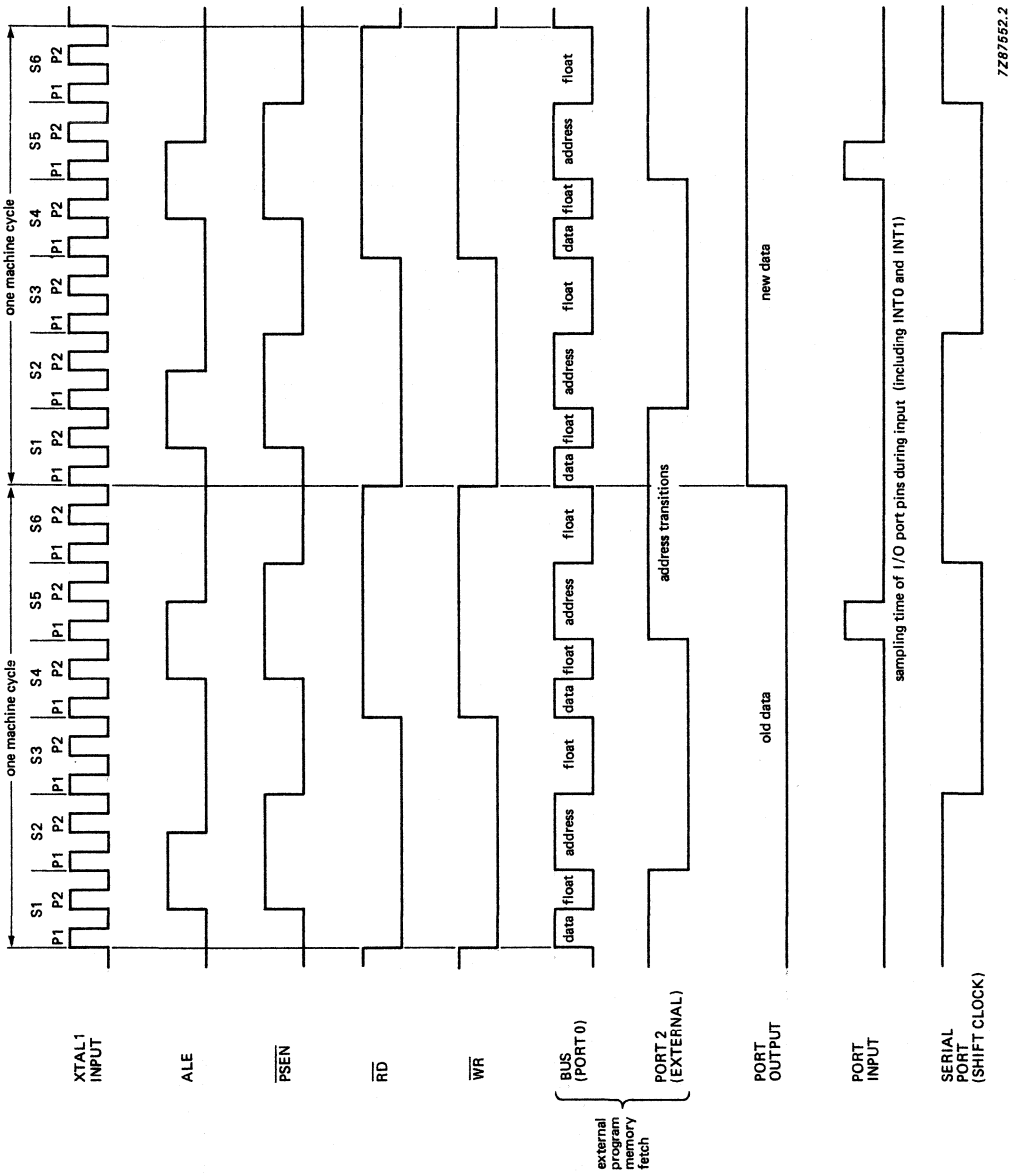


Fig. 18 Instruction timing.

CMOS MICROCONTROLLER FOR TELEPHONE SETS

GENERAL DESCRIPTION

The PCD3315C is a single-chip 8-bit microcontroller fabricated in CMOS and is a member of the PCD3343 family. It has special on-chip features for application in telephone sets.

Features

- 8-bit CPU, ROM, RAM, I/O in a single 28-lead DIL or SO package
- 1536 ROM bytes
- 160 RAM bytes
- 20 quasi-bidirectional I/O port lines
- Two test inputs: one of which is also the external interrupt input (CE/$\overline{T0}$)
- Single-level vectored interrupts: external, timer/event counter
- 8-bit programmable timer/event counter
- Over 80 instructions (based on MAB8048, MAB8400, PCD3343 and PCF8500)
- All instructions 1 or 2 cycles
- Clock frequency 100 kHz to 10 MHz
- Single supply voltage from 1,8 V to 6 V
- Low standby voltage and current
- STOP and IDLE mode
- On-chip oscillator with output drive capability for peripherals
- Configuration of all I/O port lines individually selected by mask: pull-up, open drain or push-pull
- Power-on-reset circuit and low supply voltage detection
- Reset state of all ports individually selected by mask
- Operating temperature range: −25 to + 70 °C

PACKAGE OUTLINES

PCD3315CP: 28-lead DIL; plastic (SOT117).
PCD3315CT: 28-lead mini-pack; plastic (SO28; SOT136A).

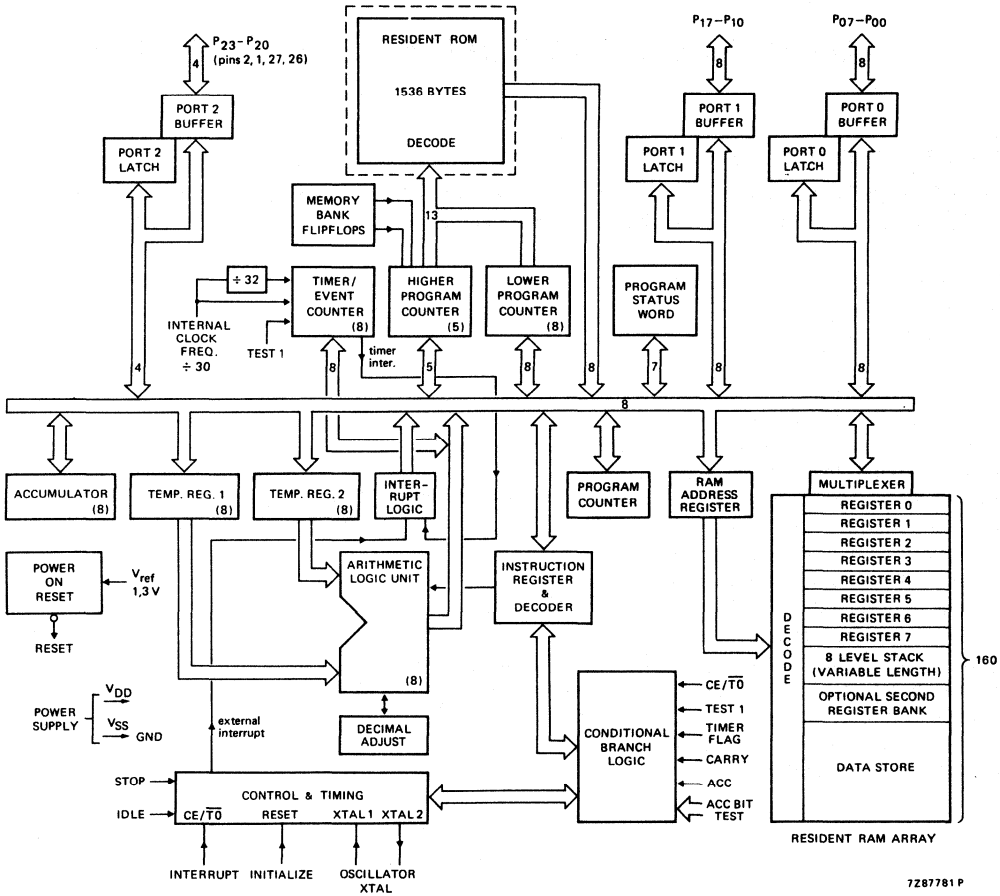
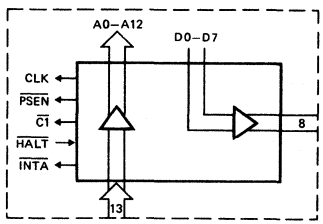
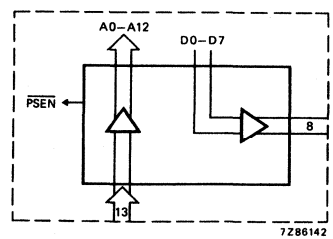


Fig. 1 Block diagram; PCD3315C.



(a)

Fig. 1a Replacement of dotted part in Fig. 1, for the PCD8500F bond-out version.



(b)

Fig. 1b Replacement of dotted part in Fig. 1, for the PCF8500B 'Piggy-back' version.

DEVELOPMENT DATA

PINNING

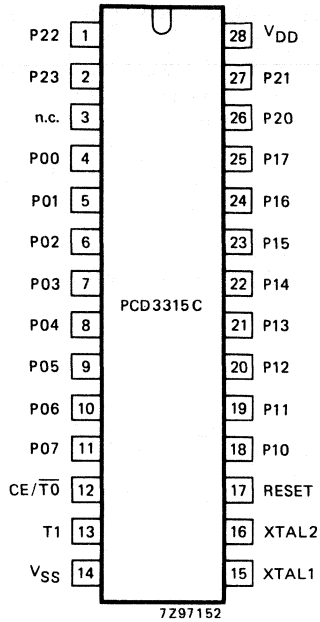


Fig. 2 Pinning diagram: PCD3315C.

PIN DESIGNATION

3	n.c.	not connected
4-11	P00-P07	**Port 0**: 8-bit quasi-bidirectional I/O port.
12	CE/$\overline{T0}$	**Interrupt/Test 0**: external interrupt input (sensitive to positive-going edge edge)/test input pin; when used as a test input directly tested by conditional branch instructions JT0 and JNT0.
13	T1	**Test 1**: test input pin, directly tested by conditional branch instructions JT1 and JNT1. T1 also functions as an input to the 8-bit timer/event counter, using the STRT CNT instruction.
14	V_{SS}	**Ground**: circuit earth potential.
15	XTAL 1	**Crystal input**: connection to timing component (crystal) which determines the frequency of the internal oscillator; also the input for an external clock source.
16	XTAL 2	connection to the other side of the timing component.
17	RESET	**Reset input**: used to initialize the processor (active HIGH), or output of power-on-reset circuit.
18-25	P10-P17	**Port 1**: 8-bit quasi-bidirectional I/O port.
26, 27, 1, 2	P20-P23	**Port 2**: 4-bit quasi-bidirectional I/O port.
28	V_{DD}	**Power supply**: 1,8 V to 6 V.

D.C. CHARACTERISTICS

V_{DD} = 2,5 to 6 V; V_{SS} = 0 V; T_{amb} = −25 to + 70 °C; all voltages with respect to V_{SS}; f = 3,58 MHz with R_S = 50 Ω; unless otherwise specified.

parameter	symbol	min.	typ.	max.	unit
Supply voltage					
operating	V_{DD}	1,8	–	6	V
STOP mode for RAM retention	V_{DD}	1,0	–	6	V
Supply current					
operating					
at V_{DD} = 3 V	I_{DD}	–	350	–	μA
IDLE mode					
at V_{DD} = 3 V	I_{DD}	–	150	–	μA
STOP mode (note 1)					
at V_{DD} = 1,8 V; T_{amb} = 25 °C	I_{DD}	–	1,2	2,5	μA
at V_{DD} = 1,8 V; T_{amb} = 55 °C	I_{DD}	–	–	5	μA
at V_{DD} = 1,8 V; T_{amb} = 70 °C	I_{DD}	–	–	10	μA
RESET I/O					
Switching level	V_{RESET}	–	1,2	–	V
Sink current					
at $V_{DD} > V_{RESET}$	I_{OL}	–	7	–	μA
Inputs					
Input voltage LOW	V_{IL}	0	–	$0{,}3V_{DD}$	V
Input voltage HIGH	V_{IH}	$0{,}7V_{DD}$	–	V_{DD}	V
Input leakage current					
at $V_{SS} < V_I < V_{DD}$	$\pm I_{IL}$	–	–	1	μA
Outputs					
Output voltage LOW					
at $V_I = V_{SS}$ or V_{DD}; $\|I_O\| < 1$ μA	V_{OL}	–	–	0,05	V
Output sink current LOW					
at V_{DD} = 3 V; V_O = 0,4 V	I_{OL}	0,6	1,5	–	mA
Pull-up output source current HIGH					
at V_{DD} = 3 V; $V_O = 0{,}9V_{DD}$	$-I_{OH}$	10	–	–	μA
at V_{DD} = 3 V; $V_O = V_{SS}$	$-I_{OH}$	–	–	200	μA
Push-pull output source current HIGH					
at V_{DD} = 3 V; $V_O = V_{DD}$−0,4 V	$-I_{OH}$	0,6	1,5	–	mA

Note 1

Crystal connected between XTAL 1 and XTAL 2; pin 2 pulled to V_{DD} via 5,6 kΩ resistor; CE and T1 at V_{SS}.

CMOS MICROCONTROLLER FOR TELEPHONE SETS

GENERAL DESCRIPTION

The PCD3343 is a single-chip 8-bit microcontroller fabricated in CMOS. It has special on-chip features for application in telephone sets.
The device is mask programmable, designed to provide telephone dialling facilities such as redial, repertory dial, emergency call, keyboard scan and control for liquid crystal display, pulse dial and/or DTMF dial via dedicated peripheral.

Features

- 8-bit CPU, ROM, RAM, I/O in a single 28-lead DIL or SO package
- 3 K ROM bytes
- 224 RAM bytes
- 20 quasi-bidirectional I/O port lines
- Two test inputs: one of which is also the external interrupt input (CE/$\overline{T0}$)
- Single-level vectored interrupts: external, timer/event counter, serial I/O
- Serial I/O which can be used in bus systems with more than one master (serial I/O data via an existing port line and clock via a dedicated line)
- 8-bit programmable timer/event counter
- Over 80 instructions (based on MAB8048, MAB8400 and PCF8500)
- All instructions 1 or 2 cycles
- Clock frequency 100 kHz to 10 MHz
- Single supply voltage from 1,8 V to 6 V
- Low standby voltage and current
- STOP and IDLE mode
- On-chip oscillator with output drive capability for peripherals (e.g. PCD3312 DTMF generator)
- Configuration of all I/O port lines individually selected by mask: pull-up, open drain or push-pull
- Power-on-reset circuit and low supply voltage detection
- Reset state of all ports individually selected by mask
- Operating temperature range: −25 to + 70 °C

PACKAGE OUTLINES

PCD3343P: 28-lead DIL; plastic (SOT117).
PCD3343D: 28-lead DIL; ceramic (CERDIP) (SOT135A).
PCD3343T: 28-lead mini-pack; plastic (SO28; SOT136A).

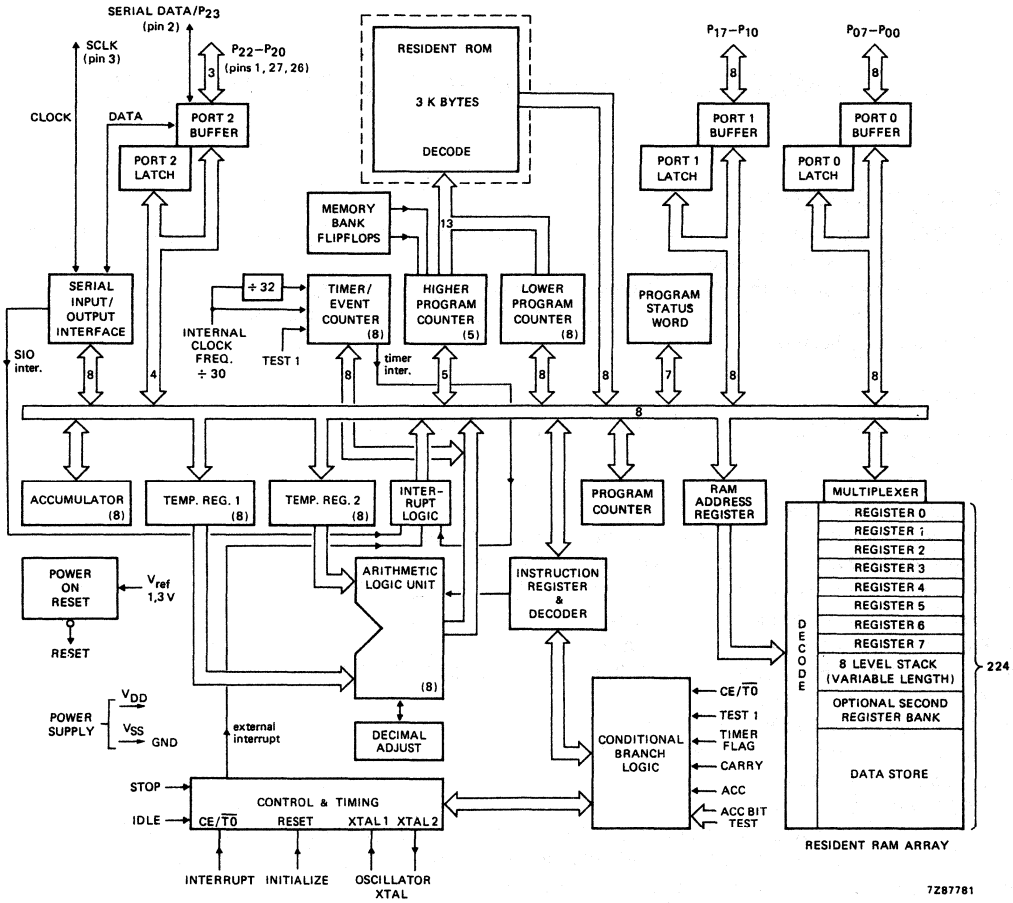
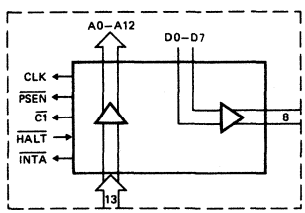
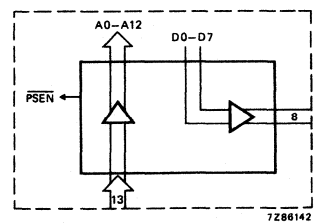


Fig. 1 Block diagram; PCD3343.



(a)

Fig. 1a Replacement of dotted part in Fig. 1, for the PCF8500F bond-out version.

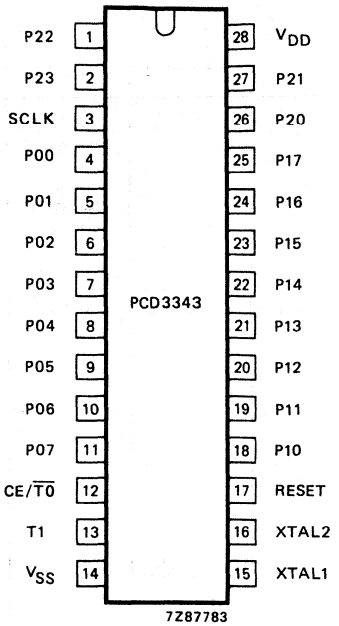


(b)

Fig. 1b Replacement of dotted part in Fig. 1, for the PCF8500B 'Piggy-back' version.

DEVELOPMENT DATA

PINNING



Note CE/$\overline{T0}$ is labelled $\overline{INT}$/T0 on the PCF8500B and has inverted polarity.

Fig. 2 Pinning diagram: PCD3343 and bottom pinning PCF8500B.

PIN DESIGNATION

3	SCLK	**Clock**: bidirectional clock for serial I/O.
4-11	P00-P07	**Port 0**: 8-bit quasi-bidirectional I/O port.
12	CE/$\overline{T0}$	**Interrupt/Test 0**: external interrupt input (sensitive to positive-going edge)/test input pin; when used as a test input directly tested by conditional branch instructions JT0 and JNT0.
13	T1	**Test 1**: test input pin, directly tested by conditional branch instructions JT1 and JNT1. T1 also functions as an input to the 8-bit timer/event counter, using the STRT CNT instruction.
14	V_{SS}	**Ground**: circuit earth potential.
15	XTAL 1	**Crystal input**: connection to timing component (crystal) which determines the frequency of the internal oscillator; also the input for an external clock source.
16	XTAL 2	connection to the other side of the timing component.
17	RESET	**Reset input**: used to initialize the processor (active HIGH), or output of power-on-reset circuit.
18-25	P10-P17	**Port 1**: 8-bit quasi-bidirectional I/O port.
26, 27, 1, 2	P20-P23	**Port 2**: 4-bit quasi-bidirectional I/O port. P23 is the serial data input/output in serial I/O mode.
28	V_{DD}	**Power supply**: 1,8 V to 6 V.

PINNING (continued)

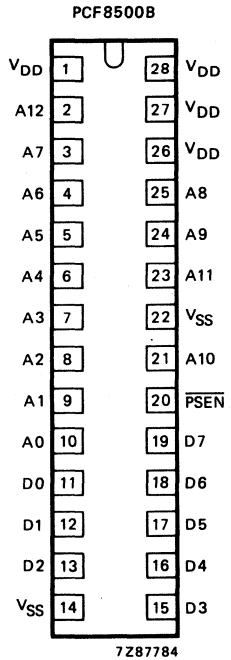


Fig. 3 Pinning diagram: PCF8500B 'Piggy-back' version top pinning; to access a 2732 or 2764 EPROM.

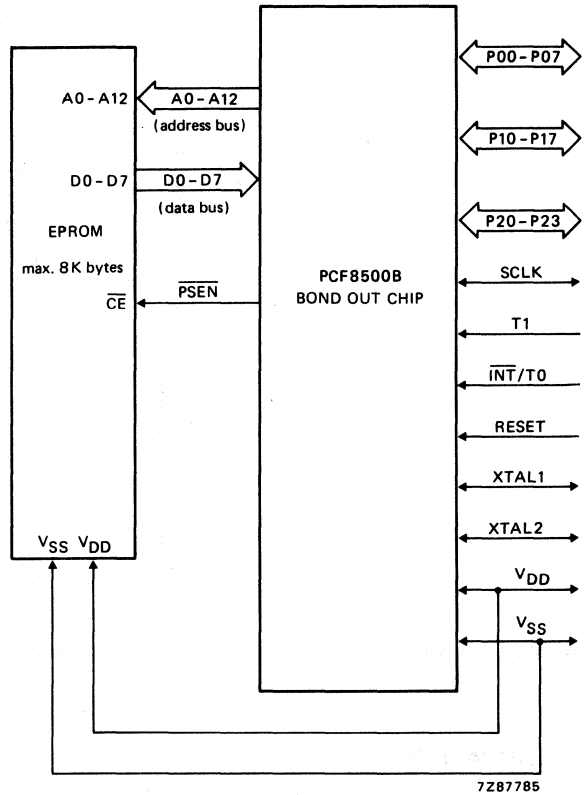


Fig. 3a Connection of EPROM to 'Piggy-back' package PCF8500B.

PIN DESIGNATION

14, 22	V_{SS}	**Ground**
1, 26-28	V_{DD}	**Power supply**
10-3, 25, 24, 21, 23, 2	A0-A12	**Address outputs**
11-13, 15-19	D0-D7	**Data**
20	$\overline{PSEN}$	**Program store enable**

Notes

1. RAM capacity of PCF8500B is 256 bytes.
2. Access time for ROMS/EPROMS to be below 7 x $1/f_{XTAL}$.
3. Pin 12 CE/$\overline{T0}$ is on the PCF8500B, inverted and labelled $\overline{INT}$/T0.

FUNCTIONAL DESCRIPTION

Bond-out version PCF8500F

The PCF8500F is a microcontroller that contains no on-board ROM, but has all address and data lines brought-out to access an external ROM or EPROM. This version has more pins than the PCD3343 with on-board ROM (see Fig. 1a). The RAM has 256 bytes. It can address 8 K bytes of ROM.

'Piggy-back' version PCF8500B

The PCF8500B is a special package that has standard pinning to the bottom which facilitates insertion as a mask-programmed device. An EPROM can be mounted on top in an additional socket. The total package height is greater than the standard DIL package. The RAM has 256 bytes and can also address 8 K bytes of program memory.

Program memory PCD3343

The program memory consists of 3072 bytes (8-bit words), in a read-only memory (ROM). Each location is directly addressable by the program counter. The memory is mask-programmed at the factory. Figure 4 shows the program memory map.

Four program memory locations are of special importance:

- Location 0; contains the first instruction to be executed after the processor is initialized (RESET),
- Location 3; contains the first byte of an external interrupt service subroutine,
- Location 5; contains the first byte of a serial I/O interrupt service subroutine,
- Location 7; contains the first byte of a timer/event counter interrupt service subroutine.

Program memory is arranged in banks of 2 K bytes, which are selected by SEL MB instructions. The program memory is further divided into location 'pages', each of 256 bytes. This latter division applies only for conditional branches. Memory bank boundaries can be crossed only by using the unconditional branch instructions after the appropriate memory bank has been selected. A CALL instruction can transfer control to a subroutine on any 'page'; RET and RETR instructions can transfer control from a subroutine back to the main program.

Data memory PCD3343

Data memory consists of 224 bytes (8-bit words), random-access data memory (RAM). All locations are indirectly addressable using RAM pointer registers; up to 16 designated locations are directly addressable. Memory also includes an 8-level program counter stack addressed by a 3-bit stack pointer. Figure 5 shows the data memory map.

Working registers

Locations 0 to 7 are designated as working registers, directly addressable by the direct register instructions. Ease of addressing, and a minimum requirement of instruction bytes to manipulate their contents, makes these locations suitable for storing frequently addressed intermediate results. This bank of registers can be selected by the SEL RB0 instruction.

Executing the select register bank instruction SEL RB1, designates locations 24 to 31 as working registers, instead of locations 0 to 7, and these are then directly addressable. This second bank of working registers may be used as an extension of the first or reserved for use during interrupt service subroutines saving the first bank for use in the main program. If the second bank is not used, locations 24 to 31 may serve as general purpose RAM.

The first locations of each bank contain the RAM pointer registers R0, R1, R0' and R1', which indirectly address all RAM locations.

All RAM locations make efficient program loop counters when used with the decrement register and test instruction DJNZ.

FUNCTIONAL DESCRIPTION (continued)

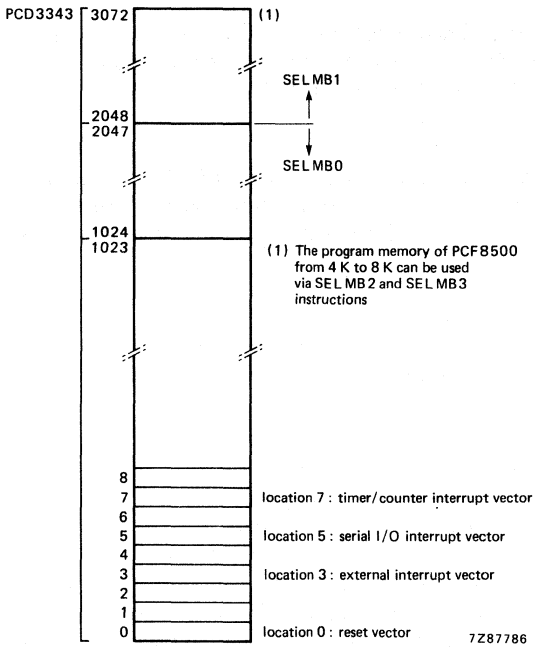


Fig. 4 Program memory map.

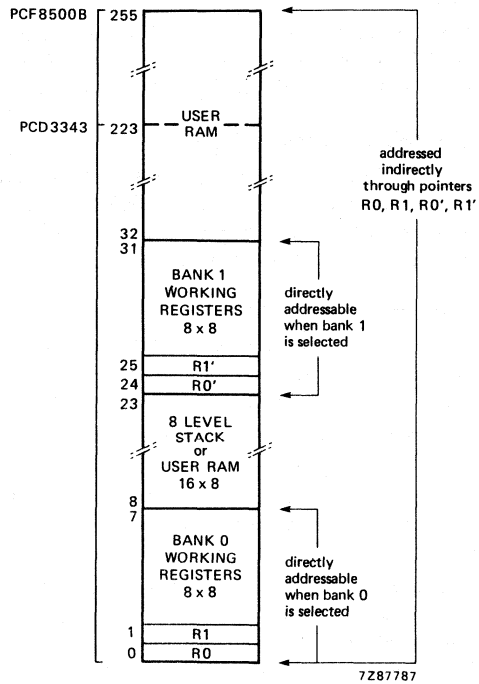


Fig. 5 Data memory map.

Program counter stack

Locations 8 to 23 may be designated as an 8-level program counter stack (2 locations per level), or as general purpose RAM. The program counter stack (Fig. 6) enables the processor to keep track of the return addresses and status generated by interrupts or CALL instructions by storing the contents of the program counter prior to servicing the subroutine. A 3-bit stack pointer determines which of the eight register pairs of the program counter stack will be loaded with next generated return address.

The stack pointer, when initialized to 000 by RESET, points to RAM locations 8 and 9. On the first subroutine CALL or interrupt, the contents of the program counter and bits 4, 6 and 7 of the program status word (PSW) are transferred to locations 8 and 9. The stack pointer increments by one and points to locations 10 and 11 ready for another CALL. Because an address may be up to 13 bits long, two bytes must be used to store each address.

At the end of a subroutine, which is signalled by a return instruction (RET or RETR), the stack pointer decrements by one and the contents of the register pair on top of the stack are transferred to the program counter. The saved PSW bits are transferred to the PSW only by the RETR instruction.

If not all 8 levels of subroutine and interrupt nesting are used, the unused portion of the stack may be used as any other indirectly addressable RAM locations. Possible locations from 32 to 223 may be used for storage of program variables or data.

Nesting of subroutines within subroutines can continue up to 8 times without overflowing the stack. If overflow does occur the deepest address stored (locations 8 and 9) will be overwritten and lost since the stack pointer overflows from 111 to 000. It also underflows from 000 to 111.

The value of the saved contents of the program counter is different for an interrupt CALL compared to a normal CALL to subroutine. With an interrupt CALL, the program counter return address is saved; with a subroutine CALL, the saved program counter value is one less than the program counter return address.

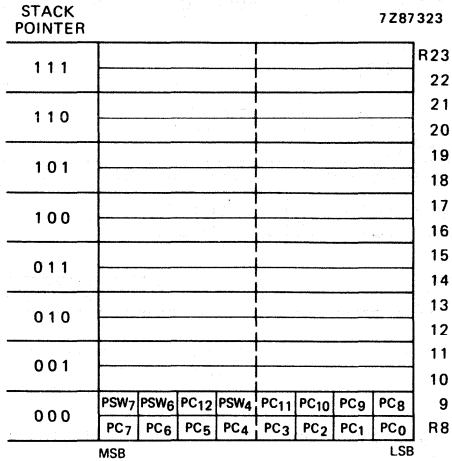


Fig. 6 Program counter stack.

IDLE and STOP modes

IDLE mode

When the microcontroller enters the IDLE mode via the IDLE instruction (H'01') the oscillator, timer/counter and serial I/O are kept running. The microcontroller exits from the IDLE mode by one of three interrupts if they are enabled or by activating a RESET. If the interrupt is not enabled the processor will remain in the IDLE mode. An active signal on the RESET pin restarts the microcontroller and a normal RESET sequence is executed (see Fig. 7).

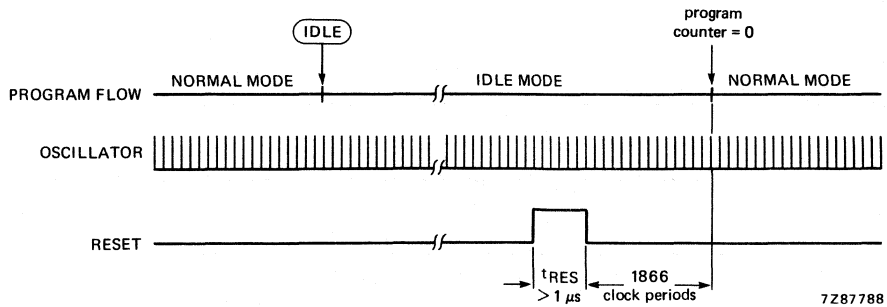


Fig. 7 Exit from IDLE mode via a RESET.

FUNCTIONAL DESCRIPTION (continued)

An active signal coming from an enabled interrupt causes the execution of the normal interrupt routine since normal interrupt scanning is still being carried out. A LOW-to-HIGH transition on the external interrupt pin (CE/$\overline{T0}$) reactivates the microcontroller. A HIGH level applied to CE/$\overline{T0}$ will reactivate the microcontroller only in the STOP mode. Thus, if CE/$\overline{T0}$ was HIGH before the microcontroller entered the IDLE mode, it must go LOW before the microcontroller can be reactivated (see Fig. 8).

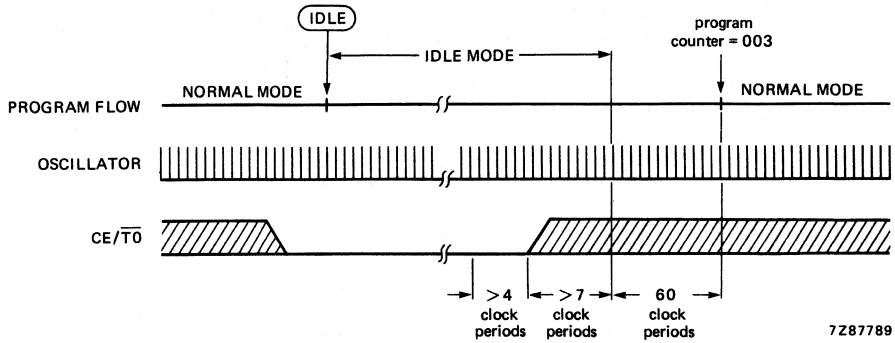


Fig. 8 Exit from IDLE mode via an interrupt.

Wake-up from the IDLE mode is ensured when CE/$\overline{T0}$ is LOW for 4 CP (clock periods) followed by a HIGH for 7 CP. After the initial forced CALL H'003' operation (60 CP) the program continues with the external interrupt service routine.

STOP mode

The microcontroller enters the STOP mode by the STOP instruction (H'22'). The oscillator is switched off. The internal status of the CPU, RAM contents and the state of I/O ports are not affected. The microcontroller can be brought-out of the STOP mode by an active signal at the external interrupt input or by an external RESET signal. When one of these two signals is applied an internal delay of 1866 CP is provided to ensure that all internal clocks are operating correctly before restart (see Fig. 9).

If the microcontroller exits from the STOP mode by activating RESET, a normal RESET sequence is executed.

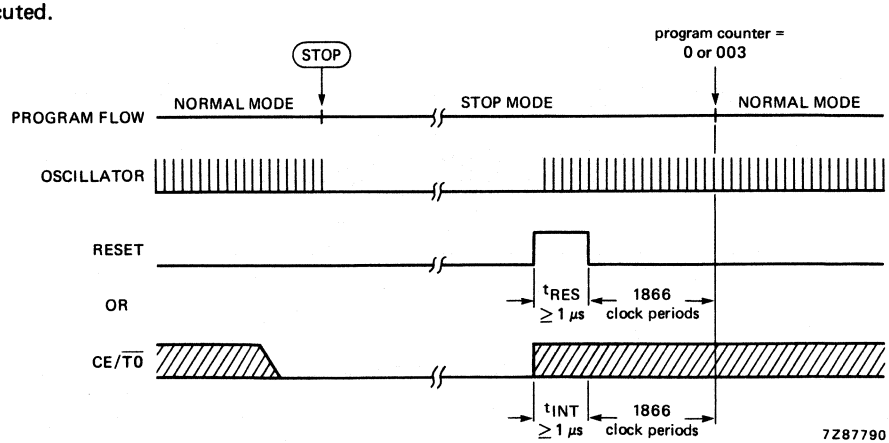


Fig. 9 Entering and exiting the STOP mode.

If the microcontroller exits the STOP mode by pulling the external interrupt input pin HIGH, an interrupt sequence is executed only if the external interrupt is enabled. In this event the microcontroller resumes the normal program sequence after returning from the interrupt routine, as in the normal mode. If the interrupt is not enabled, it continues the normal program sequence, executing the instruction following the STOP instruction.

The microcontroller is restarted by a HIGH level applied at the CE/$\overline{T0}$ pin, and not by a LOW-to-HIGH transition as in a normal interrupt mechanism.

When the CE/$\overline{T0}$ level is active during the STOP instruction then no STOP is executed.

A HIGH level on the external interrupt input of at least 1 μs will cause the microcontroller to exit the STOP mode.

I/O facilities

The PCD3343 family has 23 I/O lines arranged as:

- Port 0 parallel port of 8 lines (P00 to P07)
- Port 1 parallel port of 8 lines (P10 to P17)
- Port 2 parallel port of 4 lines (P20 to P23)
- SCLK serial I/O consisting of a data line shared with a parallel port line (P23) and a separate clock line SCLK
- CE/$\overline{T0}$ external interrupt and test input. When used as a test input can be directly tested by conditional branch instructions JT0 and JNT0
- T1 test input which can alter program sequences when tested by conditional jump instructions JT1 and JNT1. T1 also functions as an input to the 8-bit timer/event counter.

Parallel ports

All parallel ports can be used as outputs or inputs, their structure is quasi-bidirectional.
Output data written to a port is latched and remains unchanged until rewritten.
Input data is not latched and so must be present until read by an input instruction.
Input lines are fully CMOS compatible, output lines can drive one LS-TTL or CMOS load.

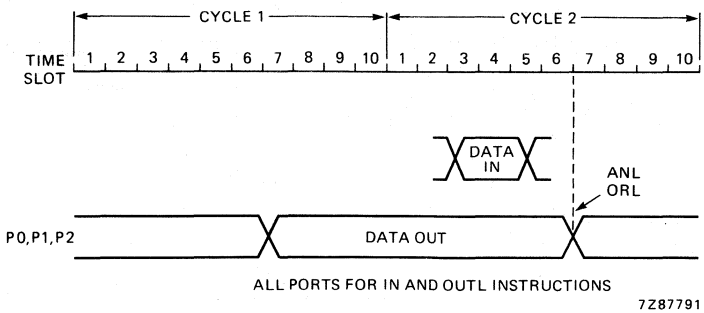


Fig. 10 Timing diagram of all ports on IN and OUTL instructions; for ANL and ORL instructions, the ports change on the time slot 7 of cycle 2.

Fig. 11 shows the quasi-bidirectional I/O interface with push-pull output and switched pull-up current source.
Each line is pulled up to V_{DD} via a constant current source (TR4), which is enabled via TR3 whenever one of the two output latches contains a logic 1. This current provides sufficient source current for a TTL HIGH level, yet can be pulled LOW by an external CMOS device, thus allowing the same pin to be used for both input and output.

FUNCTIONAL DESCRIPTION (continued)

When a logic 1 is written to the line for the first time (MQ = 1, SQ = 0), TR2 is switched on for the duration of the internal write pulse (one oscillator period), to provide a fast transition from logic 0 to logic 1. Subsequent writing of a logic 1 to the port lines will not switch TR2 on. This prevents unnecessary current through external components connected to the port lines of the same port which might be in the input mode and also connected to ground.

When a logic 0 is written to the line, TR3 switches off the current source. Current sinking capability is provided by TR1, which is now switched on. When used as an input, a logic 1 must first be written to the line, otherwise TR1 will remain low impedance.

In telephone applications this switched pull-up source may not be sufficient. Therefore the PCD3343 offers the possibility to select individually 19 of the 20 parallel port pins (not P23), by the following mask options:

Option 1- STANDARD PORT; quasi-bidirectional I/O with switched pull-up current source of 100 μA (typ.) and P-channel booster transistor TR2 (1,5 mA). TR2 is only active during 1 clock cycle (0,28 μs at 3,58 MHz).

Option 2- OPEN DRAIN; quasi-bidirectional I/O with only an N-channel open drain output. Application as an output requires connection of an external pull-up resistor (Fig. 12).

Option 3- PUSH-PULL OUTPUT; drive capability of the output will be 1,5 mA (typ.) at V_{DD} = 3 V in both polarities. To avoid a large current flowing through the output transistors during the input mode, these push-pull pins must only be used as outputs (Fig. 13).

Also, individual mask selection of the RESET state of these port pins can be achieved by appending the following options S and R to options 1, 2 or 3.

Option S-SET; after RESET this pin will be initialized to HIGH.

Option R-RESET; after RESET this pin will be initialized to LOW.

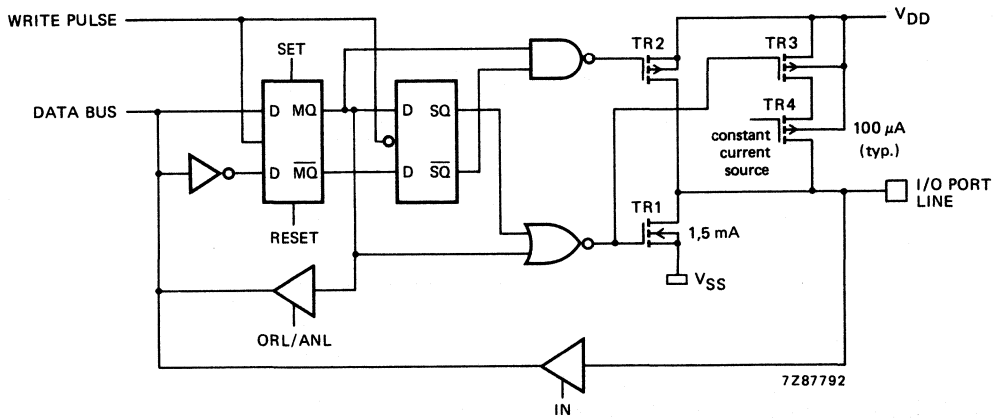


Fig. 11 Standard output with switched pull-up current source.

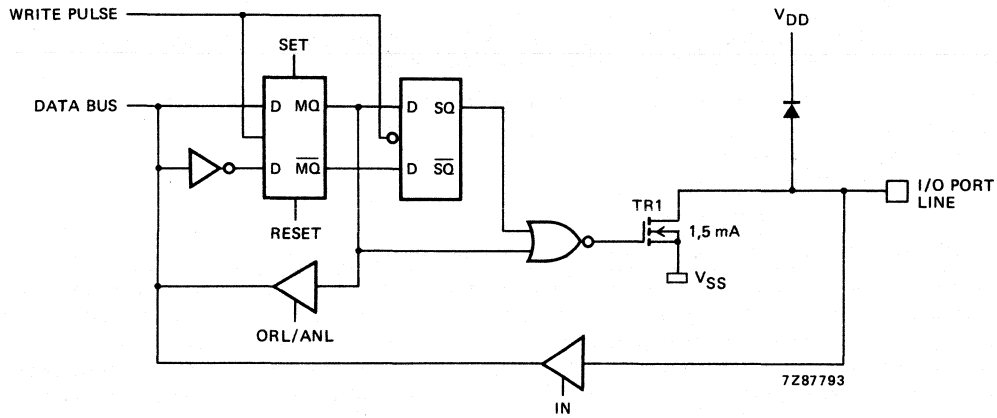


Fig. 12 Open drain output.

DEVELOPMENT DATA

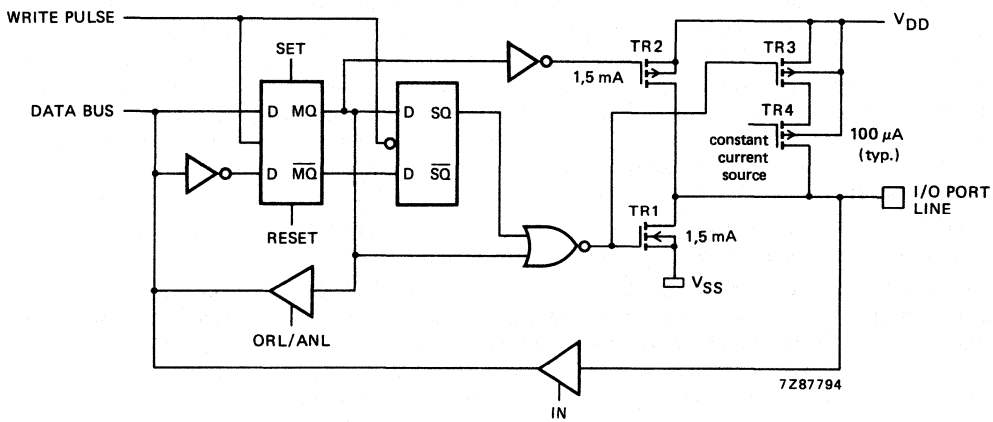


Fig. 13 Push-pull output.

FUNCTIONAL DESCRIPTION (continued)

Serial I/O (SIO)

The PCD3343 has an on-chip serial I/O interface. This SIO interface is a versatile feature in an intelligent telephone set, as shown in application diagram Fig. 32.

In this application the SIO is used to communicate with the different peripherals, such as:

- DTMF generator (PCD3312)
- LCD drivers (PCF8577)
- External RAM (PCD8571)
- Clock calendar (PCB8573)

No extra hardware is required for decoding, addressing and data processing.

Whereas a normal microcontroller must regularly monitor the serial data bus for the presence of data, the serial I/O interface detects, receives and converts the serial data stream into parallel format without interrupting the execution of the current program. An interrupt is sent to the PCD3343 only when a complete byte is received. It then reads the data byte in one instruction. Likewise during transmission the serial I/O interface performs parallel to serial conversion and subsequent serial output of the data. The microcontroller is only interrupted in the execution of its programmed tasks when a complete byte has been transmitted.

The design of the PCD3343 serial I/O system allows any number of devices from PCF8500 family (clips) to be connected via the two-line serial bus. The ability of any devices to communicate, without interrupting the operation of any other devices on the bus, is an outstanding attribute of the system. This is achieved by allocating a specific 7-bit address to each device and providing a system whereby a device reacts only to a message prefixed with its own address or the 'general CALL' address. Address recognition is performed by the interface hardware so that operation of the microcontroller need only be interrupted when a valid address has been received. This saves significant processing time and memory space compared with a conventional microcontroller employing a software serial interface. When the addressing facility is not required, for instance in a system with only two microcontrollers, direct data transfer without addressing can be performed. In multi-master systems, an automatically invoked arbitration procedure prevents two or more devices from continuing simultaneous transmission.

In NORMAL (running) and IDLE mode, the serial I/O logic remains active; its internal system clock will be switched off when there is no activity on the serial bus.

After execution of the STOP instruction, the oscillator of the PCD3343 is switched off. This means that the serial I/O logic will remain in the state it was at the occurance of the STOP instruction. To avoid "bus block" problems and to assure correct start-up of the bus after exit from the STOP mode, the user should disable the serial logic (ESO = 0) prior to the execution of the STOP instruction. This must be carried out only when the PCD3343 has finished a serial data transfer.

Serial I/O interface

Figure 14 shows the serial I/O interface. The clock line of the serial bus has exclusive use of pin 3 (SCLK) while the data line shares pin 2 (serial data) with the I/O line P23 of port 2. When the serial I/O is enabled, P23 is disabled as a parallel port line; (P23 and SCLK only open drain).

The microcontroller and interface communicate via the internal microcontroller bus and the Serial Interrupt Request line. Data and information controlling the operation of the interface are stored in four registers:

- Data shift register (S0)
- Serial I/O interface status word (S1)
- Serial clock control word (S2)
- Address register

Data shift register (S0)

Register S0 converts serial data to parallel format and vice versa. A pending interrupt is generated only after a complete byte has been transmitted, or after a complete data byte, specific address or 'general CALL' address has been received. The most significant bit is transmitted first.

Serial I/O interface status word (S1)

Register S1 provides information concerning the state of the interface and stores information from the microcontroller. Bits 0 to 3 are duplicated: control bits in these positions can only be written by the microcontroller, while interface bits can only be read.

MST and TRX (see Table 1)

These bits determine the operating mode of the serial I/O interface.

Table 1 Operating modes of the serial I/O interface

MST	TRX	operating mode
0	0	slave receiver
1	0	master receiver
0	1	slave transmitter
1	1	master transmitter

BB: Bus Busy.

This is the flag which indicates the status of the bus.

PIN: Pending Interrupt Not

PIN = '0' indicates the presence of a pending interrupt, which will cause a Serial Interrupt Request when the serial interrupt mechanism is enabled.

ESO: Enable Serial output

The ESO flag enables/disables the serial I/O interface: ESO = '1' enables, ESO = '0' disables. ESO can only be written by software.

BC0, BC1 and BC2

Bits BC0, BC1 and BC2 indicate the number of bits received or transmitted in a data stream. These bits can only be written by software.

AL: Arbitration Lost

The arbitration lost flag is set by hardware when the serial I/O interface, as master transmitter, loses a bus arbitration procedure.

AAS: Addressed As Slave

This flag is set by hardware when the interface detects either its own specific address or the 'general CALL' address as the first byte of a transfer and the interface has been programmed to operate in the address recognition mode.

AD0: Address Zero

This flag is set by hardware after detection of the 'general CALL' address when the interface is operating in the address recognition mode.

LRB: Last Received Bit

This contains either the last data bit received or, for a transmitting device in the acknowledgement mode, the acknowledgement signal from the receiving device.

Bits AL, AAS, AD0 and LRB can only be read by software.

DEVELOPMENT DATA

FUNCTIONAL DESCRIPTION (continued)

Serial clock control word (S2)

Bits 0 to 4 of the clock control register S2 are used to set the frequency of the serial clock signal. When a 3,58 MHz crystal is used, the frequency of the serial clock can be varied between 92 kHz and 580 Hz (see Table 2). An asymmetrical clock with a HIGH-to-LOW ratio of 3 : 1 can be generated using bit 5. The asymmetrical clock allows a microcontroller more time per clock period for sampling the data line, making the timing of this action less critical. Bit 6 can be used to activate the acknowledge mode of the serial I/O. S2 is a write only register.

Address register

The address register contains the 7-bit address back-up latches and the bit (ALS) used to enable/disable the address recognition mode. The address register can be written using the MOV S0, A and MOV S0, # data instructions, but only when ES0 = '0'.

Serial I/O interrupt logic

An EN SI instruction enables and a DIS SI instruction disables the interrupt logic. When the logic is enabled, a pending interrupt results in a serial I/O interrupt to the processor, causing a CALL to location 5 in the ROM. When disabled, the presence of an interrupt is still indicated by PIN in S1, allowing the interrupt to be serviced. However, vectored interrupt will not occur.

DEVELOPMENT DATA

Table 2 SIO clock pulse frequency control when using a 3,58 MHz crystal

hexadecimal S20-S24 code	divisor	f_{SCLK} (kHz) (approximate)
0	not allowed	
1	39	92
2	45	80
3	51	70
4	63	57
5	75	48
6	87	41
7	99	36
8	123	29
9	147	24
A	171	21
B	195	18
C	243	15
D	291	12
E	339	11
F	387	9,2
10	483	7,4
11	579	6,2
12	675	5,3
13	771	4,6
14	963	3,7
15	1155	3,1
16	1347	2,7
17	1539	2,3
18	1923	1,9
19	2307	1,6
1A	2691	1,3
1B	3075	1,2
1C	3843	0,93
1D	4611	0,78
1E	5379	0,67
1F	6147	0,58

FUNCTIONAL DESCRIPTION (continued)

Table 3 Serial I/O addresses for telephony peripherals

type	address 7	6	5	4	3	2	1	0	description
PCF8570A	1	0	1	0	A2	A1	X	R/$\overline{W}$	2 K RAM
PCF8570	1	0	1	0	A2	A1	A0	R/$\overline{W}$	2 K RAM
PCD8571	1	0	1	0	A2	A1	A0	R/$\overline{W}$	1 K RAM
PCD3311	0	1	0	0	1	0	A0	R/$\overline{W}$	DTMF dialler
PCD3312	0	1	0	0	1	0	A0	R/$\overline{W}$	DTMF dialler
PCE2111	0	0	0	0	0	0	1	0	LCD driver *
PCD8573	1	1	0	1	0	A1	A0	R/$\overline{W}$	clock calendar
PCF8574	0	0	1	1	A2	A1	A0	R/$\overline{W}$	8-bit I/O expander
PCF8576	0	1	1	1	0	0	SA0	R/$\overline{W}$	1 : 4 LCD driver
PCF8577	0	1	1	1	0	1	0	R/$\overline{W}$	1 : 2 LCD driver

* LCD driver requires an additional enable line.

DEVELOPMENT DATA

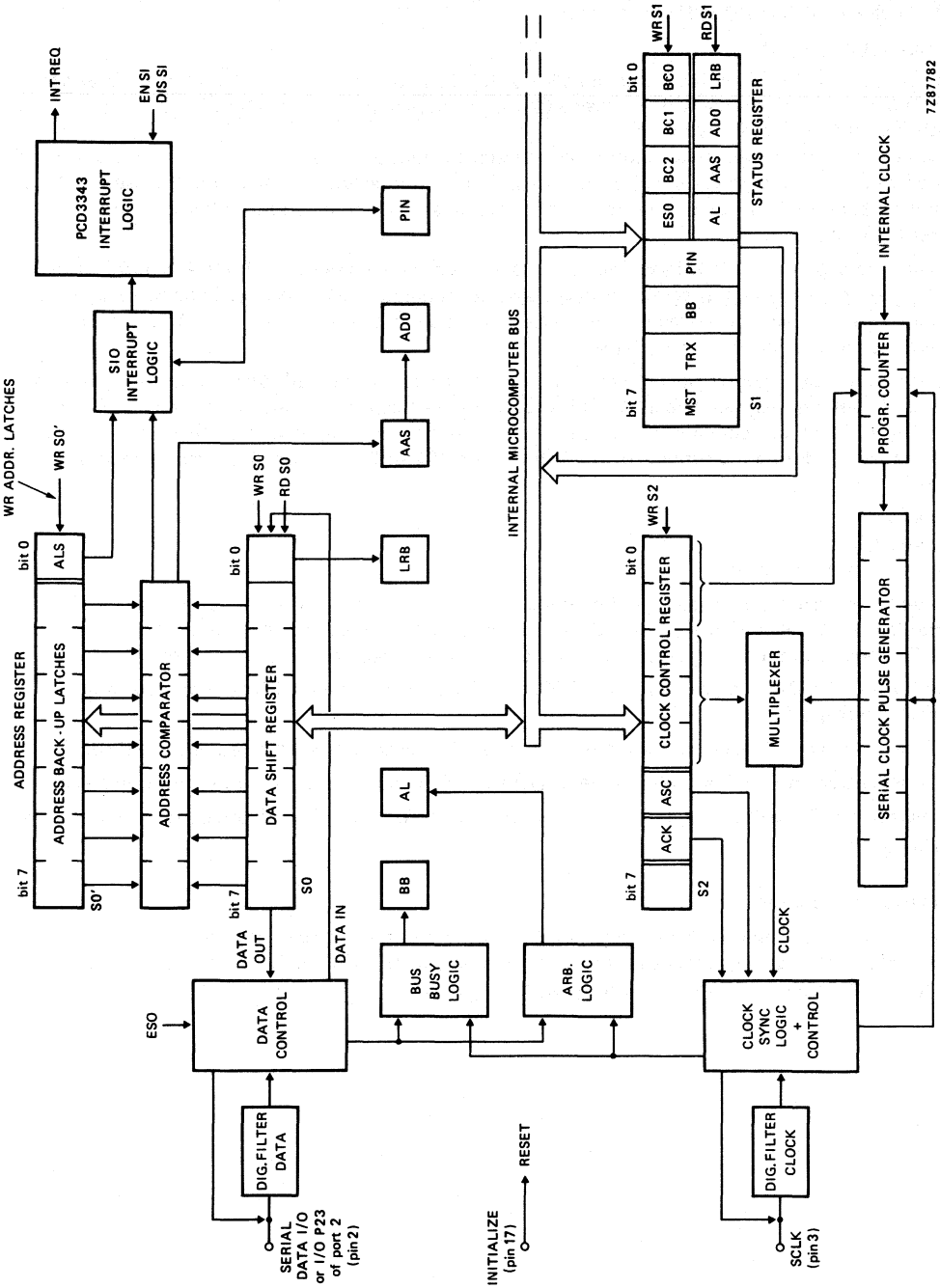


Fig. 14 Serial I/O interface.

FUNCTIONAL DESCRIPTION (continued)

Interrupts (see Fig. 15)

When the external interrupt is enabled, a LOW-to-HIGH transition on the CE/$\overline{T0}$ input initiates an external interrupt subroutine which causes a CALL to program memory location 3 following completion of the current instruction.
The interrupt must remain enabled until the interrupt instruction is completed, otherwise the next instruction of the main program will be executed. Serial I/O interrupt, when enabled, causes a CALL to location 5, and a timer/event counter overflow forces a CALL to location 7 when the timer interrupt is enabled.

When an interrupt subroutine starts, the contents of the program counter and bits 4, 6 and 7 of the PSW have been saved in the program counter stack. Accumulator contents have to be saved by software. Interrupt acknowledgement can be carried out by software via port pins. All interrupt subroutines must reside in memory bank 0.

Since the interrupt system is single level, once an interrupt is detected, all further interrupt requests are latched, but ignored, pending a RETR instruction to re-enable the interrupt input logic. After executing RETR, the program continues in the main part; this is independent of the occurance of a second interrupt during the running of the first routine. If 2 or 3 interrupts occur simultaneously, their priority is:

- (1) external
- (2) serial I/O
- (3) timer/event counter

An external interrupt can be generated by using the timer/counter in the event counter mode. The counter is first preset to (H'FF'), then EN TCNTI instruction is executed. A LOW-to-HIGH transition of the T1 input will then initiate an interrupt subroutine and cause a CALL to location 7.

On execution of a DIS I instruction, the PCD3343 always clears the digital filter/latch and the External Interrupt Flag.

The Timer Flag (TF) is reset only when the JTF or JNTF instruction is executed or after RESET.

The Timer Interrupt Flag is set when timer overflow occurs, only if the timer interrupt is enabled.

The microcontroller will exit the IDLE mode when any one of the following three interrupts is enabled:

- External
- Serial I/O
- Timer/event counter

There is no internal pull-up or pull-down device connected to the external interrupt input (pin 12). If required pin 12 must be externally connected to a resistor ($R \leqslant 100\ k\Omega$). When the external interrupt is not used pin 12 must be connected to V_{SS}.

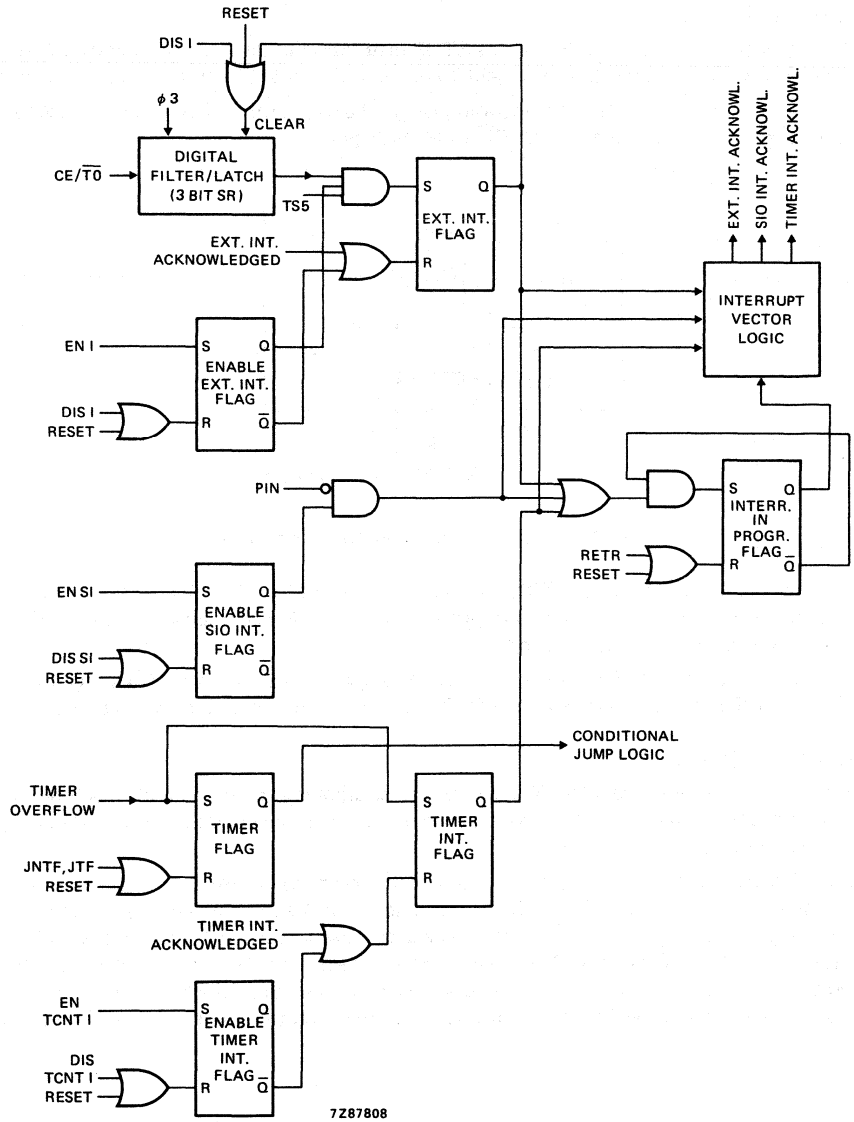


Fig. 15 Interrupt logic.

Notes to figure 15

1. CE/$\overline{T0}$ positive edge is always latched in the digital filter/latch.
2. Correct interrupt timing in ensured when CE/$\overline{T0}$ is LOW for >4 CP followed by a HIGH for >7 CP.
3. When the interrupt in progress flag is set, further interrupts are latched but ignored, until RETR is executed.
4. A DIS I instruction always clears a pending external interrupt.

FUNCTIONAL DESCRIPTION (continued)

Oscillator (see Fig. 16)

The 3,58 MHz oscillator can be inhibited by the STOP instruction under software control. It is also inhibited when a low-voltage condition is present to prevent discharge of a weak back-up battery.

Provided the supply voltage is within the operating range, the oscillator will be restarted after a STOP instruction by a HIGH level at either the CE/$\overline{T0}$ or RESET pin.

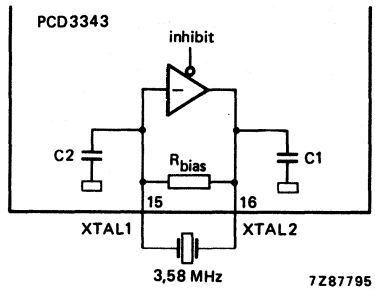


Fig. 16 Oscillator with integrated elements.

The oscillator has the output drive capability for the DTMF generator (PCD3311/3312) via pin 16 (XTAL 2). An external clock can be applied to pin 15 (XTAL 1). A machine cycle consists of 10 time slots, each time slot being 3 oscillator periods.

In telephony applications the 3,58 MHz crystal provides a 8,4 μs machine cycle. The range of the clock frequency is from 100 kHz up to a maximum which is a function of the supply voltage (see Fig. 23).

Timer/event counter (see Fig. 17)

An internal 8-bit up-counter is provided. This can count external events, modulo-32 machine cycles, or machine cycles directly. Table 4 gives the instructions that control the counter and the prescaler, and the functions performed.

When used as a timer, the input to the counter is either the overflow or input of a 5-bit prescaler. When used as an event counter, LOW-to-HIGH transitions on pin 13 (T1) are counted. The maximum rate at which the counter may be incremented is once every machine cycle (182,6 kHz for a 8,4 μs machine cycle). When the counter overflows, the timer flag is set. The flag can be tested and reset using the JTF (jump if timer flag = 1) or JNTF instruction. Overflow also generates an interrupt to the processor via setting of the Timer Interrpt Flag when the timer/event counter interrupt is enabled.

Table 4 Timer/event counter control

function	timer mode modulo-1, modulo-32*	counter mode
CLEAR	MOV T,A (A) = 0 or RESET	MOV T,A (A) = 0 or RESET
PRESET	MOV T,A	MOV T,A
START	STRT T	STRT CNT
STOP	STOP TCNT or RESET	STOP TCNT or RESET
TEST	JTF/JNTF	JTF/JNTF
READ**	MOV A,T	MOV A,T

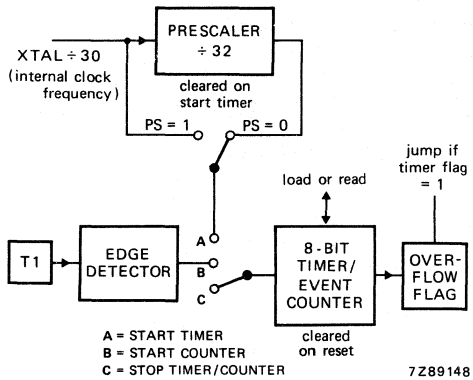


Fig. 17 Timer/event counter.

DEVELOPMENT DATA

Program status word (see Fig. 18)

The program status word (PSW) is an 8-bit word (1 byte) in the CPU which stores information about the current status of the microcontroller.
The PSW bits are:

- Bits 0 to 2 — stack pointer bits (SP_0, SP_1, SP_2)
- Bit 3 — prescaler select (PS);
  0 = modulo-32; 1 = modulo-1 (no prescaling)
- Bit 4 — working register bank select (RBS);
  0 = register bank 0; 1 = register bank 1
- Bit 5 — not used (1)
- Bit 6 — auxiliary carry (AC); half-carry bit generated by an ADD instruction and used by the decimal adjust instruction DA A
- Bit 7 — carry (CY); the carry flag indicates that previous operation has resulted in an overflow of the accumulator.

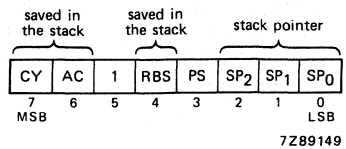


Fig. 18 Program status word.

* With prescaler select, PS = 0, the timer counts modulo-32 machine cycles, with PS = 1 it counts modulo-1 cycles (prescaler not used); prescaler cleared with STRT T, prescaler not readable.

** READ does not disturb the counting process.

FUNCTIONAL DESCRIPTION (continued)

Program status word (continued)

All bits can be read using the MOV A, PSW instruction. Bits 7 and 6 are set and cleared by CPU operation. Bit 4 can be changed by a SEL RB instruction, bit 3 by the MOV PSW, A instruction, and bits 0, 1 and 2 by the CALL, RET or RETR instructions and in the event of an interrupt. Bits 7, 6 and 4 are stored in the program counter stack during subroutine and interrupt calls. These bits are restored in the PSW with a RETR (return and restore) instruction which must be used at the end of an interrupt and can be used at the end of a normal subroutine. The RET instruction has no restore feature and cannot be used at the end of an interrupt.

Program counter (see Fig. 19)

A 13-bit program counter is used to facilitate 8 K bytes of ROM being addressed. The arrangement of the bits is shown in figure 19. During an interrupt subroutine PC_{11} and PC_{12} are forced to logic 0. All 13 bits are saved in the stack during CALL and interrupt routines.

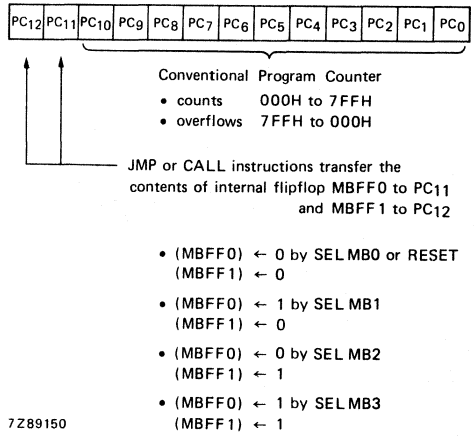


Fig. 19 Program counter.

Central processing unit

The PCD3343 has arithmetic, logical and branching capabilities. The DA A, SWAP A and XCHD instructions simplify BCD arithmetic and the handling of nibbles. The MOVP A,@A instruction permits efficient table look-up from the current ROM page.

Conditional branch logic

The conditional branch logic within the processor enables several conditions, internal and external to the processor, to be tested by the user's program. Table 5 lists the conditional jump instructions used to change the program sequence. The DJNZ instruction decrements a designated register or data memory location and branches if the contents are not zero. This instruction is useful for looping control. The JMPP@A instruction allows multiway branches to destinations determined by the contents of the accumulator.

Table 5 Conditional branches

test	jump condition	jump instruction
accumulator	all bits zero	JZ
	any bit non-zero	JNZ
accumulator bit test	1	JB0 to JB7
carry flag	1	JC
	0	JNC
timer overflow flag	1	JTF
	0	JNTF
test input T0	1	JNT0
	0	JT0*
test input T1	1	JT1
	0	JNT1
register	non-zero	DJNZ

Test input T1 (pin 13)

The T1 input line can be used as:

- A test input for branch instructions JT1 and JNT1
- An external input to the event counter

When used as a test input:

- JT1 instruction tests for logic 1 level
- JNT1 instruction tests for logic 0 level

When used as an input to the event counter, T1 must be LOW for > 4 CP, followed by a HIGH for > 4 CP. The transition can be recognized with a repetition rate of once per 30 oscillator clock periods (1 machine cycle).

There is no internal pull-up or pull-down resistor connected to the T1 input. If required it must be externally connected to a resistor ($R = \leqslant 100\ k\Omega$). When T1 is not used pin 13 must be connected to V_{DD} or V_{SS}.

Reset (pin 17)

A positive-going signal on the RESET input/output:

- Sets the program counter to zero
- Selects location 0 of memory band 0 and register bank 0
- Sets the stack pointer to zero (000); pointing to RAM address 8
- Disables the interrupts (external, timer and serial I/O)
- Stops the timer/event counter, then sets it to zero
- Sets the timer prescaler to modulo-32
- Resets the timer flag
- Sets all ports according to reset states
- Sets the serial I/O to slave receiver mode and disables the serial I/O
- Cancels IDLE and STOP mode

After the voltage is applied to RESET an internal delay of 1866 CP is introduced before the microcontroller commences operation.

* Because of the inverted interrupt input CE/$\overline{T0}$ the conditional jump JT0 is also inverted.

FUNCTIONAL DESCRIPTION (continued)

Power-on-reset and low-voltage detection (see Fig. 20)

In telephony applications, correct operation of the PCD3343 during moments of slowly changing supply voltages and low-voltage conditions is essential. This is achieved by the addition of an internal power-on-reset and low-voltage detection circuit.
To allow an external RESET signal being fed into the PCD3343, the reset pin (pin 17) has been configured as an input/output.

While a reset condition exists in the detection circuit, pin 17 is pulled HIGH by TR1 controlled by the reset circuit.
When the reset condition is not present a pull-down current source (TR2) will be activated. TR2 forces pin 17 LOW thus removing the RESET signal from the microcontroller.

Since the level at pin 17 is recognized by the microcontroller, the reset time constant can be stretched by connecting an external capacitor between V_{DD} and pin 17 (see Fig. 22).

The signal at pin 17 can also be used as an output to reset other devices in the system.

The internal reset circuit monitors the PCD3343 supply voltage. If the voltage drops below the switching level (typ. 1,3 V), a reset (HIGH) is applied to pin 17. This reset is removed (pin 17 goes LOW), after a fixed delay (t_d), when the supply voltage rises above the switching level again. The delay ensures a complete reset even when the supply voltage quickly rises above switching level after initial switch-on.

During a low-voltage condition the oscillator is inhibited to prevent complete discharge of a weak battery. The timing of the power-on-reset and low-voltage detection circuit is shown in figure 21.

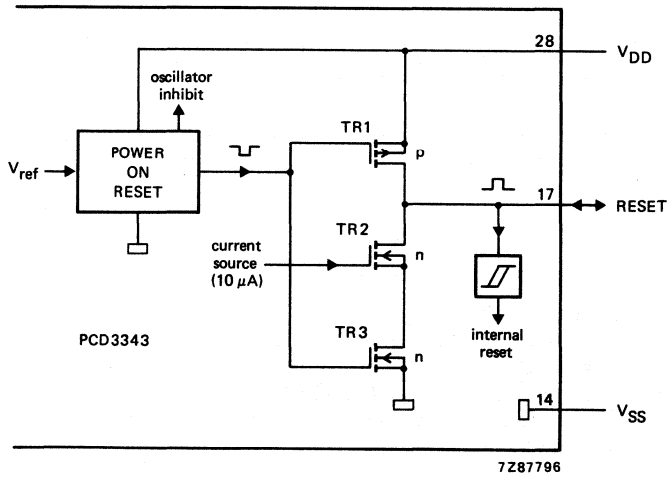
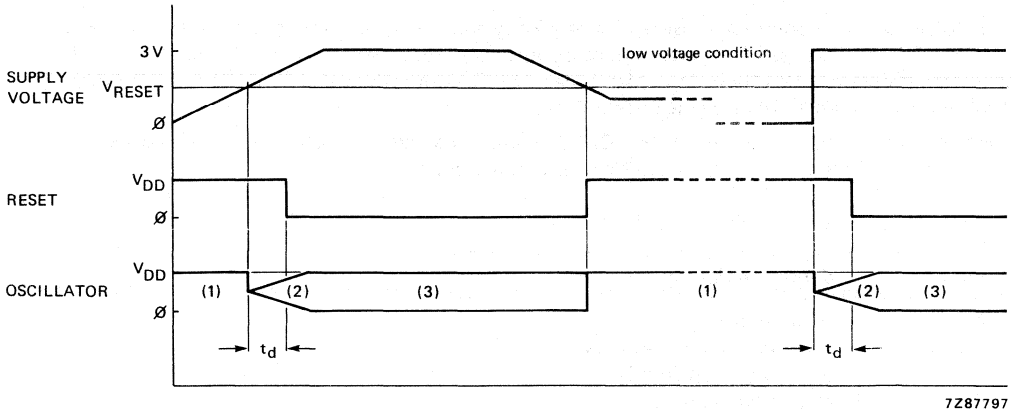


Fig. 20 Power-on-reset configuration.



Where: (1) Oscillator inhibited
(2) Oscillator starting
(3) Oscillator running, but may be stopped with a STOP condition

Fig. 21 Timing of power-on-reset and low-voltage detection.

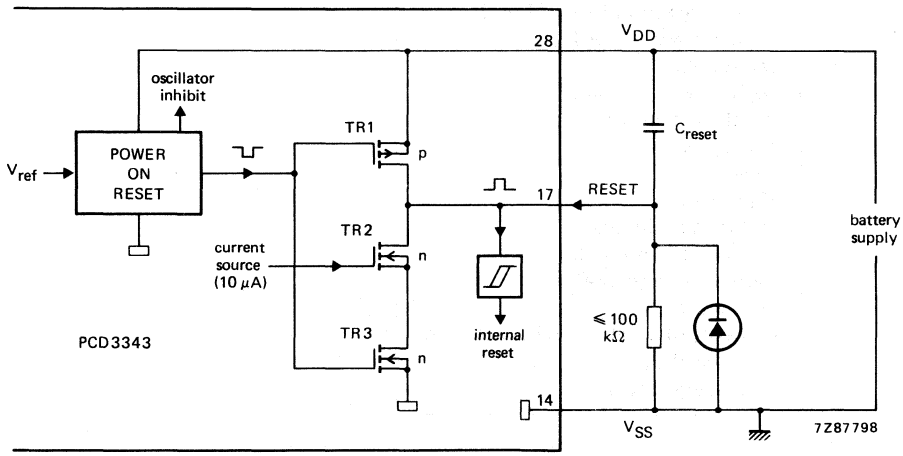


Fig. 22 Stretched power-on-reset with external capacitor.

INSTRUCTION SET

The PCD3343 instruction set consists of over 80 one and two byte instructions and is based on the MAB8048 instruction set. New instructions include those for serial I/O operation and memory bank selection. Program code efficiency is high because all RAM locations and all ROM locations on a 256 byte page require only a single byte address.

Table 8 gives the instruction set of the PCD3343. Table 7 shows the instruction map and Table 6 details the symbols and definition descriptions that are used.

Table 6 Symbols and definitions used in Table 8

symbol	definition description
A	accumulator
addr	program memory address
Bb	bit designation (b = 0-7)
RBS	register bank select
C	carry bit (bit CY)
CNT	event counter
D	mnemonic for 4-bit digit (nibble)
data	8-bit number or expression
I	interrupt
MB	memory bank
MBFF	memory bank flip-flop
P	mnemonic for 'in-page' operation
PC	program counter
Pp	port designation (p = 0, 1 or 2)
PSW	program status word
RB	register bank
Rr	register designation (r = 0-7)
Sn	serial I/O register
SP	stack pointer
T	timer
TF	timer flag
T0, T1	test 0 and 1 inputs
#	immediate data prefix
@	indirect address prefix
(X)	contents of X
((X))	contents of location addressed by X
←	is replaced by
↔	is exchanged with

Table 7 PCD3343 instruction map

first hexadecimal character of opcode

second hexadecimal character of opcode

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	NOP	IDLE		ADD A,#data	JMP page 0	EN I	JNTF addr	DEC A	IN A,Pp 0 1 2				MOV A,Sn 0 1			
1	INC @Rr 0 1		JBO addr	ADDC A,#data	CALL page 0	DIS I	JTF addr	INC A	INC Rr 0 1 2 3 4 5 6 7							
2	XCH A,@Rr 0 1		STOP	MOV A,#data	JMP page 1	EN TCNTI	JNTO addr	CLR A	XCH A,Rr 0 1 2 3 4 5 6 7							
3	XCHD A,@Rr 0 1		JB1 addr		CALL page 1	DIS TCNTI	JTO addr	CPL A	OUTL Pp,A 0 1 2				MOV Sn,A 0 1 2			
4	ORL A,@Rr 0 1		MOV A,T	ORL A,#data	JMP page 2	STRT CNT	JNT1 addr	SWAP A	ORL A,Rr 0 1 2 3 4 5 6 7							
5	ANL A,@Rr 0 1		JB2 addr	ANL A,#data	CALL page 2	STRT T	JT1 addr	DA A	ANL A,Rr 0 1 2 3 4 5 6 7							
6	ADD A,@Rr 0 1		MOV T,A		JMP page 3	STOP TCNT		RRC A	ADD A,Rr 0 1 2 3 4 5 6 7							
7	ADDC A,@Rr 0 1		JB3 addr		CALL page 3			RR A	ADDC A,Rr 0 1 2 3 4 5 6 7							
8				RET	JMP page 4	EN SI			ORL Pp,#data 0 1 2							
9			JB4 addr	RETR	CALL page 4	DIS SI	JNZ addr	CLR C	ANL Pp,#data 0 1 2				MOV Sn,#data 0 1 2			
A	MOV @Rr,A 0 1			MOVP A,@A	JMP page 5	SEL MB2		CPL C	MOV Rr,A 0 1 2 3 4 5 6 7							
B	MOV @Rr,#data 0 1		JB5 addr	JMPP @A	CALL page 5	SEL MB3			MOV Rr,#data 0 1 2 3 4 5 6 7							
C	DEC @Rr 0 1				JMP page 6	SEL RB0	JZ addr	MOV A,PSW	DEC Rr 0 1 2 3 4 5 6 7							
D	XRL A,@Rr 0 1		JB6 addr	XRL A,#data	CALL page 6	SEL RB1		MOV PSW,A	XRL A,Rr 0 1 2 3 4 5 6 7							
E	DJNZ @Rr,addr 0 1				JMP page 7	SEL MB0	JNC addr	RL A	DJNZ Rr, addr 0 1 2 3 4 5 6 7							
F	MOV A,@Rr 0 1		JB7 addr		CALL page 7	SEL MB1	JC addr	RLC A	MOV A,Rr 0 1 2 3 4 5 6 7							

INSTRUCTION SET (continued)

Table 8 Instruction set

	mnemonic	opcode (hex.)	bytes/ cycles	description	function		notes
ACCUMULATOR	ADD A, Rr	6*	1/1	Add register contents to A	(A)←(A) + (Rr)	r = 0–7	1
	ADD A, @Rr	60 61	1/1	Add RAM data, addressed by Rr, to A	(A)←(A) + ((R0)) (A)←(A) + ((R1))		1
	ADD A, #data	03 data	2/2	Add immediate data to A	(A)←(A) + data		1
	ADDC A, Rr	7*	1/1	Add carry and register contents to A	(A)←(A) + (Rr) + (C)	r = 0–7	1
	ADDC A, @Rr	70 71	1/1	Add carry and RAM data, addressed by Rr, to A	(A)←(A) + ((R0)) + (C) (A)←(A) + ((R1)) + (C)		1
	ADDC A, #data	13 data	2/2	Add carry and immediate data to A	(A)←(A) + data + (C)		1
	ANL A, Rr	5*	1/1	'AND' Rr with A	(A)←(A) AND (Rr)	r = 0–7	
	ANL A, @Rr	50 51	1/1	'AND' RAM data, addressed by Rr, with A	(A)←(A) AND ((R0)) (A)←(A) AND ((R1))		
	ANL A, #data	53 data	2/2	'AND' immediate data with A	(A)←(A) AND data		
	ORL A, Rr	4*	1/1	'OR' Rr with A	(A)←(A) OR (Rr)	r = 0–7	
	ORL A, @Rr	40 41	1/1	'OR' RAM data, addressed by Rr, with A	(A)←(A) OR ((R0)) (A)←(A) OR ((R1))		
	ORL A, #data	43 data	2/2	'OR' immediate data with A	(A)←(A) OR data		
	XRL A, Rr	D*	1/1	'XOR' Rr with A	(A)←(A) XOR (Rr)	r = 0–7	
	XRL A, @Rr	D0 D1	1/1	'XOR' RAM, addressed by Rr, with A	(A)←(A) XOR ((R0)) (A)←(A) XOR ((R1))		
	XRL A, #data	D3 data	2/2	'XOR' immediate data with A	(A)←(A) XOR data		
	INC A	17	1/1	increment A by 1	(A)←(A) + 1		
	DEC A	07	1/1	decrement A by 1	(A)←(A) −1		
	CLR A	27	1/1	clear A to zero	(A)←0		
	CPL A	37	1/1	one's complement A	(A)←NOT(A)		
	RL A	E7	1/1	rotate A left	$(A_{n+1})\leftarrow(A_n)$ $(A_0)\leftarrow(A_7)$	n = 0–6	

ACCUMULATOR (cont.)	RLC A	F7	1/1	rotate A left through carry	$(A_{n+1})\leftarrow A_n$ $(A_0)\leftarrow(C)$, $(C)\leftarrow(A_7)$	n = 0–6	2
	RR A	77	1/1	rotate A right	$(A_n)\leftarrow(A_{n+1})$ $(A_7)\leftarrow(A_0)$	n = 0–6	
	RRC A	67	1/1	rotate A right through carry	$(A_n)\leftarrow(A_{n+1})$ $(A_7)\leftarrow(C)$, $(C)\leftarrow(A_0)$	n = 0–6	2
	DA A	57	1/1	decimal adjust A			2
	SWAP A	47	1/1	swap nibbles of A	$(A_{4-7})\leftrightarrow(A_{0-3})$		
DATA MOVES	MOV A, Rr	F*	1/1	move register contents to A	(A)←(Rr)	r = 0–7	
	MOV A, @Rr	F0 F1	1/1	move RAM data, addressed by Rr, to A	(A)←((R0)) (A)←((R1))		
	MOV A, #data	23 data	2/2	move immediate data to A	(A)←data		
	MOV Rr, A	A*	1/1	move accumulator contents to register	(Rr)←(A)	r = 0–7	
	MOV @Rr, A	A0 A1	1/1	move accumulator contents to RAM location addressed by Rr	((R0))←(A) ((R1))←(A)		
	MOV Rr, #data	B* data	2/2	move immediate data to Rr	(Rr)←data		
	MOV @Rr, #data	B0 data B1 data	2/2	move immediate data to RAM location addressed by Rr	((R0))←data ((R1))←data		
	XCH A, Rr	2*	1/1	exchange accumulator contents with Rr	(A)↔(Rr)	r = 0–7	
	XCH A, @Rr	20 21	1/1	exchange accumulator contents with RAM data addressed by Rr	(A)↔((R0)) (A)↔((R1))		
	XCHD A, @Rr	30 31	1/1	exchange lower nibbles of A and RAM data addressed by Rr	$(A_{0-3})\leftrightarrow((R0_{0-3}))$ $(A_{0-3})\leftrightarrow((R1_{0-3}))$		
	MOV A, PSW	C7	1/1	move PSW contents to accumulator	(A)←(PSW)		
	MOV PSW, A	D7	1/1	move accumulator bit 3 to PSW_3	$(PSW_3)\leftarrow(A_3)$		3
	MOVP A, @A	A3	1/2	move indirectly addressed data in current page to A	$(PC_{0-7})\leftarrow(A)$, $(A)\leftarrow((PC))$		
FLAGS	CLR C	97	1/1	clear carry bit	(C)←0		2
	CPL C	A7	1/1	complement carry bit	(C)←NOT(C)		2

	mnemonic	opcode (hex.)	bytes/ cycles	description	function	notes
REGISTER	INC Rr	1*	1/1	increment register by 1	(Rr)←(Rr) + 1 r = 0–7	
	INC @Rr	10 11	1/1	increment RAM data, addressed by Rr, by 1	((R0))←((R0)) + 1 ((R1))←((R1)) + 1	
	DEC Rr	C*	1/1	decrement register by 1	(Rr)←(Rr) – 1 r = 0–7	
	DEC @Rr	C0 C1	1/1	decrement RAM data, addressed by Rr, by 1	((R0))←((R0)) – 1 ((R1))←((R1)) – 1	
BRANCH	JMP addr	● 4 address	2/2	unconditional jump within a 2 K bank	(PC_{8-10})←$addr_{8-10}$ (PC_{0-7})←$addr_{0-7}$ (PC_{11-12})←MBFF 0–1	
	JMPP @A	B3	1/2	indirect jump within a page	(PC_{0-7})←((A))	
	DJNZ Rr, addr	E* address	2/2	decrement Rr by 1 and jump if not zero to addr	(Rr)←(Rr) – 1 r = 0–7 if (Rr) not zero (PC_{0-7})←addr	
	DJNZ @Rr, addr	E0	2/2	decrement RAM data, addressed by Rr by 1 and jump if not zero to addr	((R0))←((R0)) – 1 if ((R0)) not zero (PC_{0-7})←addr	
		E1			((R1))←((R1)) – 1 if ((R1)) not zero (PC_{0-7})←addr	
	JBb addr	▲ 2 address	2/2	jump to addr if Acc. bit b = 1	if b = 1 : (PC_{0-7})←addr b = 0–7	
	JC addr	F6 address	2/2	jump to addr if C = 1	if C = 1 : (PC_{0-7})←addr	
	JNC addr	E6 address	2/2	jump to addr if C = 0	if C = 0 : (PC_{0-7})←addr	
	JZ addr	C6 address	2/2	jump to addr if A = 0	if A = 0 : (PC_{0-7})←addr	
	JNZ addr	96 address	2/2	jump to addr if A is NOT zero	if A≠0 : (PC_{0-7})←addr	
	JT0 addr	36 address	2/2	jump to addr if T0 = 0	if T0 = 0: (PC_{0-7})←addr	
	JNT0 addr	26 address	2/2	jump to addr if T0 = 1	if T0 = 1: (PC_{0-7})←addr	
	JT1 addr	56 address	2/2	jump to addr if T1 = 1	if T1 = 1: (PC_{0-7})←addr	
	JNT1 addr	46 address	2/2	jump to addr if T1 = 0	if T1 = 0: (PC_{0-7})←addr	
	JTF addr	16 address	2/2	jump to addr if Timer Flag = 1	if TF = 1: (PC_{0-7})←addr	4
	JNTF addr	06 address	2/2	jump to addr if Timer Flag = 0	if TF = 0: (PC_{0-7})←addr	

TIMER/EVENT COUNTER	MOV A, T	42	1/1	move timer/event counter contents to accumulator	(A)←(T)	
	MOV T, A	62	1/1	move accumulator contents to timer/event counter	(T)←(A)	
	STRT CNT	45	1/1	start event counter		
	STRT T	55	1/1	start timer		
	STOP TCNT	65	1/1	stop timer/event counter		
	EN TCNTI	25	1/1	enable timer/event counter interrupt		
	DIS TCNTI	35	1/1	disable timer/event counter interrupt		
CONTROL	EN I	05	1/1	enable external interrupt		
	DIS I	15	1/1	disable external interrupt		
	SEL RB0	C5	1/1	select register bank 0	(RBS)←0	5
	SEL RB1	D5	1/1	select register bank 1	(RBS)←1	5
	SEL MB0	E5	1/1	select program memory bank 0	(MBFF0)←0, (MBFF1)←0	
	SEL MB1	F5	1/1	select program memory bank 1	(MBFF0)←1, (MBFF1)←0	
	SEL MB2	A5	1/1	select program memory bank 2	(MBFF0)←0, (MBFF1)←1	
	SEL MB3	B5	1/1	select program memory bank 3	(MBFF0)←1, (MBFF1)←1	
	STOP	22	1/1	enter STOP mode		
	IDLE	01	1/1	enter IDLE mode		
SUBROUTINE	CALL addr	▲4 address	2/2	jump to subroutine	((SP))←(PC), ($PSW_{4, 6, 7}$) 6 (SP)←(SP) + 1 (PC_{8-10})←$addr_{8-10}$ (PC_{0-7})←$addr_{0-7}$ (PC_{11-12})←MBFF 0–1	6
	RET	83	1/2	return from subroutine	(SP)←(SP) – 1 (PC)←((SP))	6
	RETR	93	1/2	return from interrupt and restore bits 4, 6, 7 of PSW	(SP)←(SP) – 1 ($PSW_{4, 6, 7}$) + (PC)←((SP))	6

	mnemonic	opcode (hex.)	bytes/ cycles	description	function	notes
PARALLEL INPUT/OUTPUT	IN A, Pp	08 09 0A	1/2	input port p data to accumulator	(A)←(P0) (A)←(P1) (A)←(P2)	 7
	OUTL Pp, A	38 39 3A	1/2	output accumulator data to port p	(P0)←(A) (P1)←(A) (P2)←(A)	
	ANL Pp, #data	98 99 9A	2/2	AND port p data with immediate data	(P0)←(P0) AND data (P1)←(P1) AND data (P2)←(P2) AND data	
	ORL Pp, #data	88 89 8A	2/2	OR port p data with immediate data	(P0)←(P0) OR data (P1)←(P1) OR data (P2)←(P2) OR data	
SERIAL INPUT/OUTPUT	MOV A, S_n	0C 0D	1/2	move serial I/O register contents to accumulator	(A)←(S0) (A)←(S1)	 8
	MOV S_n, A	3C 3D 3E	1/2	move accumulator contents to serial I/O register	(S0)←(A) (S1)←(A) (S2)←(A)	 9
	MOV S_n, #data	9C 9D 9E	2/2	move immediate data to serial I/O register	(S0)←data (S1)←data (S2)←data	
	EN SI	85	1/1	enable serial I/O interrupt		
	DIS SI	95	1/1	disable serial I/O interrupt		
	NOP	00	1/1	no operation		

Notes to Table 8

1. PSW CY, AC affected
2. PSW CY affected
3. PSW PS affected
4. Execution of JTF and JNTF instructions resets the Timer Flag (TF).

* : 8, 9, A, B, C, D, E, F
● : 0, 2, 4, 6, 8, A, C, E
▲ : 1, 3, 5, 7, 9, B, D, F

5. PSW RBS affected
6. PSW SP_0, SP_1, SP_2 affected
7. (A) = 1111 P23, P22, P21, P20.
8. (S1) has a different meaning for read and write operation, see serial I/O interface.
9. (S2) is a write only register. Reading S2 will give value FFH.

RATINGS

Limiting values in accordance with the Absolute Maximum System (IEC 134)

Supply voltage (pin 28)	V_{DD}		−0,8 to + 8	V
All input voltages	V_I		0,8 to V_{DD} + 0,8	V
D.C. current into any input or output	$\pm I_I, \pm I_O$	max.	10	mA
Total power dissipation (see note)	P_{tot}	max.	500	mW
Power dissipation per output				
except P23, SCLK	P_O	max.	50	mW
P23, SCLK	P_O	max.	180	mW
Storage temperature range	T_{stg}		−65 to + 150	°C
Operating ambient temperature range	T_{amb}		−25 to + 70	°C
Operating junction temperature	T_j	max.	125	°C

Note

Thermal resistance (junction to ambient)				
for SOT-117	$R_{th\ j\text{-}a}$	max.	120	K/W
for SOT-135A	$R_{th\ j\text{-}a}$	max.	60	K/W
for SOT-136A	$R_{th\ j\text{-}a}$	max.	150	K/W

DEVELOPMENT DATA

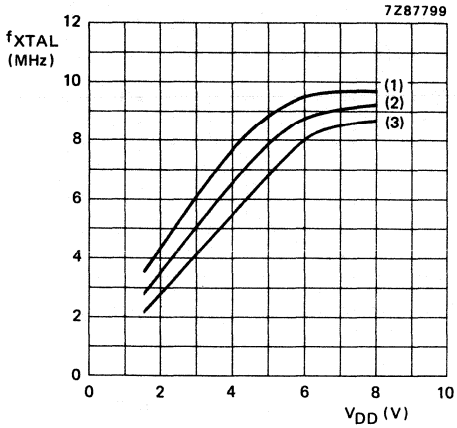
D.C. CHARACTERISTICS

V_{DD} = 2,75 to 6 V; V_{SS} = 0 V; T_{amb} = −25 to + 70 °C; all voltages with respect to V_{SS}; f = 3,58 MHz with R_S = 50 Ω; unless otherwise specified.

parameter	symbol	min.	typ.	max.	unit
Supply voltage					
operating (see Fig. 23)	V_{DD}	1,8	–	6	V
STOP mode for RAM retention	V_{DD}	1,0	–	6	V
Supply current					
operating					
at V_{DD} = 3 V (see Fig. 24)	I_{DD}	–	600	–	μA
IDLE mode					
at V_{DD} = 3 V (see Fig. 25)	I_{DD}	–	300	–	μA
STOP mode (see Fig. 26 and note 1)					
at V_{DD} = 1,8 V; T_{amb} = 25 °C	I_{DD}	–	1,2	2,5	μA
at V_{DD} = 1,8 V; T_{amb} = 55 °C	I_{DD}	–	–	5	μA
at V_{DD} = 1,8 V; T_{amb} = 70 °C	I_{DD}	–	–	10	μA
RESET I/O					
Switching level	V_{RESET}	–	1,3	–	V
Sink current					
at $V_{DD} > V_{RESET}$	I_{OL}	–	7	–	μA
Inputs					
Input voltage LOW	V_{IL}	0	–	$0{,}3V_{DD}$	V
Input voltage HIGH	V_{IH}	$0{,}7V_{DD}$	–	V_{DD}	V
Input leakage current					
at $V_{SS} < V_I < V_{DD}$	$\pm I_{IL}$	–	–	1	μA
Outputs					
Output voltage LOW					
at $V_I = V_{SS}$ or V_{DD}; $\lvert I_O \rvert < 1$ μA	V_{OL}	–	–	0,05	V
Output sink current LOW					
at V_{DD} = 3 V; V_O = 0,4 V					
except P23/SDA, SCLK (see Fig. 27)	I_{OL}	0,75	1,5	–	mA
P23/SDA, SCLK (see Fig. 28)	I_{OL}	1,5	–	–	mA
Pull-up output source current HIGH (see Fig. 29)					
at V_{DD} = 3 V; $V_O = 0{,}9V_{DD}$	$-I_{OH}$	25	–	–	μA
at V_{DD} = 3 V; $V_O = V_{SS}$	$-I_{OH}$	–	–	200	μA
Push-pull output source current HIGH					
at V_{DD} = 3 V; $V_O = V_{DD}$−0,4 V	$-I_{OH}$	0,75	1,5	–	mA

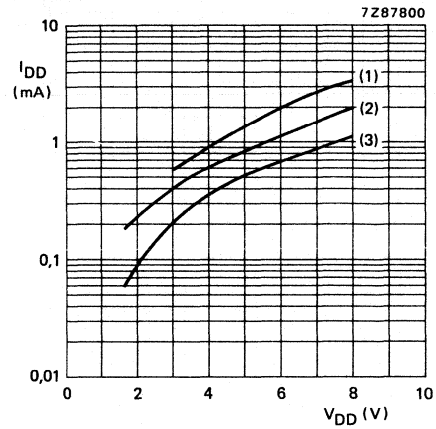
Note 1

Crystal connected between XTAL 1 and XTAL 2; SCL and SDA pulled to V_{DD} via 5,6 kΩ resistor; CE and T1 at V_{SS}.



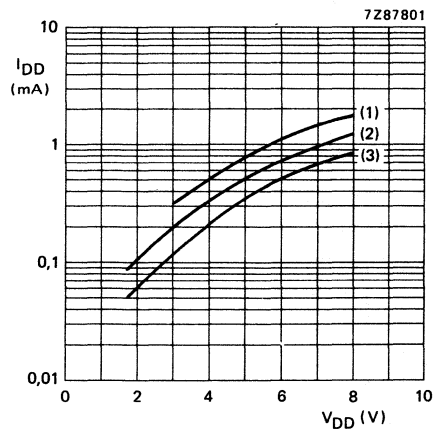
(1) T_{amb} = −25 °C
(2) T_{amb} = 25 °C
(3) T_{amb} = 70 °C

Fig. 23 Maximum clock frequency (f_{XTAL}) as a function of the supply voltage (V_{DD}).



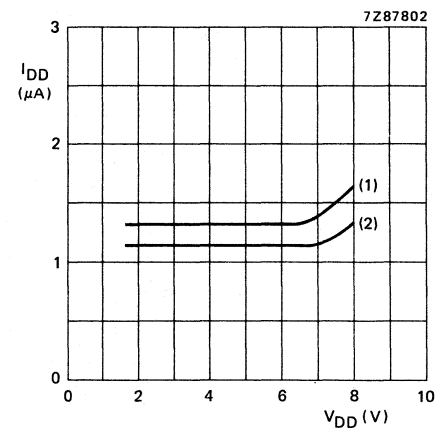
(1) clock frequency = 4 MHz
(2) clock frequency = 2 MHz
(3) clock frequency = 500 kHz

Fig. 24 Typical supply current (I_{DD}) in operating mode as a function of the supply voltage (V_{DD}); T_{amb} = 25 °C.



(1) clock frequency = 4 MHz
(2) clock frequency = 2 MHz
(3) clock frequency = 500 kHz

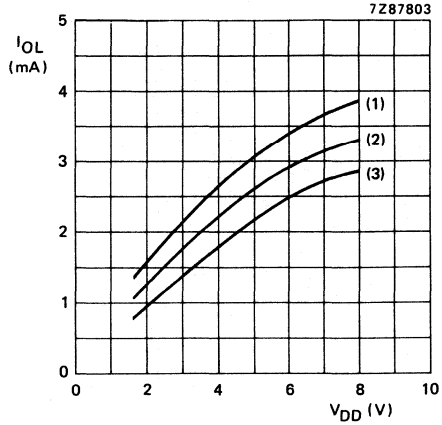
Fig. 25 Typical supply current (I_{DD}) in IDLE mode as a function of the supply voltage (V_{DD}); T_{amb} = 25 °C.



(1) T_{amb} = 70 °C
(2) T_{amb} = 25 °C

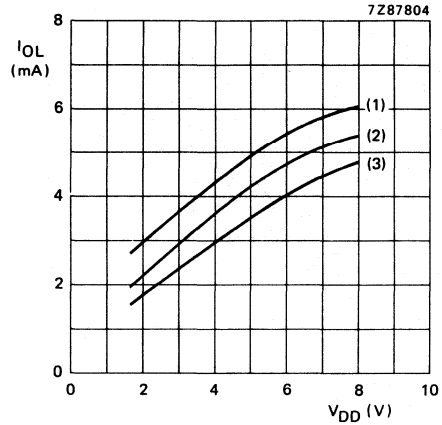
Fig. 26 Typical supply current (I_{DD}) in STOP mode as a function of the supply voltage (V_{DD}).

DEVELOPMENT DATA



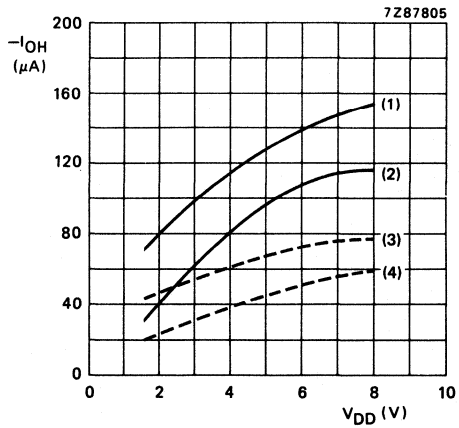
(1) $T_{amb} = -25\ ^{o}C$
(2) $T_{amb} = 25\ ^{o}C$
(3) $T_{amb} = 70\ ^{o}C$

Fig. 27 Output sink current LOW (I_{OL}), except outputs P23/SDA and SCLK, as a function of supply voltage (V_{DD}); $V_O = 0,4$ V.



(1) $T_{amb} = -25\ ^{o}C$
(2) $T_{amb} = +25\ ^{o}C$
(3) $T_{amb} = +70\ ^{o}C$

Fig. 28 Output current LOW (I_{OL}), outputs P23/SDA and SCLK, as a function of supply voltage (V_{DD}); $V_O = 0,4$ V.



(1) $T_{amb} = 25\ ^{o}C$; $V_O = V_{SS}$
(2) $T_{amb} = 25\ ^{o}C$; $V_O = 0,9V_{DD}$
(3) $T_{amb} = 70\ ^{o}C$; $V_O = V_{SS}$
(4) $T_{amb} = 70\ ^{o}C$; $V_O = 0,9V_{DD}$

Fig. 29 Output source current HIGH ($-I_{OH}$) as a function of supply voltage (V_{DD}).

A.C. CHARACTERISTICS

Rise and fall times between 10 and 90% levels; C_L = 50 pF

parameter	symbol	at 70 °C max. value			unit
	V_{DD}	1,8	3,0	6,0	V
Fall time	t_f	200	100	70	ns
Rise time	t_r	200	100	80	ns

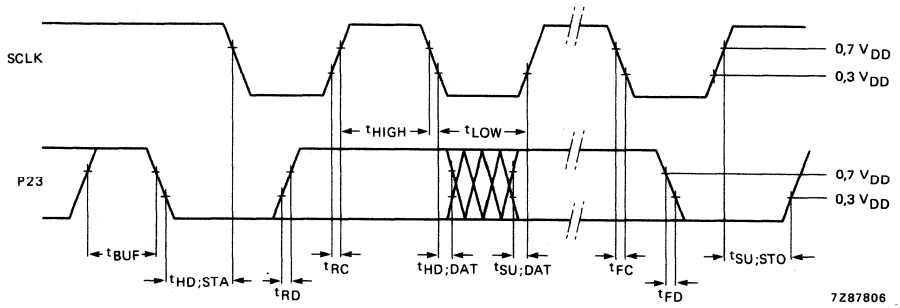


Fig. 30 PCD3343 timing requirements for the P23 and SCLK *input* signals.

Table 9 Input timing shown in figure 30

symbol	timing
t_{BUF}	$\geqslant 14t_{XTAL}$
$t_{HD;STA}$	$\geqslant 14t_{XTAL}$
t_{HIGH}	$\geqslant 17t_{XTAL}$
t_{LOW}	$\geqslant 17t_{XTAL}$
$t_{SY;STO}$	$\geqslant 14t_{XTAL}$
$t_{HD;DAT}$	> 0
$t_{SU;DAT}$	$\geqslant 250$ ns
t_{RD}	$\leqslant 1\ \mu s$
t_{RC}	$\leqslant 1\ \mu s$
t_{FD}	$\leqslant 1\ \mu s$
t_{FC}	$\leqslant 0{,}3\ \mu s$

Notes to Table 9

t_{XTAL} = one period of the XTAL input frequency (f_{XTAL})
= 280 ns for f_{XTAL} = 3,58 MHz.
These figures apply to all modes.

DEVELOPMENT DATA

A.C. CHARACTERISTICS (continued)

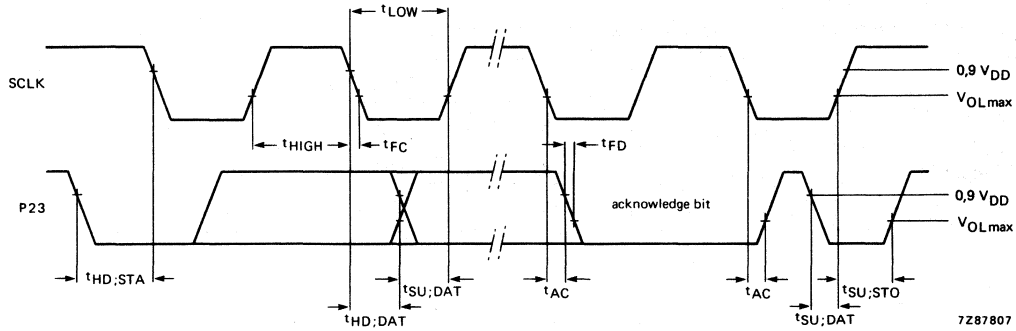


Fig. 31 PCD3343 timing requirements for the P23 and SCLK *output* signals.

Table 10 Output timing shown in figure 31

symbol	timing	
	normal mode (ASC in S2 = 0)	low-speed mode (ASC in S2 = 1)
$t_{HD;STA}$	½ (DF + 9) t_{XTAL}	¾ (DF + 9) t_{XTAL}
t_{HIGH}	½ (DF) t_{XTAL}	¾ (DF) t_{XTAL}
t_{LOW}	½ (DF) t_{XTAL}	¼ (DF) t_{XTAL}
$t_{SU;STO}$	½ (DF − 3) t_{XTAL}	¼ (DF − 3) t_{XTAL}
$t_{HD;DAT}$ (slave transmitter)		
any DF	$\geqslant 9t_{XTAL}$	$\geqslant 9t_{XTAL}$
	$\leqslant 12t_{XTAL}$	$\leqslant 12t_{XTAL}$
$t_{HD;DAT}$ (master transmitter)		
for DF $\leqslant$ 51	$\geqslant 9t_{XTAL}$	–
	$\leqslant 12t_{XTAL}$	–
for DF $\leqslant$ 99	–	$\geqslant 9t_{XTAL}$
	–	$\leqslant 12t_{XTAL}$
$t_{SU;DAT}$ (master transmitter)		
for DF > 51	$\geqslant 15t_{XTAL}$	–
	$\leqslant 24t_{XTAL}$	–
for DF > 99	–	$\geqslant 15t_{XTAL}$
for DF $\leqslant$ 51	$\geqslant 9t_{XTAL}$	$\leqslant 24t_{XTAL}$
for DF $\leqslant$ 99	–	$\geqslant 9t_{XTAL}$
t_{AC}	$\geqslant 9t_{XTAL}$	$\geqslant 9t_{XTAL}$
	$\leqslant 12t_{XTAL}$	$\leqslant 12t_{XTAL}$
t_{FD}, t_{FC}	$\leqslant$ 100 ns at C_b = 400 pF	$\leqslant$ 100 ns at C_b = 400 pF

Notes to Table 10

t_{XTAL} = one period of the XTAL input frequency (f_{XTAL})
= 280 ns for f_{XTAL} = 3,58 MHz.
DF = divisor (see Table 2 Serial I/O section).
C_b = the maximum bus capacitance for each line.

DEVELOPMENT DATA

APPLICATION INFORMATION

A block diagram of an electronic featurephone built around the PCD3343 is shown in figure 32. It comprises the following dedicated telephony IC's:

- TEA1060/1061 — transmission circuit for telephony
- PCD3312 — DTMF generator with Serial I/O
- PCE2111 or PCF8577 — 2 LCD drivers in LCD module MB7020160
- PCD8571 — 1 K RAM's with Serial I/O; the number of RAM's depends on the required amount of stored telephone numbers
- PCD3360/3361 — programmable multi-tone ringer

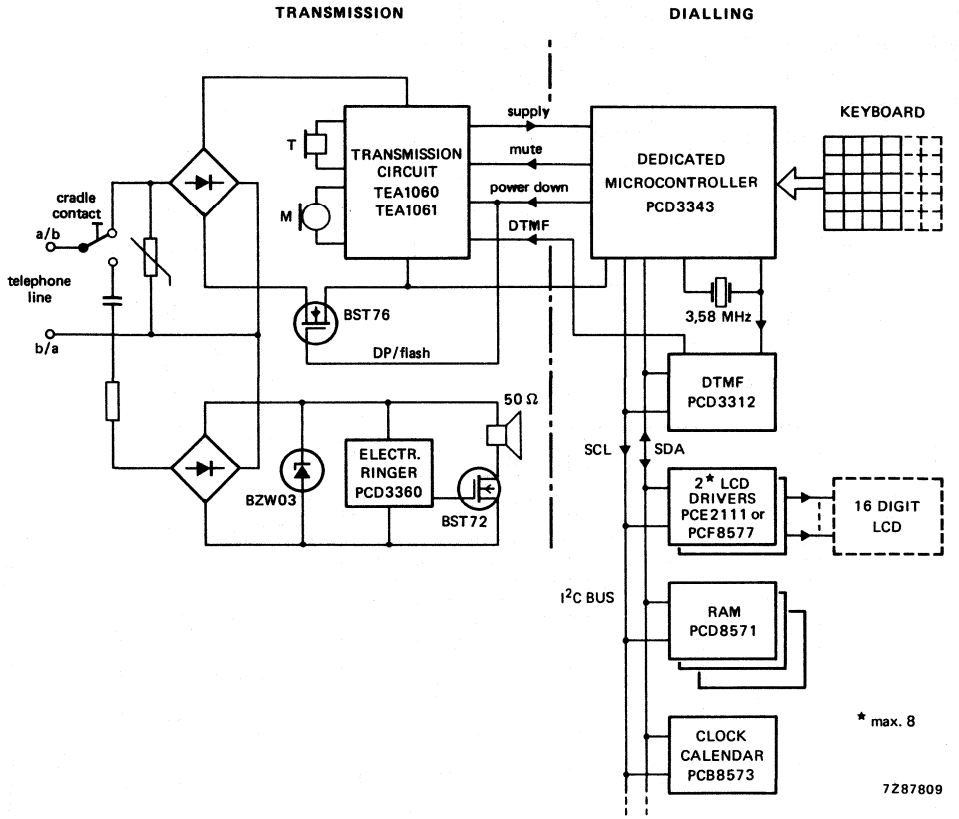


Fig. 32 Block diagram of electronic featurephone with common line interface.

A detailed application diagram of the PCD3343 with PCD3312 (DTMF), two PCD8571 (RAM) and two PCE2111 (LCD display drivers) is shown in figure 33.

Row 5 of the keyboard contains the following special keys:

- P — program and autodial
- FL — flash or register recall
- R — redial or extended redial
- AP — access pause

Row 6 contains the different diode options.

Columns 5 and 6 contain the button keys M0 to M9; single name keys for repertory telephone numbers.

APPLICATION INFORMATION (continued)

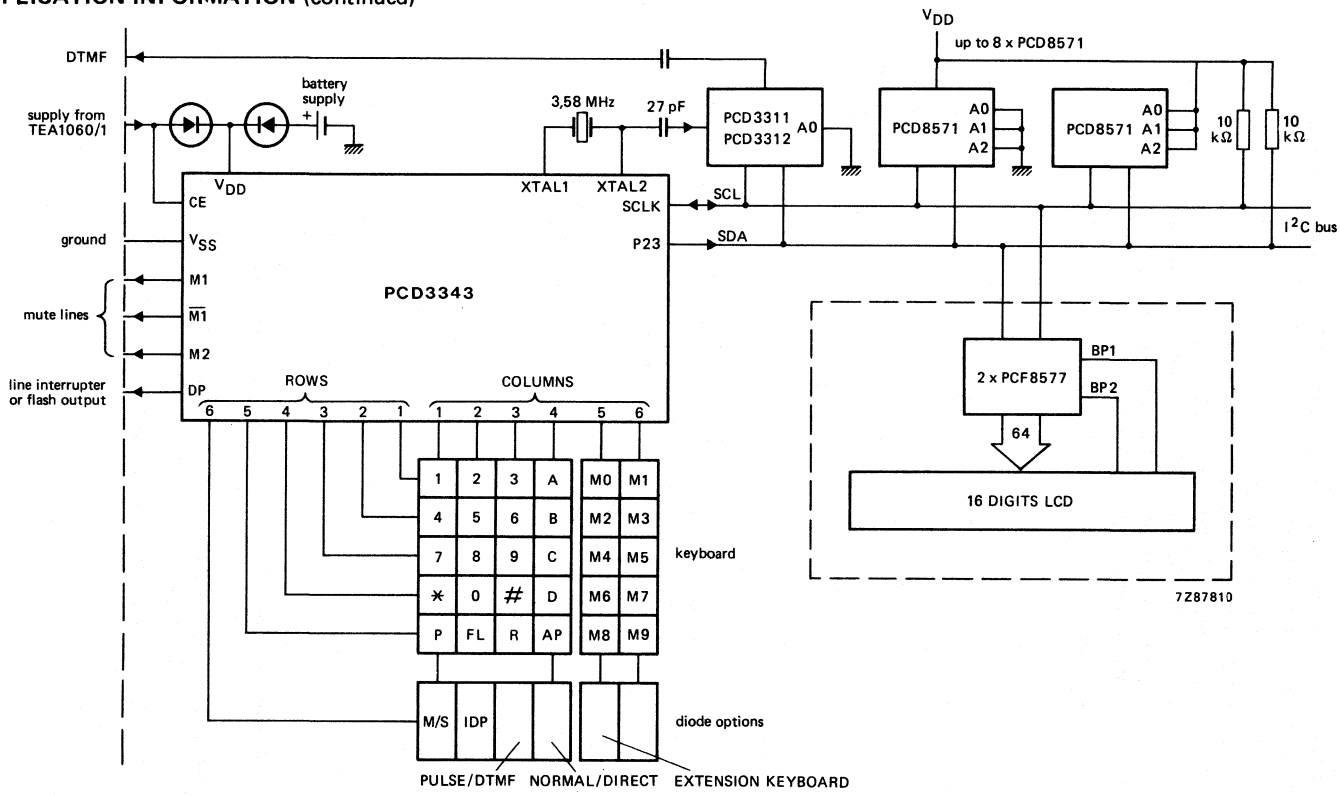


Fig. 33 Application diagram of PCD3343 for electronic featurephone with associated keyboard.

Additional information is available on request for the following:

- Serial I/O
- I^2C bus specification
- Interrupt logic
- Instruction set descriptions
- Software routines for an intelligent telephone set

CMOS MICROCONTROLLER WITH ON-CHIP DTMF GENERATOR

GENERAL DESCRIPTION

The PCD3349 is a single-chip 8-bit microcontroller fabricated in CMOS and is a member of the PCD3343 family. It has an on-chip dual tone multi-frequency (DTMF) generator and other features for application in telephone sets.

Features

- 8-bit CPU, ROM, RAM, I/O in a single 28-lead DIL or SO package
- 4096 ROM bytes
- 224 RAM bytes
- On-chip DTMF tone generator
- On-chip voltage reference for supply and temperature-independent tone output
- On-chip filtering for low output distortion (CEPT CS203 compatible)
- 20 quasi-bidirectional I/O port lines
- Two test inputs: one of which is also the external interrupt input (CE/$\overline{T0}$)
- Single-level vectored interrupts: external, timer/event counter
- 8-bit programmable timer/event counter
- Over 80 instructions (based on MAB8048, PCD3343 and PCD84C00)
- All instructions 1 or 2 cycles
- Clock frequency 3.58 MHz
- Single supply voltage from 2.5 V to 6 V
- Low standby voltage and current
- STOP and IDLE mode
- On-chip oscillator
- Configuration of all I/O port lines individually selected by mask: pull-up, open drain or push-pull
- Power-on-reset circuit and low supply voltage detection
- Reset state of all ports individually selected by mask
- Operating temperature range: −25 to + 70 °C

PACKAGE OUTLINES

PCD3349P: 28-lead DIL; plastic (SOT117).
PCD3349T: 28-lead mini-pack; plastic (SO28; SOT136A).

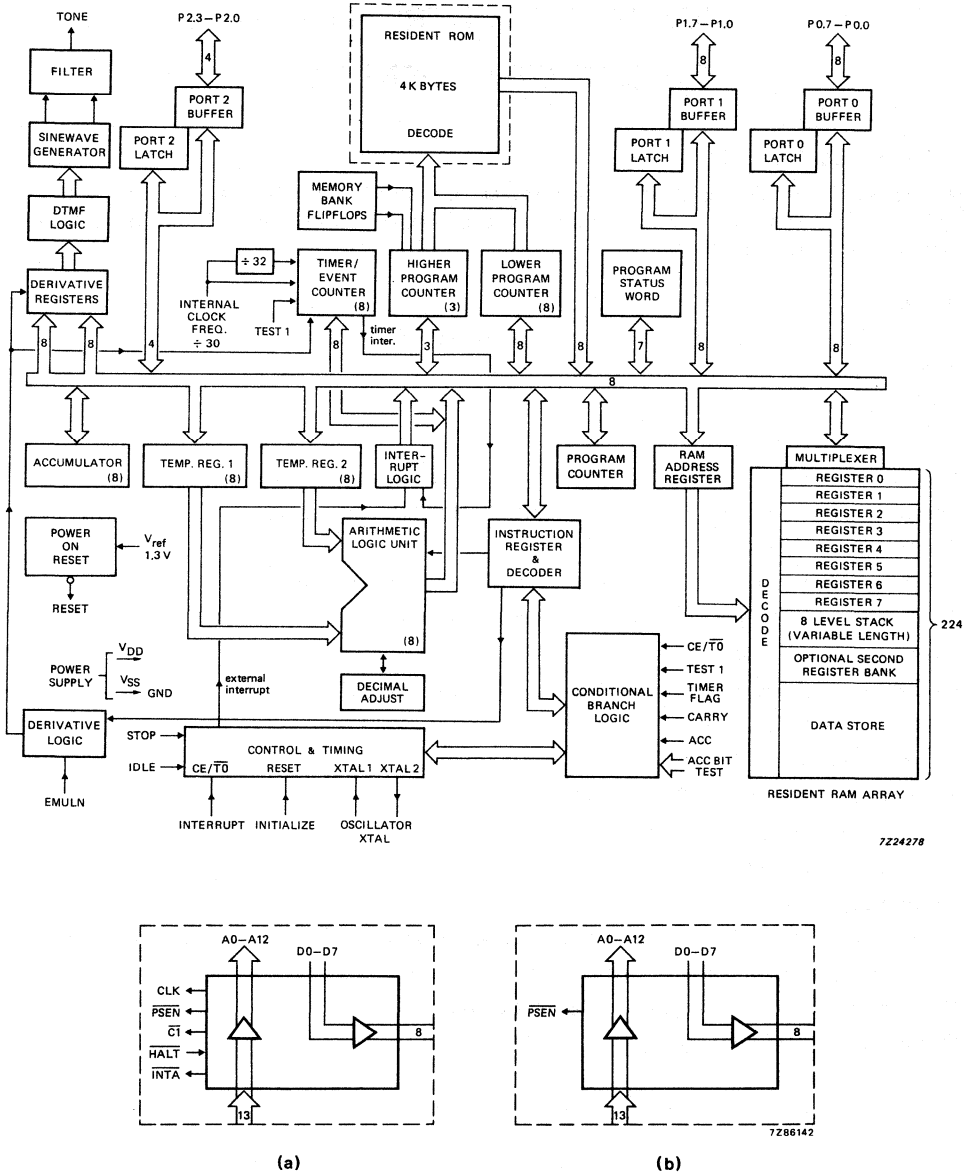


Fig. 1 PCD3349 block diagram: the function in the dotted outline is replaced as shown in (a) for the PCF84C00 bond-out version or as shown in (b) for the PCF84C00 'piggy-back' version.

DEVELOPMENT DATA

PINNING (for normal operation)

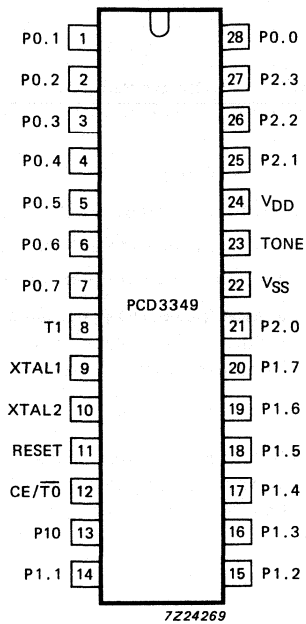


Fig. 2 Pinning diagram.

PIN DESIGNATION

28, 1-7	P0.0-P0.7	Port 0: 8-bit quasi-bidirectional I/O port.
8	T1	Test 1: test input, directly tested by conditional branch instructions JT1 and JNT1. T1 also functions as an input to the 8-bit timer/event counter using the STRT CNT instruction.
9	XTAL1	Crystal input: connection to the timing component (crystal) which determines the frequency of the internal oscillator; is also the input for an external clock source.
10	XTAL2	Connection to other side of timing component.
11	RESET	Reset input (active HIGH): used to initialize the processor or output of the power-on-reset circuit.
12	CE/$\overline{T0}$	Interrupt/Test 0: external interrupt input (sensitive to positive-going edge)/test input pin. When used as a test input is directly tested by conditional branch instructions JT0 and JNT0.
13-20	P1.0-P1.7	Port 1: 8-bit quasi-bidirectional I/O port.
21, 25-27	P2.0-P2.3	Port 2: 4-bit quasi-bidirectional I/O port.
22	V_{SS}	Ground: circuit earth potentional.
23	TONE	Tone output: single or dual tone frequency output with on-chip filtering for low output distortion (CEPT CS203 compatible). This generator is controlled via the internal processor bus.
24	V_{DD}	Power supply: 2.5 to 6 V.

Emulation version PCD3349/800

Emulation of the PCD3349 is done using the special emulation version (the PCD3349/800) together with the PCF84C00 bond-out chip. Of the PCD3349/800 only the DTMF generator is used, the complete control and computing function being performed by the PCF84C00.

FUNCTIONAL DESCRIPTION

Program memory PCD3349

The program memory comprises 4096 bytes (8-bit words) in a read-only memory (ROM). Each location is directly addressable by the program counter. The memory is mask-programmed at the factory. Figure 3 shows the program memory map.

Three program memory locations are of special importance:

- Location 0; contains the first instruction to be executed after the processor is initialized (RESET),
- Location 3; contains the first byte of an external interrupt service subroutine,
- Location 7; contains the first byte of a timer/event counter interrupt service subroutine.

The program memory is divided into location 'pages', each of 256 bytes. This division applies only for conditional branches. A CALL instruction can transfer control to a subroutine on any 'page'; RET and RETR instructions can transfer control from a subroutine back to the main program.

Data memory PCD3349

Data memory consists of 224 bytes (8-bit words) of random-access data memory (RAM). All locations are indirectly addressable using RAM pointer registers; up to 16 designated locations are directly addressable. Memory also includes an 8-level program counter stack addressed by a 3-bit stack pointer. Figure 4 shows the data memory map.

Working registers

Locations 0 to 7 are designated as working registers, directly addressable by the direct register instructions. Ease of addressing, and a minimum requirement of instruction bytes to manipulate their contents, makes these locations suitable for storing frequently-addressed intermediate results. This bank of registers can be selected by the SEL RB0 instruction.

Executing the select register bank instruction SEL RB1, designates locations 24 to 31 as working registers, instead of locations 0 to 7, and these are then directly addressable. This second bank of working registers may be used as an extension of the first or reserved for use during interrupt service subroutines saving the first bank for use in the main program. If the second bank is not used, locations 24 to 31 may serve as general purpose RAM.

The first locations of each bank contain the RAM pointer registers R0, R1, R0' and R1', which indirectly address all RAM locations.

All RAM locations make efficient program loop counters when used with the decrement register and test instruction DJNZ.

Program counter stack

Locations 8 to 23 may be designated as an 8-level program counter stack (2 locations per level), or as general purpose RAM. The program counter stack (Fig. 5) enables the processor to keep track of the return addresses and status generated by interrupts or CALL instructions by storing the contents of the program counter prior to servicing the subroutine. A 3-bit stack pointer determines which of the eight register pairs of the program counter stack will be loaded with next generated return address.

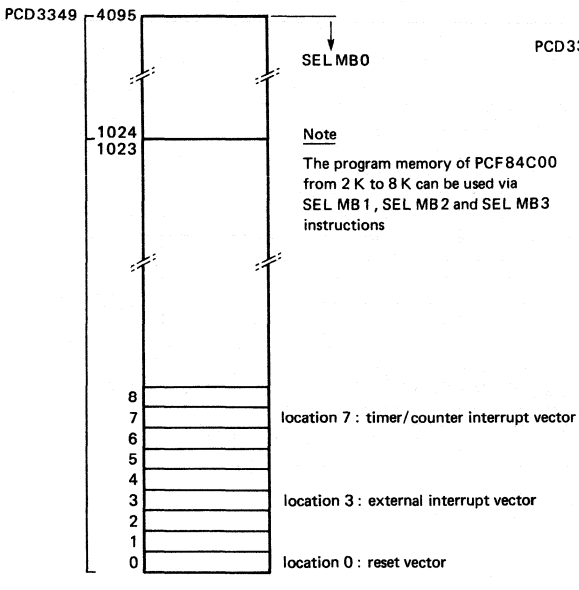


Fig. 3 Program memory map.

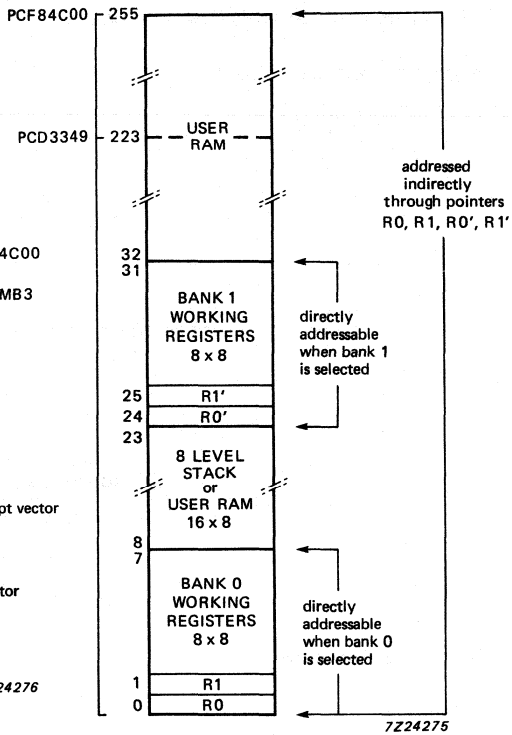


Fig. 4 Data memory map.

The stack pointer, when initialized to 000 by RESET, points to RAM locations 8 and 9. On the first subroutine CALL or interrupt, the contents of the program counter and bits 4, 6 and 7 of the program status word (PSW) are transferred to locations 8 and 9. The stack pointer increments by one and points to locations 10 and 11 ready for another CALL. Because an address may be up to 13 bits long, two bytes must be used to store each address.

At the end of a subroutine, which is signalled by a return instruction (RET or RETR), the stack pointer decrements by one and the contents of the register pair on top of the stack are transferred to the program counter. The saved PSW bits are transferred to the PSW only by the RETR instruction.

If not all 8 levels of subroutine and interrupt nesting are used, the unused portion of the stack may be used as any other indirectly addressable RAM locations. Possible locations from 32 to 223 may be used for storage of program variables or data.

Nesting of subroutines within subroutines can continue up to 8 times without overflowing the stack. If overflow does occur the deepest address stored (locations 8 and 9) will be overwritten and lost since the stack pointer overflows from 111 to 000. It also underflows from 000 to 111.

The value of the saved contents of the program counter is different for an interrupt CALL compared to a normal CALL to subroutine. With an interrupt CALL, the program counter return address is saved; with a subroutine CALL, the saved program counter value is one less than the program counter return address.

FUNCTIONAL DESCRIPTION (continued)

Program counter stack (continued)

stack pointer									
1 1 1	----------								R23/22
1 1 0	----------								R21/20
1 0 1	----------								R19/18
1 0 0	----------								R17/16
0 1 1	----------								R15/14
0 1 0	----------								R13/12
0 0 1	----------								R11/10
0 0 0	PSW7	PSW6	PC12	PSW4	PC11	PC10	PC9	PC8	R9
	PC7	PC6	PC5	PC4	PC3	PC2	PC1	PC0	R8

Fig. 5 Program counter stack.

IDLE and STOP modes

IDLE mode

When the microcontroller enters the IDLE mode via the IDLE instruction (H'01') the oscillator and timer/counter are kept running. The microcontroller exits from the IDLE mode by one of two interrupts if they are enabled or by activating a RESET. If the interrupt is not enabled the processor will remain in the IDLE mode. An active signal on the RESET pin restarts the microcontroller and a normal RESET sequence is executed (see Fig. 6).

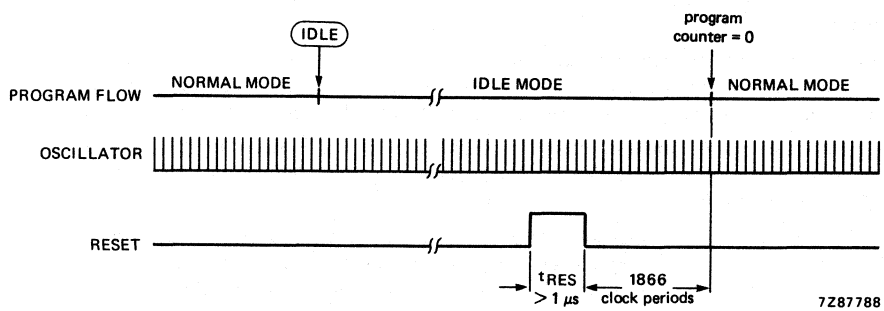


Fig. 6 Exit from IDLE mode via a RESET.

An active signal coming from an enabled interrupt causes the execution of the normal interrupt routine since normal interrupt scanning is still being carried out. A LOW-to-HIGH transition on the external interrupt pin (CE/$\overline{T0}$) reactivates the microcontroller. A HIGH level applied to CE/$\overline{T0}$ will reactivate the microcontroller only in the STOP mode. Thus, if CE/$\overline{T0}$ was HIGH before the microcontroller entered the IDLE mode, it must go LOW before the microcontroller can be reactivated (see Fig. 7).

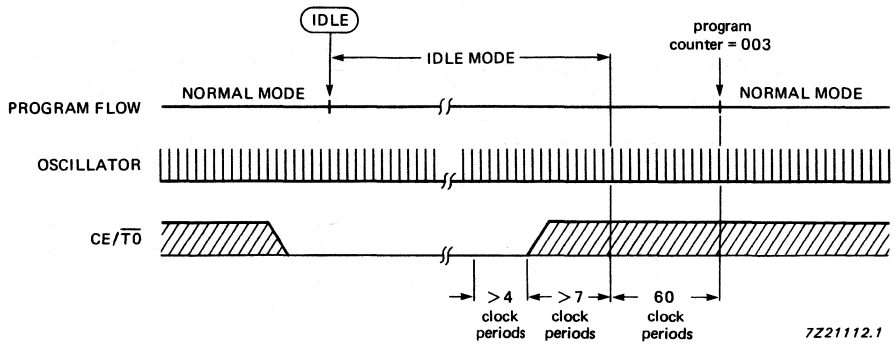


Fig. 7 Exit from IDLE mode via an interrupt.

Wake-up from the IDLE mode is ensured when CE/$\overline{T0}$ is LOW for 4 CP (clock periods) followed by a HIGH for 7 CP. After the initial forced CALL H'003' operation (60 CP) the program continues with the external interrupt service routine.

STOP mode

The microcontroller enters the STOP mode by the STOP instruction (H'22'). The oscillator is switched off. The internal status of the CPU, RAM contents and the state of I/O ports are not affected. The microcontroller can be brought-out of the STOP mode by an active signal at the external interrupt input or by an external RESET signal. When one of these two signals is applied an internal delay of 1866 CP is provided to ensure that all internal clocks are operating correctly before restart (see Fig. 8).

If the microcontroller exits from the STOP mode by activating RESET, a normal RESET sequence is executed.

If the microcontroller exits the STOP mode by pulling the external interrupt input pin HIGH, an interrupt sequence is executed only if the external interrupt is enabled. In this event the microcontroller resumes the normal program sequence after returning from the interrupt routine, as in the normal mode. If the interrupt is not enabled, it continues the normal program sequence, executing the instruction following the STOP instruction.

The microcontroller is restarted by a HIGH level applied at the CE/$\overline{T0}$ pin, and not by a LOW-to-HIGH transition as in a normal interrupt mechanism.

When the CE/$\overline{T0}$ level is active during the STOP instruction then no STOP is executed.

A HIGH level on the external interrupt input of at least 1 μs will cause the microcontroller to exit the STOP mode.

DEVELOPMENT DATA

FUNCTIONAL DESCRIPTION (continued)

STOP mode (continued)

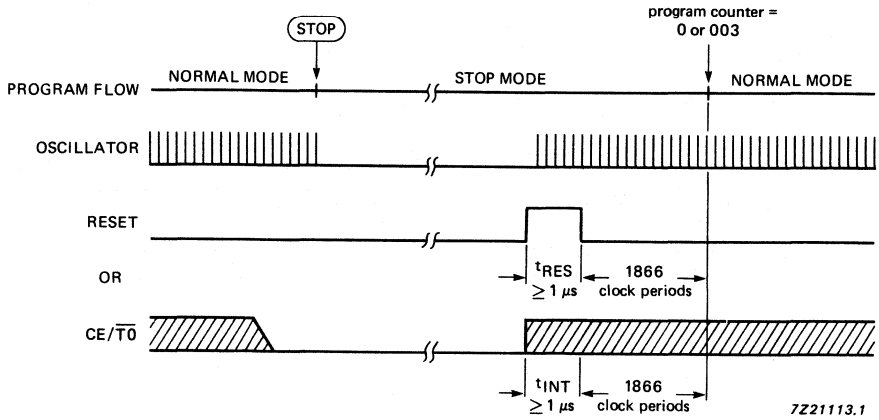


Fig. 8 Entering and exiting the STOP mode.

Tone output (DTMF mode)

Control of the sinewave generator

The on-chip sinewave oscillator is controlled by the 'derivative' registers Dx (x = H'0' to 'FF'). The instruction that controls the derivative registers is shown in Table 1.

Table 1 Derivative register control

mnemonic	opcode	description	function
MOV Dx,A	8D Dx	move accumulator contents to derivative register	(Dx) ← (A)

The instruction is 2 cycles/2 bytes. The second byte selects the derivative register to be addressed (H'0' to 'FF'). Register H'01' is for control of HIGH group frequencies, and register H'02' for control of LOW group frequencies. Thus data transport from accumulator to derivative register D01 is done by the 2-byte opcode 8D,01.

Generation of frequencies

The single and dual tones at the tone output are filtered by an on-chip switched-capacitor filter followed by an on-chip active RC low-pass filter. These ensure that the total harmonic distortion of the DTMF tones fulfil the CEPT CS 203 recommendations. An on-chip reference voltage provides output tone levels that are independent of the supply voltage.

The output frequency can be calculated as follows:

$$f_{out} = \frac{f_{XTAL}}{23\,(x+2)}\ \text{Hz}$$

x = 60 to 255 and is the decimal value of the appropriate ROM-code (see Table 2)

Table 2 ROM-codes for DTMF applications

telephone keyboard symbol	contents of low register (hex)	contents of high register (hex)
0	A3	72
1	DD	7F
2	DD	72
3	DD	67
4	C8	7F
5	C8	72
6	C8	67
7	B5	7F
8	B5	72
9	B5	67
A	DD	5D
B	C8	5D
C	B5	5D
D	A3	5D
*	A3	7F
#	A3	67

DTMF generation is stopped by loading H'00' into both derivative registers.

I/O facilities

The PCD3349 family has 22 I/O lines arranged as:

- Port 0 parallel port of 8 lines (P0.0 to P0.7)
- Port 1 parallel port of 8 lines (P1.0 to P1.7)
- Port 2 parallel port of 4 lines (P2.0 to P2.3)
- CE/$\overline{T0}$ external interrupt and test input. When used as a test input it can be directly tested by conditional branch instructions JT0 and JNT0
- T1 test input which can alter program sequences when tested by conditional jump instructions JT1 and JNT1. T1 also functions as an input to the 8-bit timer/event counter.

DEVELOPMENT DATA

FUNCTIONAL DESCRIPTION (continued)

Parallel ports

All parallel ports can be used as outputs or inputs, their structure is quasi-bidirectional. Output data written to a port is latched and remains unchanged until rewritten. Input data is not latched and so must be present until read by an input instruction.

Input lines are fully CMOS compatible, output lines can drive one LS-TTL or CMOS load.

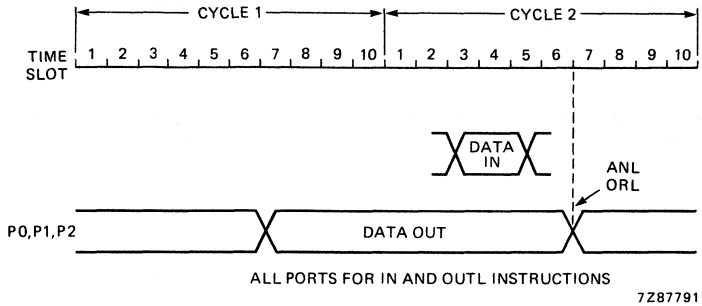


Fig. 9 Timing diagram of all ports on IN and OUTL instructions; for ANL and ORL instructions, the ports change on the time slot 7 of cycle 2.

Fig. 10 shows the quasi-bidirectional I/O interface with push-pull output and switched pull-up current source. Each line is pulled up to V_{DD} via a constant current source (TR4), which is enabled via TR3 whenever one of the two output latches contains a logic 1. This current source provides sufficient current for a TTL HIGH level, yet can be pulled LOW by an external CMOS device, thus allowing the same pin to be used for both input and output.

When a logic 1 is written to the line for the first time (MQ = 1, SQ = 0), TR2 is switched on for the duration of the internal write pulse (one oscillator period) to provide a fast transition from logic 0 to logic 1. Subsequent writing of a logic 1 to the port lines will not switch TR2 on. This prevents unnecessary current through external components connected to the port lines of the same port which might be in the input mode and also connected to ground.

When a logic 0 is written to the line, TR3 switches off the current source. Current sinking capability is provided by TR1, which is now switched on. When used as an input, a logic 1 must first be written to the line, otherwise TR1 will remain low impedance.

In telephone applications this switched pull-up source may not be sufficient. Therefore the PCD3349 offers the possibility to select individually the 20 parallel port pins by the following mask options:

Option 1 –STANDARD PORT; quasi-bidirectional I/O with switched pull-up current source of 100 μA (typ.) and P-channel booster transistor TR2 (1.5 mA). TR2 is active only during 1 clock cycle (0.28 μs at 3.58 MHz).

Option 2 –OPEN DRAIN; quasi-bidirectional I/O with only an N-channel open drain output. Application as an output requires connection of an external pull-up resistor (Fig. 11).

Option 3 –PUSH-PULL OUTPUT; drive capability of the output will be 1.5 mA (typ.) at V_{DD} = 3 V in both polarities. To avoid a large current flowing through the output transistors during the input mode, these push-pull pins must be used only as outputs (Fig. 12).

Also, individual mask selection of the RESET state of these port pins can be achieved by appending the following options S and R to options 1, 2 or 3.

Option S-SET; after RESET this pin will be initialized to HIGH.

Option R-RESET; after RESET this pin will be initialized to LOW.

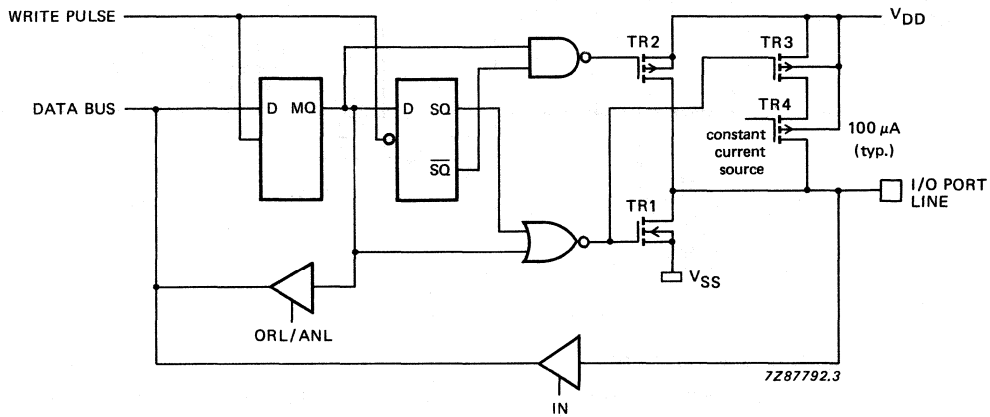


Fig. 10 Standard output with switched pull-up current source.

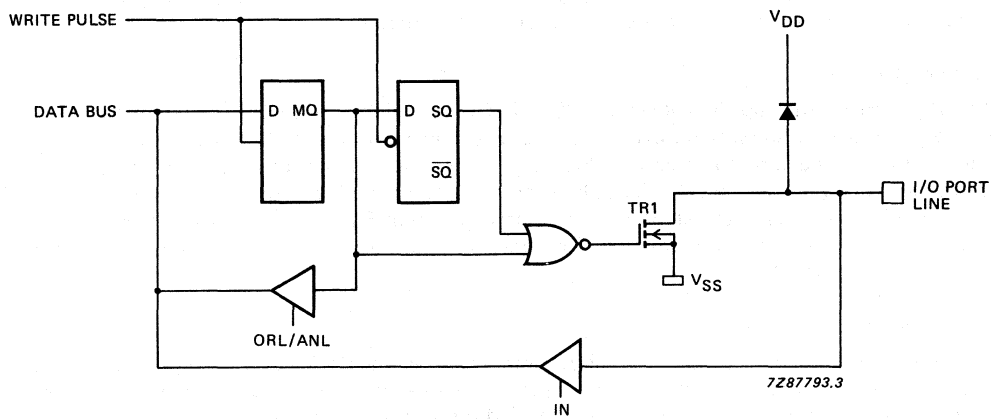


Fig. 11 Open drain output.

DEVELOPMENT DATA

FUNCTIONAL DESCRIPTION (continued)

Parallel ports (continued)

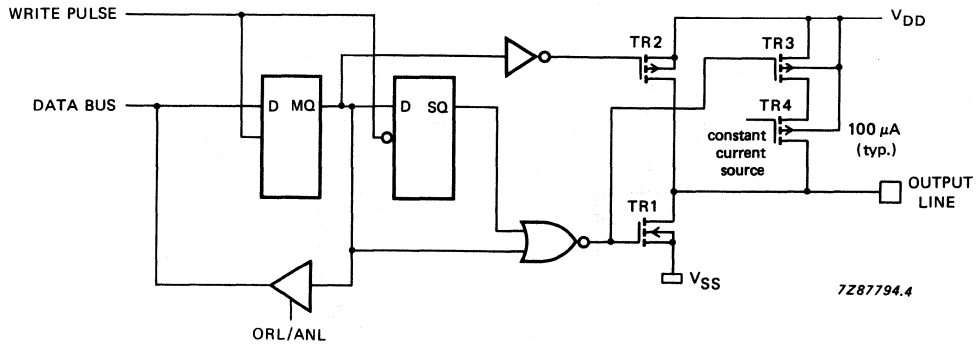


Fig. 12 Push-pull output.

Interrupts (see Fig. 13(a) and Fig. 13(b))

When an interrupt routine is entered, the contents of the program counter and bits 4, 6 and 7 of the PSW are saved in the program counter stack. The contents of the accumulator can only be saved by user software. Interrupt acknowledgement can be carried out by software via I/O ports. All interrupt routines must reside in memory bank 0; the SEL MB1, SEL MB2 and SEL MB3 instructions may not be used in an interrupt routine. An interrupt routine can only be terminated by the RETR (return and restore) instruction. During an interrupt routine, subroutine calls must be terminated by the RET instruction. Using the RETR instruction to terminate a subroutine called in an interrupt routine would terminate the interrupt routine prematurely and result in a wrong return address.

1. **External interrupt**

When the external interrupt is enabled, a LOW-to-HIGH transition on the CE/$\overline{TO}$ input initiates an external interrupt routine which forces a call to program memory location 3. The program counter points to the external interrupt vector address (003 H) between 2.6 and 3.6 machine cycles after the transition occurs. Interrupt latency depends on the instruction that is being executed when the transition occurs. External interrupts are latched in the External Interrupt Flag (EIF) even when they are not enabled. Execution of a DIS I instruction clears previously latched interrupts, the digital filter latch and the external interrupt flag.

2. **Timer/counter interrupt**

When the timer interrupt is enabled, a timer/counter overflow sets the Timer Interrupt Flag (TIF) and forces a CALL to location 7. The timer interrupts are only latched when they are enabled. The timer flag is set every time the timer/counter overflows and is not automatically reset when the timer/counter interrupt routine is called. It can only be cleared by the JTF and JNTF instructions or by a hardware RESET.

3. Simultaneous interrupts

If simultaneous interrupts occur their priority is as follows:

external (highest);
timer/counter (lowest).

An interrupt routine can only be interrupted by a hardware RESET and cannot be interrupted by other interrupts (which will be latched if enabled). When the interrupt routine is terminated by the RETR instruction, at least one instruction of the main program will be executed before another interrupt routine is entered.

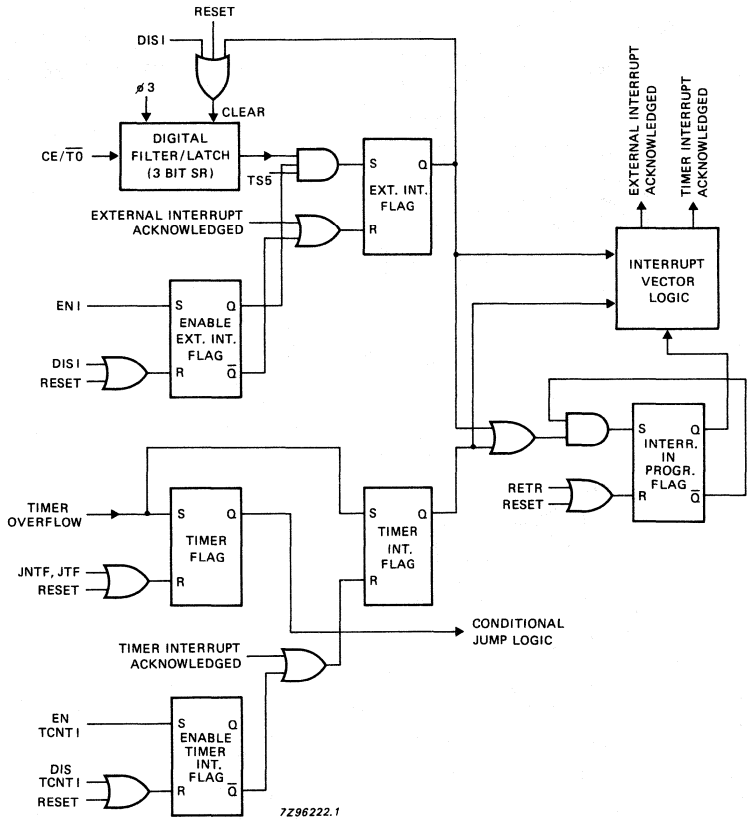


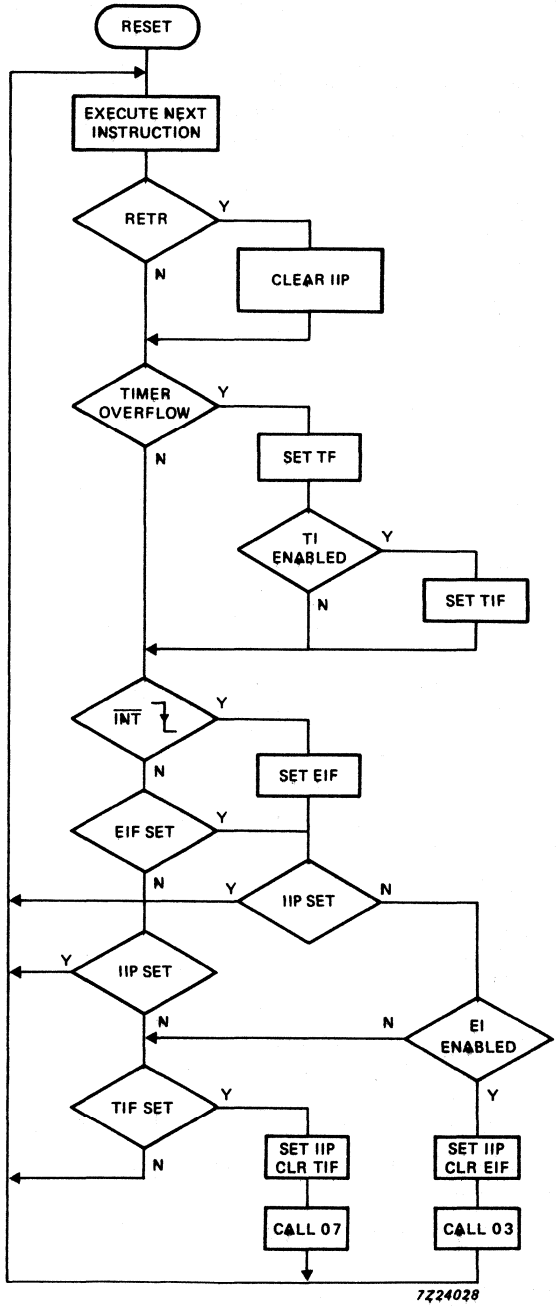
Fig. 13(a) Interrupt logic.

Notes to figure 13(a)

1. CE/$\overline{T0}$ positive edge is always latched in the digital filter/latch.
2. Correct interrupt timing is ensured when CE/$\overline{T0}$ is LOW for $>$ 4 CP followed by a HIGH for $>$ 7 CP.
3. When the interrupt in progress flag is set, further interrupts are latched but ignored, until RETR is executed.
4. A DIS I instruction always clears a pending external interrupt.
5. For all flip-flops, RESET overrules SET.

FUNCTIONAL DESCRIPTION (continued)

Interrupts (continued)



EI	External Interrupt
TI	Timer/counter Interrupt
EIF	External Interrupt Flag
TF	Timer Flag
IIP	Interrupt In Progress flag

Fig. 13(b) Interrupt flowchart.

Oscillator (see Fig. 14)

The 3.58 MHz oscillator can be inhibited by the STOP instruction under software control. It is also inhibited when a low-voltage condition is present to prevent discharge of a weak back-up battery.

Provided the supply voltage is within the operating range, the oscillator will be restarted after a STOP instruction by a HIGH level at either the CE/$\overline{TO}$ or RESET pin.

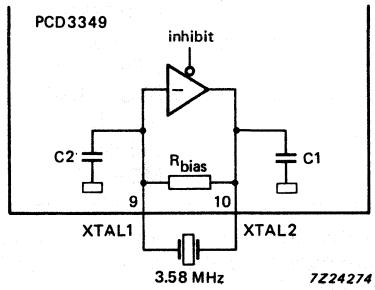


Fig. 14 Oscillator with integrated elements.

The oscillator has an output drive capability from pin 10 (XTAL2). An external clock can be applied to pin 9 (XTAL1). A machine cycle comprises 10 time slots, each time slot being 3 oscillator periods.

In telephony applications the 3.58 MHz crystal provides an 8.4 μs machine cycle. The range of the clock frequency is from 100 kHz up to a maximum which is a function of the supply voltage.

Timer/event counter (see Fig. 15)

An internal 8-bit up-counter is provided. This can count external events, modulo-32 machine cycles, or machine cycles directly. Table 3 gives the instructions that control the counter and the prescaler, and the functions performed.

When used as a timer, the input to the counter is either the overflow or input of a 5-bit prescaler. When used as an event counter, LOW-to-HIGH transitions on pin 8 (T1) are counted. The maximum rate at which the counter may be incremented is once every machine cycle (182.6 kHz for an 8.4 μs machine cycle). When the counter overflows, the timer flag is set. The flag can be tested and reset using the JTF (jump if timer flag = 1) or JNTF instruction. Overflow also generates an interrupt to the processor via setting of the Timer Interrupt Flag when the timer/event counter interrupt is enabled.

FUNCTIONAL DESCRIPTION (continued)

Timer/event counter (continued)

Table 3 Timer/event counter control

function	timer mode modulo-1, modulo-32*	counter mode
CLEAR	MOV T,A (A) = 0 or RESET	MOV T,A (A) = 0 or RESET
PRESET	MOV T,A	MOV T,A
START	STRT T	STRT CNT
STOP	STOP TCNT or RESET	STOP TCNT or RESET
TEST	JTF/JNTF	JTF/JNTF
READ**	MOV A,T	MOV A,T

* With prescaler select, PS = 0, the timer counts modulo-32 machine cycles, with PS = 1 it counts modulo-1 cycles (prescaler not used); prescaler cleared with STRT T, prescaler not readable.

** READ does not disturb the counting process.

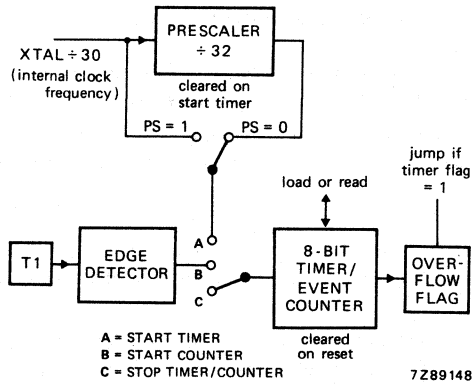


Fig. 15 Timer/event counter.

Program status word (see Fig. 16)

The program status word (PSW) is an 8-bit word (1 byte) in the CPU which stores information about the current status of the microcontroller.
The PSW bits are:

- Bits 0 to 2 — stack pointer bits (SP_0, SP_1, SP_2)
- Bit 3 — prescaler select (PS); 0 = modulo-32; 1 = modulo-1 (no prescaling)
- Bit 4 — working register bank select (RBS); 0 = register bank 0; 1 = register bank 1
- Bit 5 — not used (1)
- Bit 6 — auxiliary carry (AC); half-carry bit generated by an ADD instruction and used by the decimal adjust instruction DA A
- Bit 7 — carry (CY); the carry flag indicates that previous operation has resulted in an overflow of the accumulator.

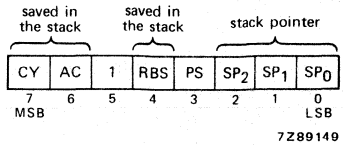


Fig. 16 Program status word.

All bits can be read using the MOV A, PSW instruction. Bits 7 and 6 are set and cleared by CPU operation. Bit 4 can be changed by a SEL RB instruction, bit 3 by the MOV PSW, A instruction, and bits 0, 1 and 2 by the CALL, RET or RETR instructions and in the event of an interrupt. Bits 7, 6 and 4 are stored in the program counter stack during subroutine and interrupt calls. These bits are restored in the PSW with a RETR (return and restore) instruction which must be used at the end of an interrupt and can be used at the end of a normal subroutine. The RET instruction has no restore feature and cannot be used at the end of an interrupt.

Program counter (see Fig. 17).

A 12-bit program counter is used to facilitate 8 K bytes of ROM being addressed. The arrangement of the bits is shown in figure 17. During an interrupt subroutine PC_{11} and PC_{12} are forced to logic 0. All 12 bits are saved in the stack during CALL and interrupt routines.

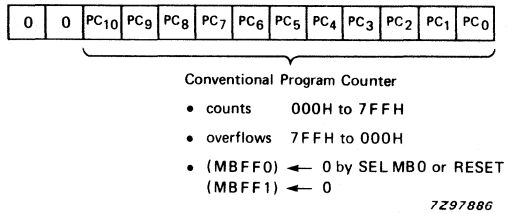


Fig. 17 Program counter.

Central processing unit

The PCD3349 has arithmetic, logical and branching capabilities. The DA A, SWAP A and XCHD instructions simplify BCD arithmetic and the handling of nibbles. The MOVP A,@A instruction permits efficient table look-up from the CURRENT ROM page.

FUNCTIONAL DESCRIPTION (continued)

Conditional branch logic

The conditional branch logic within the processor enables several conditions, internal and external to the processor, to be tested by the user's program.
Table 4 lists the conditional jump instructions used to change the program sequence. The DJNZ instruction decrements a designated register or data memory location and branches if the contents are not zero. This instruction is useful for looping control. The JMPP@A instruction allows multiway branches to destinations determined by the contents of the accumulator.

Table 4 Conditional branches

test	jump condition	jump instruction
accumulator	all bits zero	JZ
	any bit non-zero	JNZ
accumulator bit test	1	JB0 to JB7
carry flag	1	JC
	0	JNC
timer overflow flag	1	JTF
	0	JNTF
test input T0	1	JNT0
	0	JT0*
test input T1	1	JT1
	0	JNT1
register	non-zero	DJNZ

* Because of the inverted interrupt input CE/$\overline{T0}$ the conditional jump JT0 is also inverted.

Test input T1 (pin 8)

The T1 input line can be used as:

- A test input for branch instructions JT1 and JNT1
- An external input to the event counter

When used as a test input:

- JT1 instruction tests for logic 1 level
- JNT1 instruction tests for logic 0 level

When used as an input to the event counter, T1 must be LOW for $>$ 4 CP, followed by a HIGH for $>$ 4 CP. The transition can be recognized with a repetition rate of once per 30 oscillator clock periods (1 machine cycle).

There is no internal pull-up or pull-down resistor connected to the T1 input. If required it must be externally connected to a resistor ($R = \leqslant 100\ k\Omega$).
When T1 is not used pin 8 must be connected to V_{DD} or V_{SS}.

Reset (pin 11)

A positive-going signal on the RESET input/output:

- Sets the program counter to zero
- Selects location 0 of memory bank 0 and register bank 0
- Sets the stack pointer to zero (000); pointing to RAM address 8
- Disables the interrupts (external and timer)
- Stops the timer/event counter, then sets it to zero
- Sets the timer prescaler to modulo-32
- Resets the timer flag
- Sets all ports according to reset states
- Cancels IDLE and STOP mode

After the voltage is applied to RESET an internal delay of 1866 CP is introduced before the microcontroller commences operation.

Power-on reset

The internal power-on reset circuit monitors the supply voltage V_{DD}. As long as V_{DD} remains below the internal reference level V_{ref} (typically 1.3 V), the oscillator is inhibited and RESET (pin 11) has an undefined level. When V_{DD} rises above V_{ref}, the oscillator is released and RESET is pulled HIGH to V_{DD} by TR1 for a period t_D (typically 50 μs). Note that the start-up time of the oscillator is typically 10 ms because of the narrow bandwidth of the crystal.

Three modes of power-on reset are possible:

1. If V_{DD} has a fast rise time, i.e. V_{DD} reaches its minimum value before the RESET signal finishes (t_D), then no additional circuit is required (see Figs 18 and 19). Note that the first instruction is executed after the oscillator start-up time plus 1866 clock periods.
2. If V_{DD} has a slow rise time then the RESET signal should be stretched by an external CR circuit (see Figs 20 and 21). In the event of a short drop in V_{DD}, the diode path discharges the capacitor rapidly to ensure a reliable power-on reset. The RESET signal should reach at least 70% of the final value of V_{DD} to ensure a correct reset. Given that the RESET voltage and V_{DD} rise exponentially, the above requirement is satisfied when the time constant of the RESET pulse is >8 times the time constant of V_{DD}. If V_{DD} rises linearly then a RESET time constant >2 times the rise time of V_{DD} is required.

   When a reset is completed (RESET goes LOW) before the oscillator has started up, program execution begins after the oscillator start-up time plus 1866 clock periods (see Fig. 21). If the oscillator starts up prior to the completion of RESET then program execution begins 1866 clock periods after RESET goes LOW.
3. Fig. 22 shows an external reset applied during power-on. The external reset signal must remain HIGH until V_{DD} has reached its minimum operating value corresponding to the selected oscillator frequency. When a reset is completed (RESET goes LOW) before the oscillator has started up, program execution begins after the oscillator start-up time plus 1866 clock periods (see Fig. 23). If the oscillator starts up prior to the completion of RESET then program execution begins 1866 clock periods after RESET goes LOW.

FUNCTIONAL DESCRIPTION (continued)

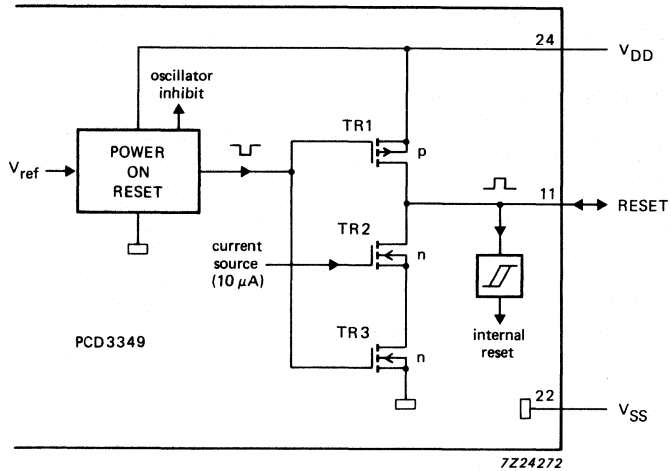


Fig. 18 Power-on reset configuration.

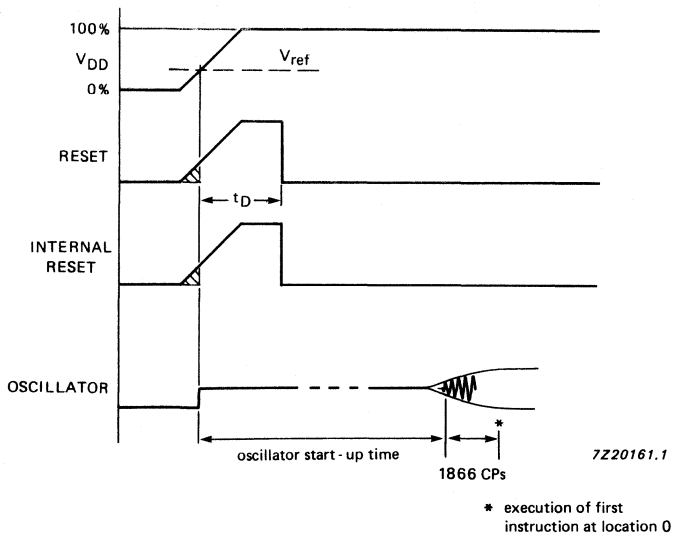


Fig. 19 Timing of power-on reset with fast rise time of V_{DD}.

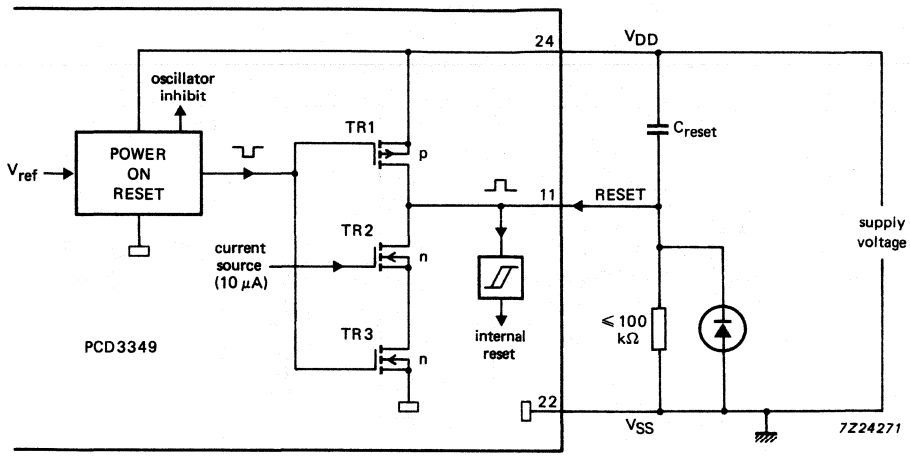


Fig. 20 Stretched power-on reset with external CR circuit.

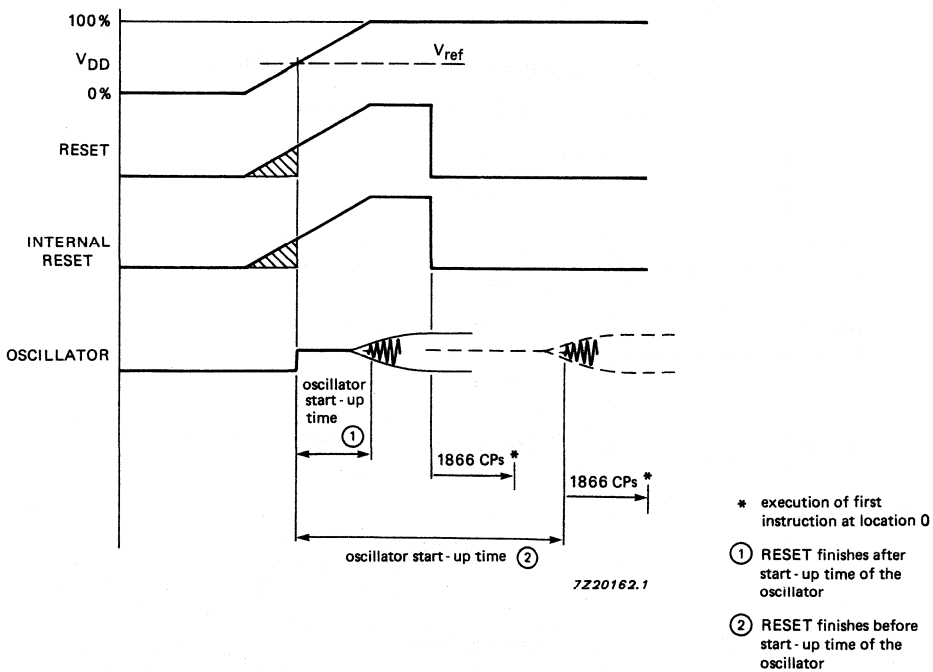


Fig. 21 Timing of power-on reset with a slowly rising V_{DD} and a stretched RESET pulse.

FUNCTIONAL DESCRIPTION (continued)

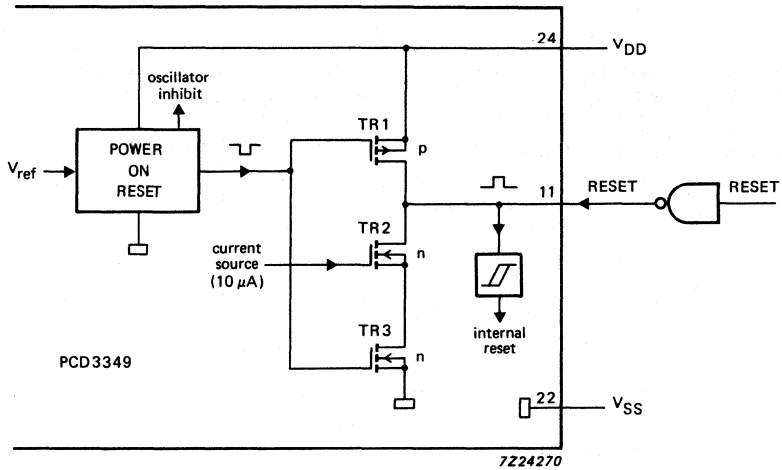


Fig. 22 External power-on reset configuration.

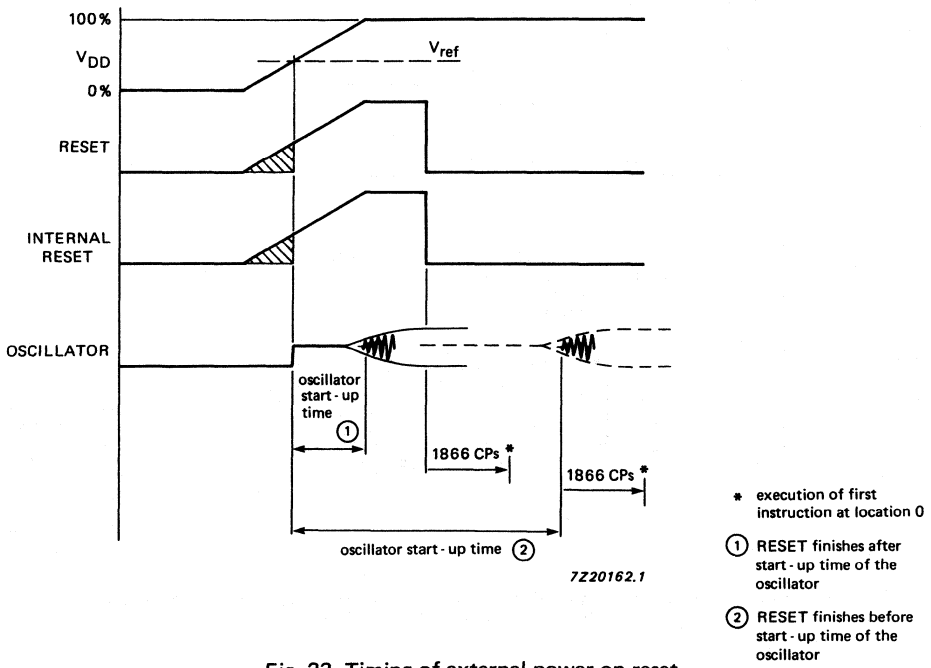


Fig. 23 Timing of external power-on reset.

INSTRUCTION SET

The PCD3349 instruction set consists of over 80 one and two byte instructions and is based on the MAB8048 instruction set. New instructions include those for memory bank selection and derivative control. Program code efficiency is high because all RAM locations and all ROM locations on a 256 byte page require only a single byte address.

Table 5 details the symbols and definition descriptions that are used in the instruction set of the PCD3349. Table 6 shows the instruction map and Table 7 gives the instruction set.

Table 5 Symbols and definitions used in Tables 6 and 7

symbol	definition description
A	accumulator
addr	program memory address
Bb	bit designation (b = 0 to 7)
RBS	register bank select
C	carry bit (bit CY)
CNT	event counter
D	mnemonic for 4-bit digit (nibble)
data	8-bit number or expression
I	interrupt
MB	memory bank
MBFF	memory bank flip-flop
P	mnemonic for 'in page' operation
PC	program counter
Pp	port designation (p = 0, 1 or 2)
PSW	program status word
RB	register bank
Rr	register designation (r = 0 to 7)
SP	stack pointer
T	timer
TF	timer flag
T0, T1	test 0 and 1 inputs
#	immediate data prefix
@	indirect address prefix
(X)	contents of X
((X))	contents of location addressed by X
Dx	derivative register (0 to H'FF')
←	is replaced by
↔	is exchanged with

INSTRUCTION SET (continued)

Table 6 PCD3349 instruction map

first hexadecimal character of opcode ↓ / second hexadecimal character of opcode →	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	NOP	IDLE		ADD A, # data	JMP page 0	EN I	JNTF addr	DEC A	IN A,Pp 0	1	2					
1	INC @Rr 0	1	JB0 addr	ADDC A, # data	CALL page 0	DIS I	JTF addr	INC A	INC Rr 0	1	2	3	4	5	6	7
2	XCH A,@Rr 0	1	STOP	MOV A, # data	JMP page 1	EN TCNTI	JNT0 addr	CLR A	XCH A,Rr 0	1	2	3	4	5	6	7
3	XCHD A,@Rr 0	1	JB1 addr		CALL page 1	DIS TCNTI	JT0 addr	CPL A	OUTL Pp,A 0	1	2					
4	ORL A,@Rr 0	1	MOV A , T	ORL A, # data	JMP page 2	STRT CNT	JNT1 addr	SWAP A	ORL A,Rr 0	1	2	3	4	5	6	7
5	ANL A,@Rr 0	1	JB2 addr	ANL A, # data	CALL page 2	STRT T	JT1 addr	DA A	ANL A,Rr 0	1	2	3	4	5	6	7
6	ADD A,@Rr 0	1	MOV T , A		JMP page 3	STOP TCNT		RRC A	ADD A,Rr 0	1	2	3	4	5	6	7
7	ADDC A,@Rr 0	1	JB3 addr		CALL page 3			RR A	ADDC A,Rr 0	1	2	3	4	5	6	7
8				RET	JMP page 4				ORL Pp, # data 0	1	2			MOV Dx,A		
9			JB4 addr	RETR	CALL page 4		JNZ addr	CLR C	ANL Pp, # data 0	1	2					
A	MOV @Rr, A 0	1		MOVP A,@A	JMP page 5	SEL MB2		CPL C	MOV Rr,A 0	1	2	3	4	5	6	7
B	MOV @Rr, # data 0	1	JB5 addr	JMPP @A	CALL page 5	SEL MB3			MOV Rr, # data 0	1	2	3	4	5	6	7
C	DEC @Rr 0	1			JMP page 6	SEL RB0	JZ addr	MOV A, PSW	DEC Rr 0	1	2	3	4	5	6	7
D	XRL A,@Rr 0	1	JB6 addr	XRL A, # data	CALL page 6	SEL RB1		MOV PSW, A	XRL A,Rr 0	1	2	3	4	5	6	7
E	DJNZ @Rr,addr 0	1			JMP page 7	SEL MB0	JNC addr	RL A	DJNZ Rr, addr 0	1	2	3	4	5	6	7
F	MOV A,@Rr 0	1	JB7 addr		CALL page 7	SEL MB1	JC addr	RLC A	MOV A,Rr 0	1	2	3	4	5	6	7

Table 7 Instruction set

	mnemonic	opcode (hex.)	bytes/ cycles	description	function		notes
ACCUMULATOR	ADD A, Rr	6*	1/1	Add register contents to A	(A)←(A) + (Rr)	r = 0–7	1
	ADD A, @Rr	60 61	1/1	Add RAM data, addressed by Rr, to A	(A)←(A) + ((R0)) (A)←(A) + ((R1))		1
	ADD A, #data	03 data	2/2	Add immediate data to A	(A)←(A) + data		1
	ADDC A, Rr	7*	1/1	Add carry and register contents to A	(A)←(A) + (Rr) + (C)	r = 0–7	1
	ADDC A, @Rr	70 71	1/1	Add carry and RAM data, addressed by Rr, to A	(A)←(A) + ((R0)) + (C) (A)←(A) + ((R1)) + (C)		1
	ADDC A, #data	13 data	2/2	Add carry and immediate data to A	(A)←(A) + data + (C)		1
	ANL A, Rr	5*	1/1	'AND' Rr with A	(A)←(A) AND (Rr)	r = 0–7	
	ANL A, @Rr	50 51	1/1	'AND' RAM data, addressed by Rr, with A	(A)←(A) AND ((R0)) (A)←(A) AND ((R1))		
	ANL A, #data	53 data	2/2	'AND' immediate data with A	(A)←(A) AND data		
	ORL A, Rr	4*	1/1	'OR' Rr with A	(A)←(A) OR (Rr)	r = 0–7	
	ORL A, @Rr	40 41	1/1	'OR' RAM data, addressed by Rr, with A	(A)←(A) OR ((R0)) (A)←(A) OR ((R1))		
	ORL A, #data	43 data	2/2	'OR' immediate data with A	(A)←(A) OR data		
	XRL A, Rr	D*	1/1	'XOR' Rr with A	(A)←(A) XOR (Rr)	r = 0–7	
	XRL A, @Rr	D0 D1	1/1	'XOR' RAM, addressed by Rr, with A	(A)←(A) XOR ((R0)) (A)←(A) XOR ((R1))		
	XRL A, #data	D3 data	2/2	'XOR' immediate data with A	(A)←(A) XOR data		
	INC A	17	1/1	increment A by 1	(A)←(A) + 1		
	DEC A	07	1/1	decrement A by 1	(A)←(A) −1		
	CLR A	27	1/1	clear A to zero	(A)←0		
	CPL A	37	1/1	one's complement A	(A)←NOT(A)		
	RL A	E7	1/1	rotate A left	$(A_{n+1})\leftarrow(A_n)$ $(A_0)\leftarrow(A_7)$	n = 0–6	

INSTRUCTION SET (continued)

Table 7 (continued)

ACCUMULATOR (cont.)	RLC A	F7	1/1	rotate A left through carry	$(A_{n+1})\leftarrow A_n$ $(A_0)\leftarrow(C)$, $(C)\leftarrow(A_7)$	n = 0–6	2
	RR A	77	1/1	rotate A right	$(A_n)\leftarrow(A_{n+1})$ $(A_7)\leftarrow(A_0)$	n = 0–6	
	RRC A	67	1/1	rotate A right through carry	$(A_n)\leftarrow(A_{n+1})$ $(A_7)\leftarrow(C)$, $(C)\leftarrow(A_0)$	n = 0–6	2
	DA A	57	1/1	decimal adjust A			2
	SWAP A	47	1/1	swap nibbles of A	$(A_{4-7})\leftrightarrow(A_{0-3})$		
DATA MOVES	MOV A, Rr	F*	1/1	move register contents to A	(A)←(Rr)	r = 0–7	
	MOV A, @Rr	F0 F1	1/1	move RAM data, addressed by Rr, to A	(A)←((R0)) (A)←((R1))		
	MOV A, #data	23 data	2/2	move immediate data to A	(A)←data		
	MOV Rr, A	A*	1/1	move accumulator contents to register	(Rr)←(A)	r = 0–7	
	MOV @Rr, A	A0 A1	1/1	move accumulator contents to RAM location addressed by Rr	((R0))←(A) ((R1))←(A)		
	MOV Rr, #data	B* data	2/2	move immediate data to Rr	(Rr)←data	r = 0–7	
	MOV @Rr, #data	B0 data B1 data	2/2	move immediate data to RAM location addressed by Rr	((R0))←data ((R1))←data		
	XCH A, Rr	2*	1/1	exchange accumulator contents with Rr	(A)↔(Rr)	r = 0–7	
	XCH A, @Rr	20 21	1/1	exchange accumulator contents with RAM data addressed by Rr	(A)↔((R0)) (A)↔((R1))		
	XCHD A, @Rr	30 31	1/1	exchange lower nibbles of A and RAM data addressed by Rr	$(A_{0-3})\leftrightarrow((R0_{0-3}))$ $(A_{0-3})\leftrightarrow((R1_{0-3}))$		
	MOV A, PSW	C7	1/1	move PSW contents to accumulator	(A)←(PSW)		
	MOV PSW, A	D7	1/1	move accumulator bit 3 to PSW_3	$(PSW_3)\leftarrow(A_3)$		3
	MOVP A, @A	A3	1/2	move indirectly addressed data in current page to A	$(PC_{0-7})\leftarrow(A)$, (A)←((PC))		
FLAGS	CLR C	97	1/1	clear carry bit	(C)←0		2
	CPL C	A7	1/1	complement carry bit	(C)←NOT(C)		2

	mnemonic	opcode (hex.)	bytes/ cycles	description	function		notes
REGISTER	INC Rr	1*	1/1	increment register by 1	(Rr)←(Rr) + 1	r = 0–7	
	INC @Rr	10 11	1/1	increment RAM data, addressed by Rr, by 1	((R0))←((R0)) + 1 ((R1))←((R1)) + 1		
	DEC Rr	C*	1/1	decrement register by 1	(Rr)←(Rr) – 1	r = 0–7	
	DEC @Rr	C0 C1	1/1	decrement RAM data, addressed by Rr, by 1	((R0))←((R0)) – 1 ((R1))←((R1)) – 1		
BRANCH	JMP addr	● 4 address	2/2	unconditional jump within a 2 K bank	(PC_{8-10})←$addr_{8-10}$ (PC_{0-7})←$addr_{0-7}$ (PC_{11-12})←MBFF 0–1		
	JMPP @A	B3	1/2	indirect jump within a page	(PC_{0-7})←((A))		
	DJNZ Rr, addr	E* address	2/2	decrement Rr by 1 and jump if not zero to addr	(Rr)←(Rr) – 1 if (Rr) not zero (PC_{0-7})←addr	r = 0–7	
	DJNZ @Rr, addr	E0 address	2/2	decrement RAM data, addressed by Rr by 1 and jump if not zero to addr	((R0))←((R0)) – 1 if ((R0)) not zero (PC_{0-7})←addr		
		E1 address			((R1))←((R1)) – 1 if ((R1)) not zero (PC_{0-7})←addr		
	JBb addr	▲ 2 address	2/2	jump to addr if Acc. bit b = 1	if b = 1 : (PC_{0-7})←addr	b = 0–7	
	JC addr	F6 address	2/2	jump to addr if C = 1	if C = 1 : (PC_{0-7})←addr		
	JNC addr	E6 address	2/2	jump to addr if C = 0	if C = 0 : (PC_{0-7})←addr		
	JZ addr	C6 address	2/2	jump to addr if A = 0	if A = 0 : (PC_{0-7})←addr		
	JNZ addr	96 address	2/2	jump to addr if A is NOT zero	if A≠0 : (PC_{0-7})←addr		
	JT0 addr	36 address	2/2	jump to addr if T0 = 1	if T0 = 1: (PC_{0-7})←addr		
	JNT0 addr	26 address	2/2	jump to addr if T0 = 0	if T0 = 0: (PC_{0-7})←addr		
	JT1 addr	56 address	2/2	jump to addr if T1 = 1	if T1 = 1: (PC_{0-7})←addr		
	JNT1 addr	46 address	2/2	jump to addr if T1 = 0	if T1 = 0: (PC_{0-7})←addr		
	JTF addr	16 address	2/2	jump to addr if Timer Flag = 1	if TF = 1: (PC_{0-7})←addr		4
	JNTF addr	06 address	2/2	jump to addr if Timer Flag = 0	if TF = 0: (PC_{0-7})←addr		

INSTRUCTION SET (continued)

Table 7 (continued)

TIMER/EVENT COUNTER	MOV A, T	42	1/1	move timer/event counter contents to accumulator	(A)←(T)	
	MOV T, A	62	1/1	move accumulator contents to timer/event counter	(T)←(A)	
	STRT CNT	45	1/1	start event counter		
	STRT T	55	1/1	start timer		
	STOP TCNT	65	1/1	stop timer/event counter		
	EN TCNTI	25	1/1	enable timer/event counter interrupt		
	DIS TCNTI	35	1/1	disable timer/event counter interrupt		
CONTROL	EN I	05	1/1	enable external interrupt		
	DIS I	15	1/1	disable external interrupt		
	SEL RB0	C5	1/1	select register bank 0	(RBS)←0	5
	SEL RB1	D5	1/1	select register bank 1	(RBS)←1	5
	SEL MB0	E5	1/1	select program memory bank 0	(MBFF0)←0, (MBFF1)←0	
	SEL MB1	F5	1/1	select program memory bank 1	(MBFF0)←1, (MBFF1)←0	
	SEL MB2	A5	1/1	select program memory bank 2	(MBFF0)←0, (MBFF1)←1	
	SEL MB3	B5	1/1	select program memory bank 3	(MBFF0)←1, (MBFF1)←1	
	STOP	22	1/1	enter STOP mode		
	IDLE	01	1/1	enter IDLE mode		
SUBROUTINE	CALL addr	▲ 4 address	2/2	jump to subroutine	((SP))←(PC), ($PSW_{4, 6, 7}$) (SP)←(SP) + 1 (PC_{8-10})←$addr_{8-10}$ (PC_{0-7})←$addr_{0-7}$ (PC_{11-12})←MBFF 0–1	6
	RET	83	1/2	return from subroutine	(SP)←(SP) – 1 (PC)←((SP))	6
	RETR	93	1/2	return from interrupt and restore bits 4, 6, 7 of PSW	(SP)←(SP) – 1 ($PSW_{4, 6, 7}$) + (PC)←((SP))	6

	mnemonic	opcode (hex.)	bytes/ cycles	description	function	notes
PARALLEL INPUT/OUTPUT	IN A, Pp	08 09 0A	1/2	input port p data to accumulator	(A)←(P0) (A)←(P1) (A)←(P2)	7
	OUTL Pp, A	38 39 3A	1/2	output accumulator data to port p	(P0)←(A) (P1)←(A) (P2)←(A)	
	ANL Pp, #data	98 data 99 data 9A data	2/2	AND port p data with immediate data	(P0)←(P0) AND data (P1)←(P1) AND data (P2)←(P2) AND data	
	ORL Pp, #data	88 data 89 data 8A data	2/2	OR port p data with immediate data	(P0)←(P0) OR data (P1)←(P1) OR data (P2)←(P2) OR data	
DERIVATIVE INPUT/OUTPUT	MOV Dx, A	8D	2/2	move accumulator contents to derivative register	(Dx)←(A) x = 0 to 255	8
	NOP	00	1/1	no operation		

Notes to Table 7

1. PSW CY, AC affected
2. PSW CY affected
3. PSW PS affected
4. Execution of JTF and JNTF instructions resets the Timer Flag (TF).
5. PSW RBS affected
6. PSW SP_0, SP_1, SP_2 affected
7. (A) = 0000 P2.3, P2.2, P2.1, P2.0
8. Instructions for PCF84C00 only.

* : 8, 9, A, B, C, D, E, F
● : 0, 2, 4, 6, 8, A, C, E
▲ : 1, 3, 5, 7, 9, B, D, F

RATINGS

Limiting values in accordance with the Absolute Maximum System (IEC 134)

parameter	conditions	symbol	min.	max.	unit
Supply voltage (pin 24)		V_{DD}	−0.8	+ 8	V
Input voltage (any pin)		V_I	−0.8	V_{DD} + 0.8	V
DC current (any input or output)		$\pm I_I, \pm I_O$	–	10	mA
Total power dissipation	note 1	P_{tot}	–	500	mW
Power dissipation (per output)		P_O	–	50	mW
Storage temperature range		T_{stg}	−65	+ 150	°C
Operating ambient temperature range		T_{amb}	−25	+ 70	°C
Operating junction temperature		T_j	–	+ 125	°C

Note to the ratings

1. Derate according to thermal resistance.

THERMAL RESISTANCE

From junction to ambient

SOT117	$R_{th\ j\text{-}a}$	=	120	K/W
SOT136A	$R_{th\ j\text{-}a}$	=	150	K/W

DEVELOPMENT DATA

DC CHARACTERISTICS

V_{DD} = 2.5 to 6 V; V_{SS} = 0 V; T_{amb} = −25 to + 70 °C; all voltages with respect to V_{SS}; f = 3.58 MHz with R_S = 100 Ω; unless otherwise specified

parameter	conditions	symbol	min.	typ.	max.	unit
Supplies						
Supply voltage operating		V_{DD}	2.5	–	6	V
STOP mode for RAM data retention		V_{DD}	1.0	–	6	V
Supply current (Fig. 25)						
operating with tone generator on	V_{DD} = 3 V	I_{DD}	–	1.2	–	mA
operating without tone generator	V_{DD} = 3 V	I_{DD}	–	600	–	μA
IDLE mode (Fig. 26)						
with tone generator on	V_{DD} = 3 V	I_{DD}	–	900	–	μA
without tone generator	V_{DD} = 3 V	I_{DD}	–	300	–	μA
STOP mode (Fig. 27)	note 1; V_{DD} = 1.8 V					
	T_{amb} = 25 °C	I_{DD}	–	1.2	2.5	μA
	T_{amb} = 55 °C	I_{DD}	–	–	5	μA
	T_{amb} = 70 °C	I_{DD}	–	–	10	μA
RESET I/O						
Switching level		V_{RESET}	–	1.3	–	V
Sink current	$V_{DD} > V_{RESET}$	I_{OL}	–	7	–	μA
Inputs						
Input voltage LOW		V_{IL}	0	–	0.3 V_{DD}	V
Input voltage HIGH		V_{IH}	0.7 V_{DD}	–	V_{DD}	V
Input leakage current	$V_{SS} < V_I < V_{DD}$	$\pm I_{IL}$	–	–	1	μA
Outputs						
Output voltage LOW	V_I = V_{SS} or V_{DD}; $\lvert I_O \rvert < 1$ μA	V_{OL}	–	–	0.05	V
Output sink current LOW P1.2, P1.3, P1.4, P2.0, P2.1, P2.2, P2.3 (Fig. 28)	V_{DD} = 3 V; V_O = 0.4 V	I_{OL}	0.7	1.5	–	mA
for remaining ports (Fig. 29)	V_O = 0.4 V	I_{OL}	1.5	–	–	mA
Pull-up output source current HIGH (Fig. 30)	V_{DD} = 3 V; V_O = 0.9 V_{DD}	$-I_{OH}$	10	–	–	μA
	V_O = V_{SS}	$-I_{OH}$	–	–	300	μA
Push-pull output source current HIGH	V_{DD} = 3 V; V_O = V_{DD} −0.4 V	$-I_{OH}$	0.6	1.5	–	mA

TONE GENERATOR CHARACTERISTICS

V_{DD} = 2.5 to 6 V; V_{SS} = 0 V; T_{amb} = −25 to + 70 °C; all voltages with respect to V_{SS}; f = 3.58 MHz with R_S = 100 Ω; unless otherwise specified

parameter	conditions	symbol	min.	typ.	max.	unit
Tone output (Fig. 24)						
DTMF output voltage levels (RMS values)						
HIGH group		$V_{HG(rms)}$	158	192	205	mV
LOW group		$V_{LG(rms)}$	125	150	160	mV
Frequency deviation		$\Delta f/f$	−0.6	–	+ 0.6	%
DC voltage level		V_{dc}	–	½ V_{DD}	–	V
Output impedance		$\|Z_O\|$	–	0.1	0.5	kΩ
Pre-emphasis of group		ΔV_G	1.85	2.1	2.35	dB
Total harmonic distortion	note 2; T_{amb} = 25 °C	THD	–	−25	–	dB

Notes to the characteristics

1. Crystal connected between XTAL1 and XTAL2; CE and T1 at V_{SS}.
2. Related to the level of the LOW group frequency component (CEPT CS 203).

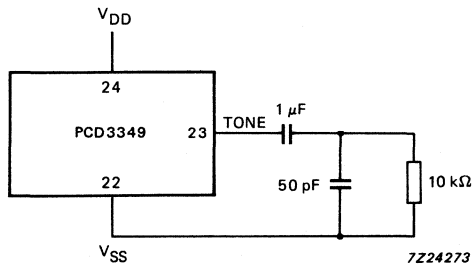


Fig. 24 Tone output test circuit.

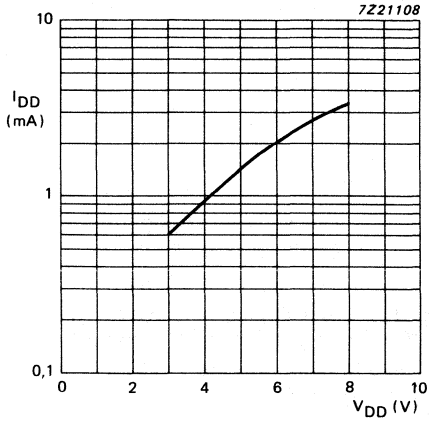


Fig. 25 Typical supply current (I_{DD}) in operating mode as a function of the supply voltage (V_{DD}); T_{amb} = 25 °C; clock frequency = 3.58 MHz; I_{DD} is increased by approximately 0.6 mA when the DTMF function is operating.

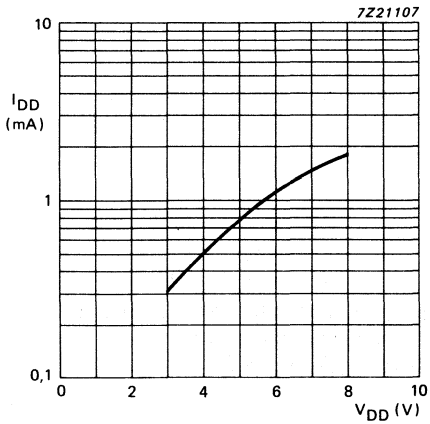
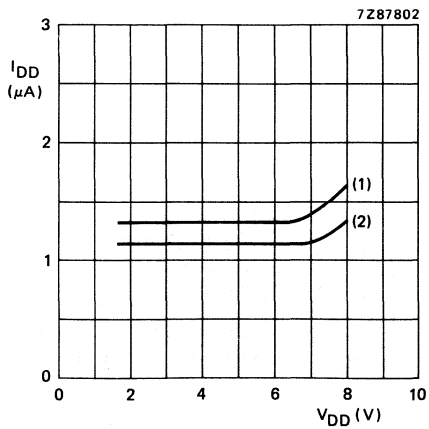
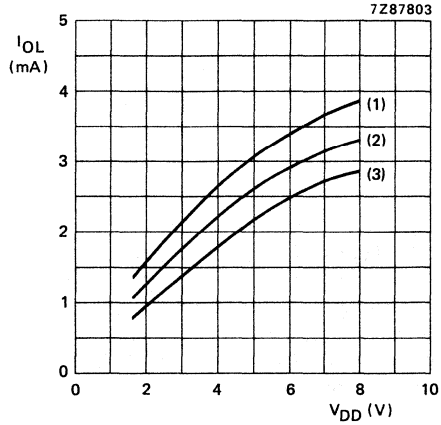


Fig. 26 Typical supply current (I_{DD}) in IDLE mode as a function of the supply voltage (V_{DD}); T_{amb} = 25 °C; clock frequency = 3.58 MHz.



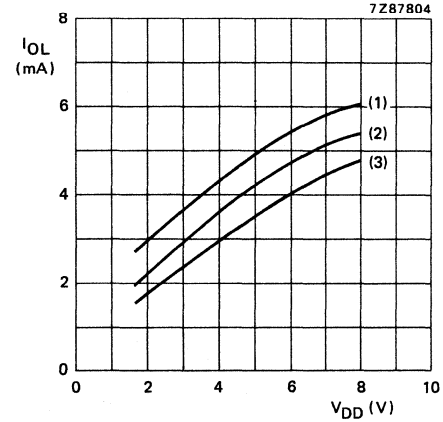
(1) T_{amb} = 25 °C
(2) T_{amb} = 70 °C

Fig. 27 Typical supply current (I_{DD}) in STOP mode as a function of the supply voltage (V_{DD}).



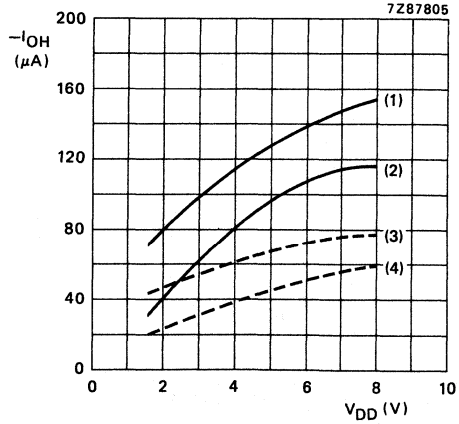
(1) $T_{amb} = -25$ °C
(2) $T_{amb} = +25$ °C
(3) $T_{amb} = +70$ °C

Fig. 28 Output sink current LOW (I_{OL}), for P1.2, P1.3, P1.4, P2.0, P2.1, P2.2, P2.3 as a function of supply voltage (V_{DD}); $V_O = 0.4$ V.



(1) $T_{amb} = -25$ °C
(2) $T_{amb} = +25$ °C
(3) $T_{amb} = +70$ °C

Fig. 29 Output sink current LOW (I_{OL}), for remaining ports as a function of supply voltage (V_{DD}); $V_O = 0.4$ V.



(1) $T_{amb} = 25$ °C; $V_O = V_{SS}$
(2) $T_{amb} = 25$ °C; $V_O = 0.9\ V_{DD}$
(3) $T_{amb} = 70$ °C; $V_O = V_{SS}$
(4) $T_{amb} = 70$ °C; $V_O = 0.9\ V_{DD}$

Fig. 30 Output source current HIGH ($-I_{OH}$) as a function of supply voltage (V_{DD}).

DEVELOPMENT DATA

APPLICATION INFORMATION

A block diagram of an electronic featurephone built around the PCD3349 is shown in Figure 31. It comprises the following dedicated telephony ICs:

- TEA1060/1061/1067/1068 transmission circuit for telephony
- PCF8576 or PCF8577 2 LCD drivers in LCD module MB7020160
- PCF8571 1 K RAMs with Serial I/O; the number of RAMs depends on the required amount of stored telephone numbers
- PCD3360 programmable multi-tone ringer

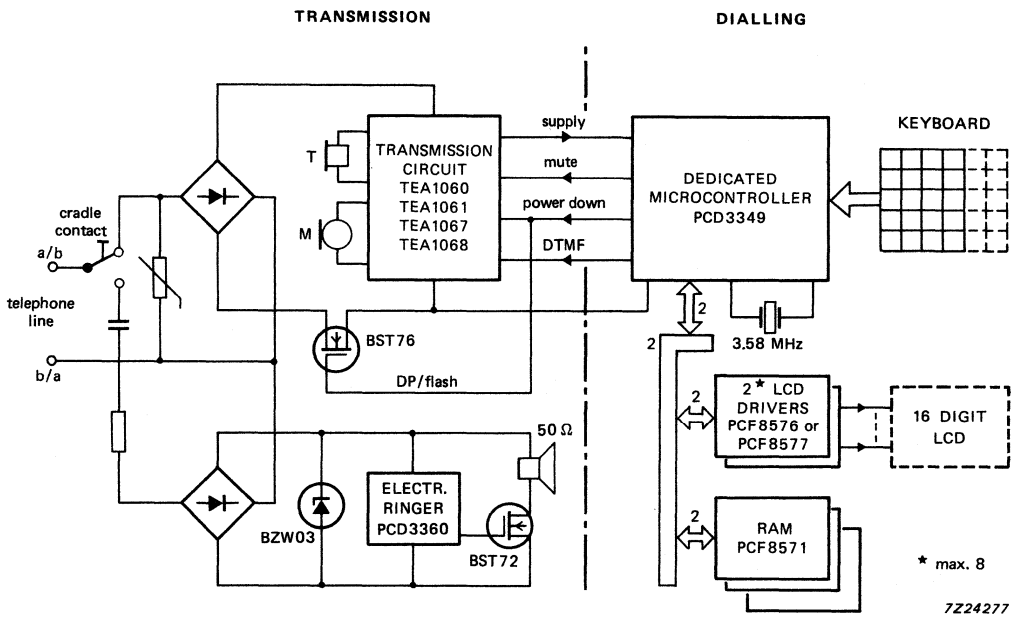


Fig. 31 Block diagram of electronic featurephone with common line interface.

SINGLE-CHIP 8-BIT MICROCONTROLLER FAMILY

DESCRIPTION

An advanced CMOS process is used to manufacture the PCF84CXXX microcontroller family. The family consists of the following devices:

- PCF84C00
- PCF84C12
- PCF84C121
- PCF84C430
- PCF84C21
- PCF84C22
- PCF84C230
- PCF84C470
- PCF84C41
- PCF84C42
- PCF84C270
- PCF84C640
- PCF84C81
- PCF84C85
- PCF84C271

This data sheet describes features of the PCF84CXXX microcontroller family which are common to several family members. For details on a particular device, consult the relevant data sheet.

All family members have quasi-bidirectional I/O port lines, a single-level vectored interrupt structure, an 8-bit timer/event counter and on-chip clock oscillator and clock circuits.

These efficient controllers also perform well as arithmetic processors. They have facilities for both binary and BCD arithmetic plus bit-handling capabilities. The instruction set is similar to that of the MAB8048 and the PCF84CXXX family is very similar to the MAB8400 family.

Features common to all family members are listed below.

Features

- 8-bit CPU, ROM, RAM, I/O in a single DIL or SO package
- 1 K, 2 K, 4 K or 8 K x 8 ROM; there is also a ROM-less device
- 64, 128 or 256 x 8 RAM
- Quasi-bidirectional I/O port lines
- Two test inputs: one of which is also an external interrupt input
- Single-level vectored interrupt structure
- 8-bit programmable timer/event counter
- Clock frequency range: 100 kHz to 10 MHz
- Over 80 instructions (similar to those of the MAB8048) all of 1 or 2 cycles
- Single supply voltage (2,5 V to 5,5 V)
- STOP and IDLE modes
- Power-on-reset circuit
- Operating temperature range: −40 to + 85 °C

PACKAGE OUTLINES

Consult individual data sheets.

Contents	Page no.	Contents	Page no.
Block diagram	2	Oscillator	18
Program memory	3	Timer/event counter	19
Data memory	3	Program status word	20
Program and Data memory maps	4	Program counter	20
IDLE and STOP modes	5	CPU	21
I/O facilities	7	Conditional branch logic	21
Serial I/O	10	Test input T1	22
Interrupts	15	Power-on-reset	22

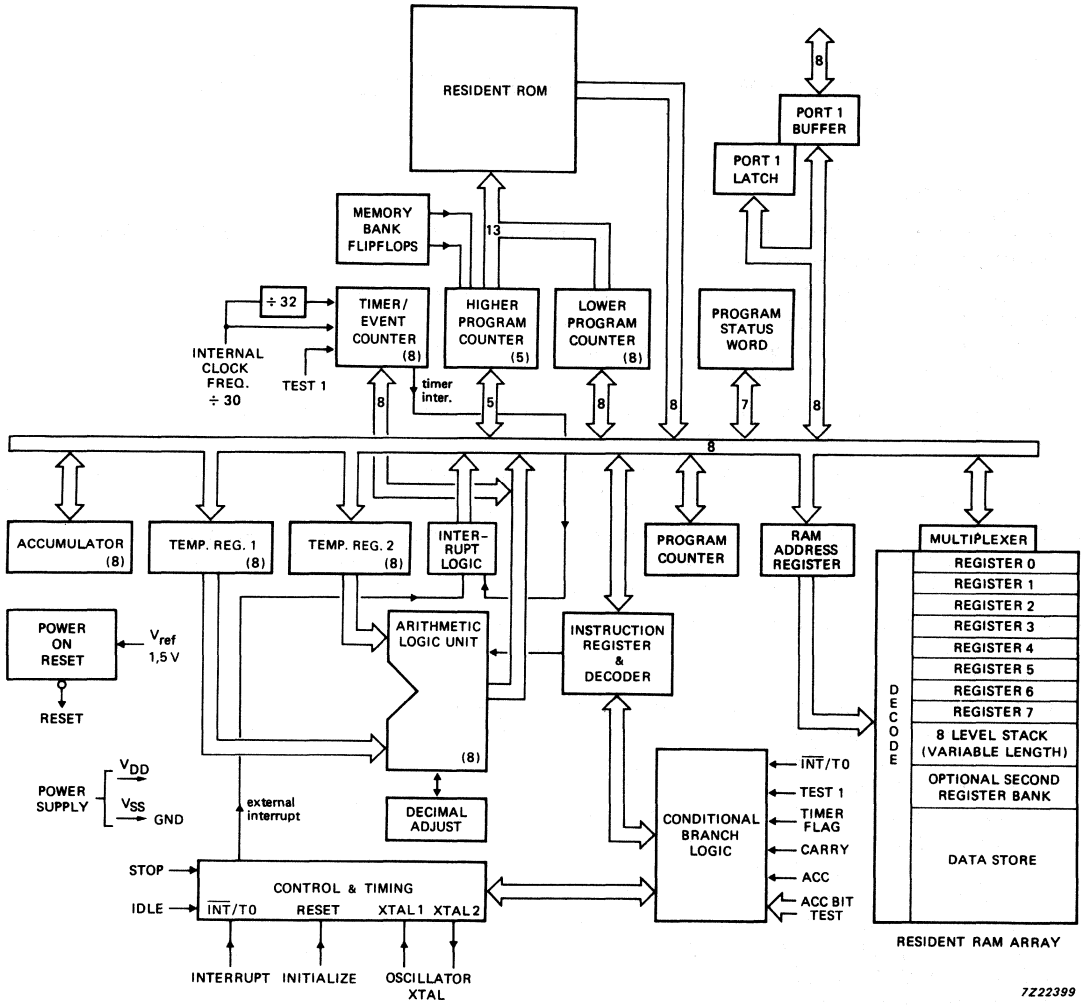


Fig. 1 PCF84CXXX block diagram.

FUNCTIONAL DESCRIPTION

Program memory

The program memory consists of 1 K, 2 K, 4 K or 8 K bytes of read-only memory (ROM). Each location is directly addressable by the program counter. The ROM is mask-programmed at the factory. Fig. 2 shows the program memory map.

Four program memory locations are of special importance:

- Location 0: contains the first instruction to be executed after the microcontroller is initialized (RESET)
- Location 3: contains the first byte of an external interrupt service routine
- Location 5: contains the first byte of a serial I/O and derivative interrupt service routine
- Location 7: contains the first byte of a timer/event counter interrupt service routine

Program memory is arranged in banks of 2 K bytes, which are selected by SEL MB instructions (family members with more than 2 K ROM only). The program memory is further divided into location 'pages', each of 256 bytes. This latter division applies only for conditional branches. Memory bank boundaries can be crossed only by using unconditional branch instructions after the appropriate memory bank has been selected. A CALL instruction can transfer control to a subroutine on any 'page'; RET and RETR instructions can transfer control from a subroutine back to the main program.

Data memory

Data memory consists of 64, 128 or 256 bytes of random-access memory (RAM). All locations are indirectly addressable using RAM pointer registers; up to 16 designated locations are directly addressable. Memory also includes an 8-level program counter stack addressed by a 3-bit stack pointer. Fig. 3 shows the data memory map.

Working registers

Locations 0 to 7 are designated as working registers, directly addressable by the direct register instructions. Ease of addressing, and a minimum requirement of instruction bytes to manipulate their contents, makes these locations suitable for storing frequently addressed intermediate results. This bank of registers can be selected by the SEL RB0 instruction.

Executing the select register bank instruction SEL RB1, designates locations 24 to 31 as working registers, instead of locations 0 to 7, and these are then directly addressable. This second bank of working registers may be used as an extension of the first or reserved for use during interrupt service routines saving the first bank for use in the main program. If the second bank is not used, locations 24 to 31 may serve as general purpose RAM.

The first two locations of each bank contain the RAM pointer registers R0, R1, R0' and R1', which indirectly address all RAM locations.

All RAM locations make efficient program loop counters when used with the decrement register and test instruction DJNZ.

FUNCTIONAL DESCRIPTION (continued)

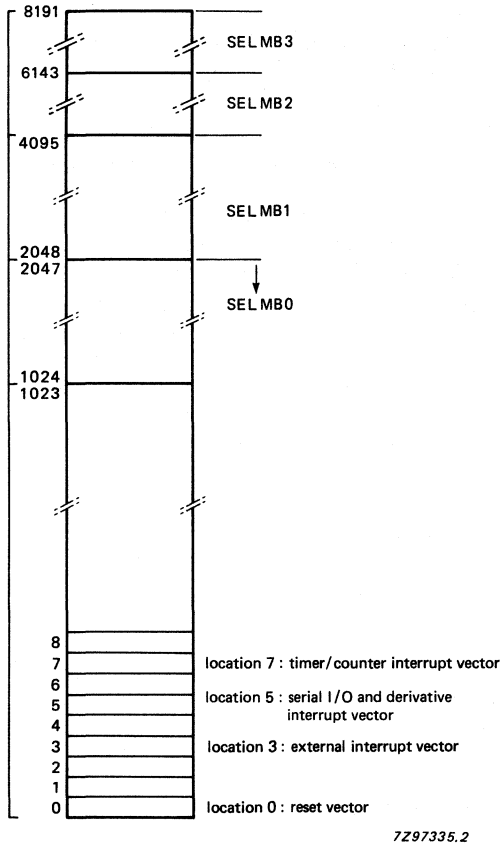


Fig. 2 Program memory map.

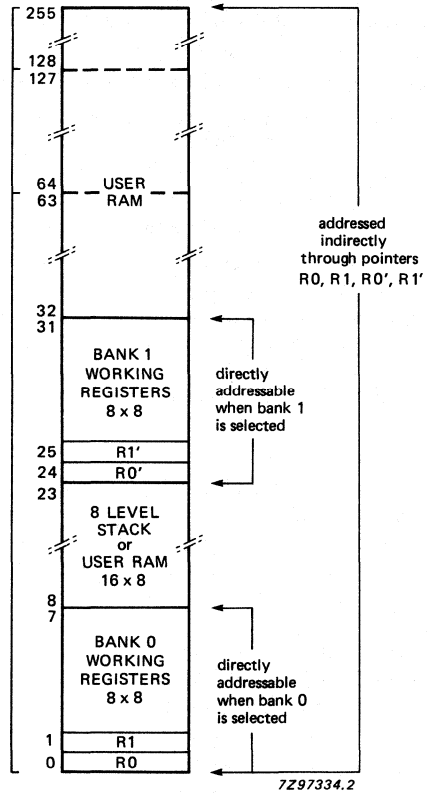


Fig. 3 Data memory map.

Program counter stack

Locations 8 to 23 may be designated as an 8-level prógram counter stack (2 locations per level), or as general purpose RAM. The program counter stack (Fig. 4) enables the microcontroller to keep track of the return addresses and status prior to interrupts or CALL instructions by storing the contents of the program counter before servicing the subroutine. A 3-bit stack pointer determines which of the eight register pairs of the program counter stack will be loaded with the next return address and status.

The stack pointer, when initialized to 000 by RESET, points to RAM locations 8 and 9. On the first subroutine CALL or interrupt, the contents of the program counter and bits 4, 6 and 7 of the program status word (PSW) are transferred to locations 8 and 9. The stack pointer increments by one and points to locations 10 and 11, ready for another CALL. Because an address may be up to 13 bits long, two bytes must be used to store each address.

At the end of a subroutine, which is signalled by a return instruction (RET or RETR), the stack pointer decrements by one and the contents of the register pair on top of the stack are transferred to the program counter. The saved PSW bits are transferred to the PSW only by the RETR instruction.

If all 8 levels of subroutine and interrupt nesting are not used, the unused portion of the stack may be used as any other indirectly addressable RAM locations.

Nesting of subroutines within subroutines can continue up to 8 times without overflowing the stack. If overflow does occur the deepest address stored (locations 8 and 9) will be overwritten and lost since the stack pointer overflows from 111 to 000. It also underflows from 000 to 111.

The value of the saved contents of the program counter during an interrupt CALL is not the same as the value saved during a normal CALL to subroutine. During an interrupt CALL, the program counter return address is saved; during a subroutine CALL, the saved program counter value is one less than the program counter return address.

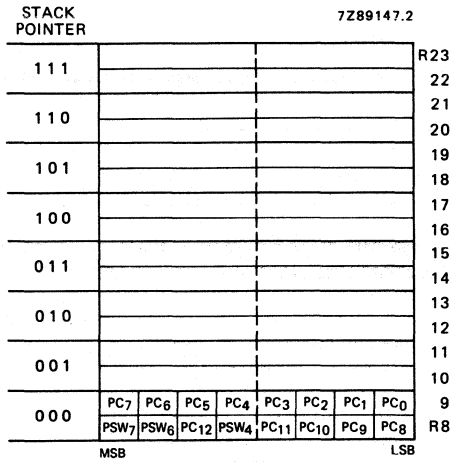


Fig. 4 Program counter stack.

IDLE and STOP modes

IDLE mode

When the microcontroller enters the IDLE mode via the IDLE instruction (01 H), the oscillator, timer/counter and serial I/O are kept running. The microcontroller exits from the IDLE mode via a RESET or one of three interrupts if they are enabled. If the interrupt is not enabled, the microcontroller will remain in the IDLE mode. An active signal on the RESET pin restarts the microcontroller and a normal RESET sequence is executed (see Fig. 5).

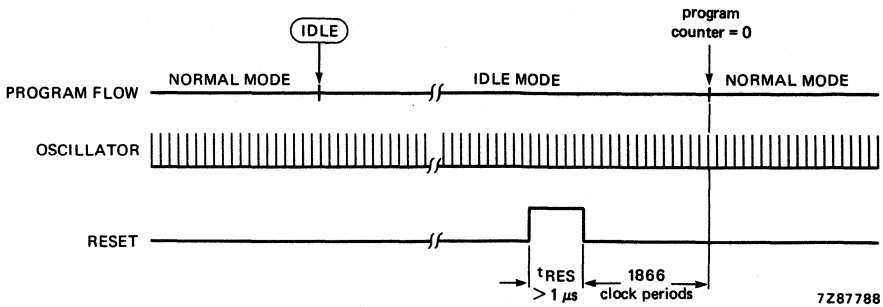


Fig. 5 Exit from IDLE mode via a RESET.

FUNCTIONAL DESCRIPTION (continued)

An active signal coming from an enabled interrupt causes the execution of the normal interrupt routine since normal interrupt scanning is still being carried out. A HIGH-to-LOW transition on the external interrupt pin ($\overline{INT}$/T0) reactivates the microcontroller. A LOW level applied to $\overline{INT}$/T0 will reactivate the microcontroller only in the STOP mode. Thus, if $\overline{INT}$/T0 was LOW before the microcontroller entered the IDLE mode, it must go HIGH before the microcontroller can be reactivated (see Fig. 6).

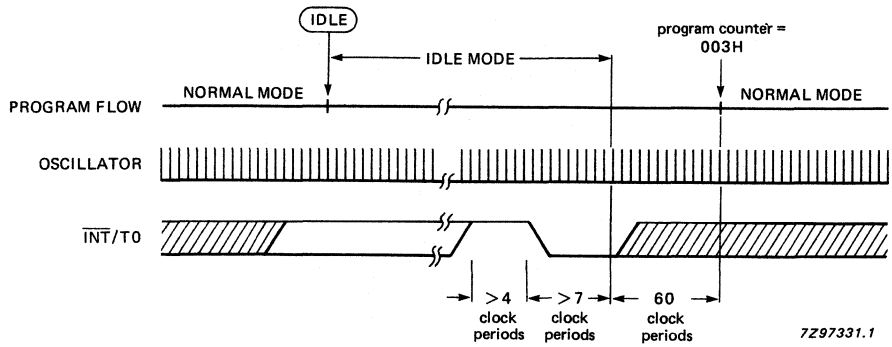


Fig. 6 Exit from IDLE mode via an interrupt.

Wake-up from the IDLE mode is ensured when $\overline{INT}$/T0 is HIGH for at least 4 CP (clock periods) and then LOW for 7 CP. After the initial forced CALL 003 H operation (60 CP), the program continues with the external interrupt service routine. During the STOP mode, the address of the instruction immediately following the instruction that caused the microcontroller to enter the IDLE mode is present on the address bus.

STOP mode

The microcontroller enters the STOP mode via the STOP instruction (22 H). The oscillator is switched off and the internal status of the CPU, RAM contents and the state of I/O ports are not affected. The microcontroller can be brought out of the STOP mode by an active signal at the external interrupt input or by an external RESET signal. When one of these two signals is applied, an internal delay of 1866 CP is provided to ensure that all internal clocks are operating correctly before restart (see Fig. 7).

Note: the start-up time of a crystal oscillator is measured in milliseconds, and the 1866 CP count begins after this start-up time.

If the microcontroller exits from the STOP mode by activating RESET, a normal RESET sequence is executed.

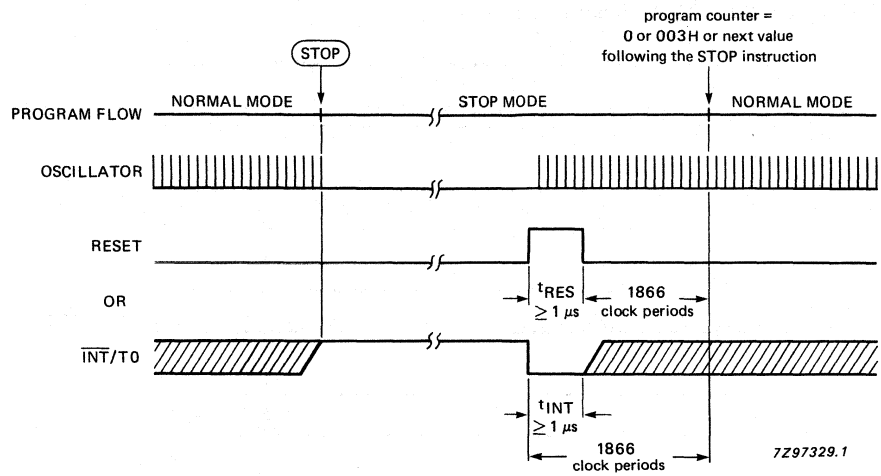


Fig. 7 Entering and exiting the STOP mode.

If the microcontroller exits the STOP mode by pulling the external interrupt input pin LOW, an interrupt sequence is executed only if the external interrupt is enabled. In this event the microcontroller executes one instruction of the main program (the instruction following the STOP instruction) and then calls the interrupt routine. If the interrupt is not enabled, it continues the normal program sequence by executing the instruction following the STOP instruction.

The microcontroller is restarted by a LOW level applied at the $\overline{INT}$/T0 pin.

Note: When exiting the STOP mode via an interrupt, a further instruction in the main program is executed before the interrupt routine is entered.

If the $\overline{INT}$/T0 level is active (LOW) during the STOP instruction, no STOP is executed.

A LOW level on the external interrupt input of at least 1 µs will cause the microcontroller to exit the STOP mode. During the STOP mode, the address of the instruction immediately following the instruction that caused the microcontroller to enter the STOP mode is present on the address bus.

I/O facilities

Parallel ports

Parallel port lines can be individually configured as outputs or inputs; theri structure is quasi-bidirectional. Output data written to a port is latched and remains unchanged until rewritten. Input data is not latched and must be present until read by an input instruction.

Input lines are fully CMOS compatible; output lines can drive one TTL or CMOS load.

Fig. 8 shows the timing diagram for all ports using IN, OUTL, ANL, and ORL instructions. For the OUTL instruction, data changes on time slot 7 of cycle 1. For the MOV, ANL and ORL instructions, the ports change on time slot 7 of cycle 2.

FUNCTIONAL DESCRIPTION (continued)

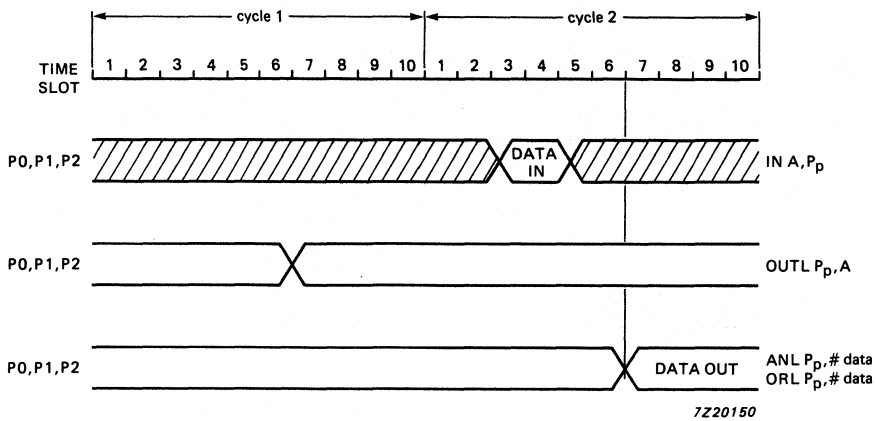


Fig. 8 Timing diagram for all ports using IN, OUTL, ANL, and ORL instructions.

Fig. 9 shows the quasi-bidirectional I/O interface with push-pull output and switched pull-up current source.

Each line is pulled up to V_{DD} via a constant current source (TR4), which is enabled via TR3 whenever one of the two output latches contains a logic 1. This current source is sufficient for a TTL HIGH level, yet can be pulled LOW by an external CMOS device, thus allowing the same pin to be used for both input and output.

When a logic 1 is written to a port line for the first time (MQ = 1, SQ = 0), TR2 is switched on for the duration of the internal write pulse (one oscillator period), to provide a fast transition from logic 0 to logic 1. Subsequent writing of a logic 1 to a port line will not switch TR2 on. This ensures that the port lines can be pulled LOW when configured as inputs.

When a logic 0 is written to a port line, TR3 switches off the current source. Current sinking capability is provided by TR1, which is now switched on. When used as an input, a logic 1 must first be written to the port line; otherwise TR1 will remain low impedance.

PCF84CXXX mask options make it possible for each individual port line to be configured as one of the following:

1. STANDARD PORT: quasi-bidirectional I/O with switched pull-up current source of 100 μA (typ.) and P-channel booster transistor TR2. TR2 is only active during 1 clock cycle (Fig. 9).
2. OPEN DRAIN: Quasi-bidirectional I/O with only an N-channel open drain output. Application as an output requires connection of an external pull-up resistor (Fig. 10). When an open drain port is unused it must be connected to V_{SS}.
3. PUSH-PULL OUTPUT: the outputs can sink or source 1,6 mA (min.) at V_{DD} = 5 V. Thse pins may not be used as inputs (Fig. 11). The contents of the port latch may be read using the IN A,pp instruction.

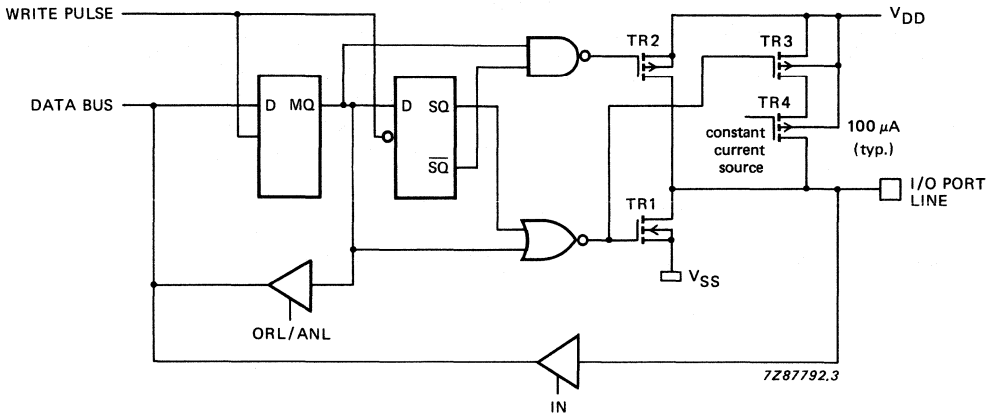


Fig. 9 Standard output with switched pull-up current source.

DEVELOPMENT DATA

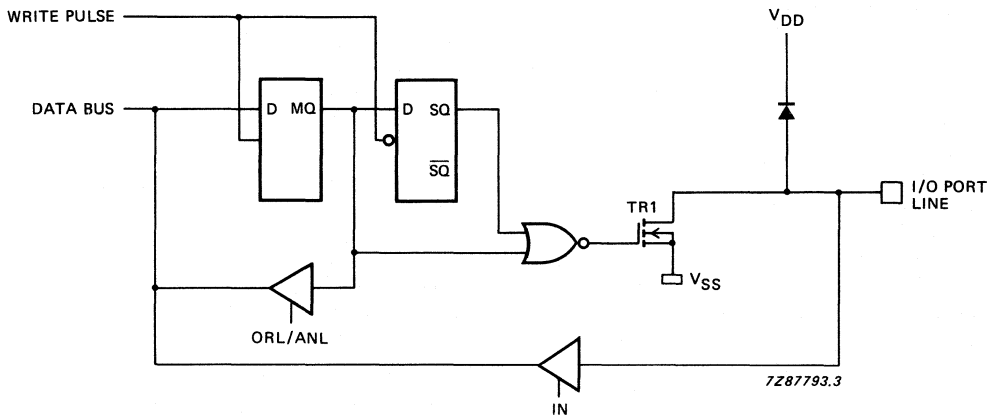


Fig. 10 Open drain output.

FUNCTIONAL DESCRIPTION (continued)

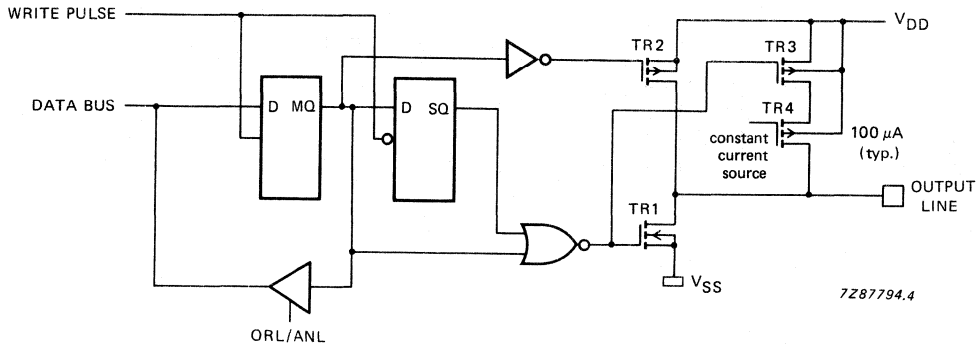


Fig. 11 Push-pull output.

Serial I/O (SIO)

The PCF84CXXX has an on-chip serial I/O interface that supports the I^2C-bus. Whereas a normal microcontroller must regularly monitor the serial data bus for the presence of data, the serial I/O interface detects, receives and converts the serial data stream into parallel format without interrupting the execution of the current program. An interrupt is sent to the PCF84CXXX only when a complete byte is received. It then reads the data byte in one instruction.

The design of the PCF84CXXX serial I/O system allows any number of devices from the PCF8500 family (clips) to be connected via the two-line serial bus. The ability of any devices to communicate, without interrupting the operation of any other devices on the bus, is an outstanding attribute of the system. This is achieved by allocating a specific 7-bit address to each device and providing a system whereby a device reacts only to a message prefixed with its own address or the 'general CALL' address. Address recognition is performed by the interface hardware so that operation of the microcontroller need only be interrupted when a valid address has been received. This saves significant processing time and memory space compared with a conventional microcontroller employing a software serial interface. When the addressing facility is not required, for instance in a system with only two microcontrollers, direct data transfer without addressing can be performed. In multi-master systems, an automatically invoked arbitration procedure prevents two or more devices from continuing simultaneous transmission.

In NORMAL (running) and IDLE mode, the serial I/O logic remains active; its internal system clock will be switched off when there is no activity on the serial bus.

After execution of the STOP instruction, the oscillator of the PCF84CXXX is switched off. This means that the serial I/O logic will remain in the state it was in when the STOP mode was entered. To avoid "bus block" problems and to assure correct start-up of the bus after exit from the STOP mode, the user should disable the serial logic (ESO = 0) prior to the execution of the STOP instruction.

Serial I/O interface

Figure 12 shows the serial I/O interface. The clock line of the serial bus has exclusive use of the SCLK pin while the data line shares the SERIAL DATA pin with I/O line P2.3. When the serial I/O is enabled, P2.3 is disabled as a parallel port line; (P2.3 and SCLK are open drain; when unused these lines must be connected to V_{SS}).

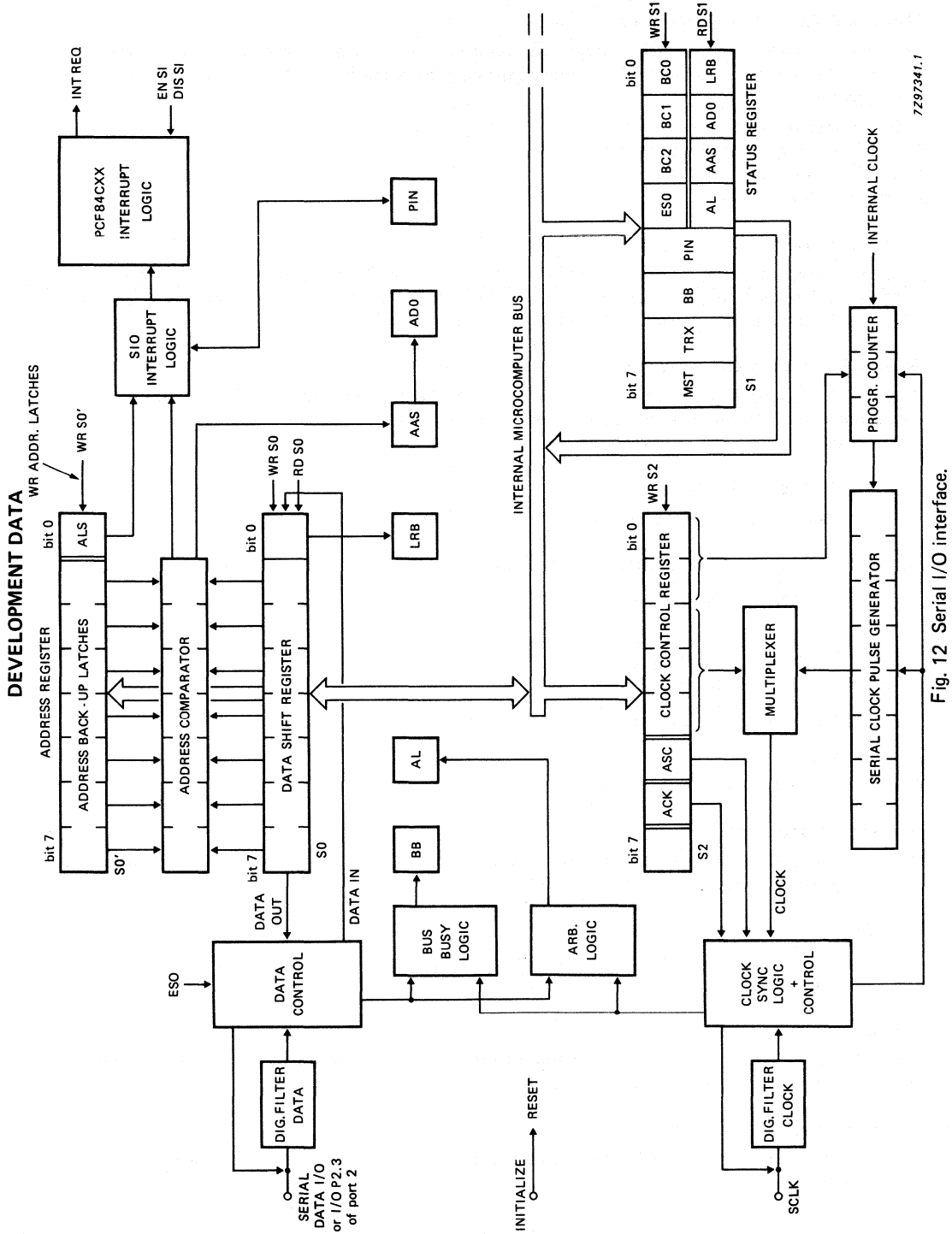


Fig. 12 Serial I/O interface.

FUNCTIONAL DESCRIPTION (continued)

The microcontroller and interface communicate via the internal microcontroller bus and the Serial Interrupt Request line. Data and information controlling the operation of the interface are stored in four registers:

- Data shift register (S0)
- Serial I/O interface status word (S1)
- Serial clock control word (S2)
- Address register (SO')

Data shift register (S0)

Register S0 converts serial data to parallel format and vice versa. An interrupt is generated only after a complete byte has been transmitted, or after a complete data byte, specific address or 'general CALL' address has been received. The most significant bit is transmitted first.

Serial I/O interface status word (S1)

Register S1 provides information concerning the state of the interface and stores information from the microcontroller. Bits 0 to 3 are duplicated: control bits in these positions can only be written by the microcontroller, while status bits can only be read.

MST and TRX (see Table 1)

These bits determine the operating mode of the serial I/O interface.

Table 1 Operating modes of the serial I/O interface

MST	TRX	operating mode
0	0	slave receiver
1	0	master receiver
0	1	slave transmitter
1	1	master transmitter

BB: Bus Busy.

This flag indicates the status of the bus.

PIN: Pending Interrupt NOT

PIN = '0' indicates the presence of a pending interrupt, which will cause a Serial Interrupt Request when the serial interrupt mechanism is enabled.

ESO: Enable Serial output

The ESO flag enables/disables the serial I/O interface: ESO = '1' enables, ESO = '0' disables. ESO can only be written by software.

BC0, BC1 and BC2

Bits BC0, BC1 and BC2 indicate the number of bits received or transmitted in a data stream. These bits can only be written by software.

AL: Arbitration Lost

The arbitration lost flag is set by hardware when the serial I/O interface, as master transmitter, loses a bus arbitration procedure.

AAS: Addressed As Slave

This flag is set by hardware when the interface detects either its own specific address or the 'general CALL' address as the first byte of a transfer and the interface has been programmed to operate in the address recognition mode.

AD0: Address Zero

This flag os set by hardware after detection of the 'general CALL' address when the interface is operating in the address recognition mode.

LRB: Last Received Bit

This contains either the last data bit received or, for a transmitting device in the acknowledgement mode, the acknowledgement signal from the receiving device.

Bits AL, AAS, AD0 and LRB can only be read by software.

Serial clock control word (S2)

Bits 0 to 4 of the clock register S2 determine the frequency of the serial clock signal. When a 6 MHz crystal is used, the frequency of the serial clock can be varied between 1 kHz and 154 kHz (see Table 2). An asymmetrical clock with a HIGH-to-LOW ratio of 3 : 1 can be generated using bit 5. The asymmetrical clock allows a microcontroller more time per clock period for sampling the data line, making the timing of this action less critical. Bit 6 can be used to activate the acknowledge mode of the serial I/O. S2 is a write only register.

Address register (S0)

The address register contains the 7-bit address back-up latches and the bit (ALS) used to enable/disable the address recognition mode. The address register can be written using the MOV S0, A and MOV S0, # data instructions, but only when ESO = logic 0.

Serial I/O interrupt logic

An EN SI instruction enables and a DIS SI instruction disables the interrupt logic. When enabled, a pending interrupt results in a serial I/O interrupt to the processor, causing a CALL to location 5 in the ROM. When disabled, the presence of an interrupt request is still indicated by PIN in S1, but the interrupt will not be serviced.

FUNCTIONAL DESCRIPTION (continued)

Table 2 SIO clock pulse frequency control when using a 6 MHz and a 10 MHz crystal

hexadecimal S20-S24 code	f_{OSC} divided by	f_{OSC} = 6 MHz f_{SCLK} (kHz)**	f_{OSC} = 10 MHz f_{SCLK} (kHz)**
0	not allowed		
1	39	*154	*256
2	45	*133	*222
3	51	*118	*196
4	63	95	*159
5	75	80	*133
6	87	69	*115
7	99	61	*101
8	123	49	81
9	147	41	68
A	171	35	58
B	195	31	51
C	243	25	41
D	291	21	34
E	339	18	29
F	387	16	26
10	483	12	21
11	579	10	17
12	675	8,9	15
13	771	7,8	13,4
14	963	6,2	10,4
15	1155	5,2	8,7
16	1347	4,5	7,4
17	1539	3,9	6,5
18	1923	3,1	5,2
19	2307	2,6	4,3
1A	2691	2,2	3,7
1B	3075	2,0	3,3
1C	3843	1,6	2,6
1D	4611	1,3	2,2
1E	5379	1,1	1,9
1F	6147	1,0	1,6

* Not permitted for I^2C operation.

** The maximum clock frequency in the I^2C systems is 100 kHz.

Interrupts

When an interrupt routine is entered, the contents of the program counter and bits 4, 6 and 7 of the PSW are saved in the program counter stack. The contents of the accumulator can only be saved by user software. Interrupt acknowledgement may be carried out by software via I/O ports. All interrupt routines must reside in memory bank 0; the SEL MB instructions may not be used within an interrupt routine. An interrupt routine can only be terminated by the RETR (return and restore) instruction. During an interrupt routine, subroutine calls must be terminated by the RET instruction. Using the RETR instruction to terminate a subroutine called in an interrupt routine would terminate the interrupt routine prematurely and result in a wrong return address.

External Interrupts

When the external interrupt is enabled and no interrupt routine is in progress, a HIGH-to-LOW transition on the $\overline{INT}$/T0 pin sets the External Interrupt Flag (EIF) and initiates the external interrupt routine by forcing a CALL to program memory location 3.* The program counter points to the external interrupt vector address (003 H) between 2,6 and 3,6 machine cycles after the transition occurs. Interrupt latency depends on the instruction that is being executed when the transition occurs. If an interrupt routine is already in progress, an external interrupt request is stored in the External Interrupt Flag (EIF). When the external interrupt has been disabled, the request is still latched into the digital filter. Execution of a DIS I instruction cancels stored interrupt requests by clearing both the digital filter latch and the external interrupt flag.

An additional external interrupt can be created by enabling the timer/counter interrupt and loading FFH into the counter (one less than overflow). If the event counter mode is enabled by executing the STRT CNT instruction, a LOW-to-HIGH transition on the T1 input will then initiate an interrupt routine by forcing a call to the timer/counter interrupt vector (location 7).

SIO Interrupt

An interrupt request from the SIO hardware will set the PIN flag to its active LOW state. This action is independent of the Enable SIO interrupt flag. When the SIO interrupt is enabled, the active LOW PIN flag will invoke the SIO interrupt routine by forcing a CALL to program memory location 5. After the SIO interrupt is initiated , the PIN flag is not automatically reset HIGH. PIN must be cleared by software during the interrupt routine.

Timer/Counter Interrupt

When no interrupt is in progress and the timer/counter is enabled, a timer/counter overflow sets the Timer Interrupt Flag (TIF). This initiates the timer interrupt routine by forcing a CALL to program memory location 7.** If an interrupt routine is in progress, the interrupt request is stored in the timer interrrupt flag only if the timer interrupt has been enabled. Execution of a DIS TCNTI instruction cancels a previously stored interrupt request. The timer flag (TF) is set every time the timer/counter overflows and is not automatically reset when the timer/counter interrupt routine is called. It can only be cleared by the JTF and JNTF instructions or by a hardware RESET.

Interrupt Priority

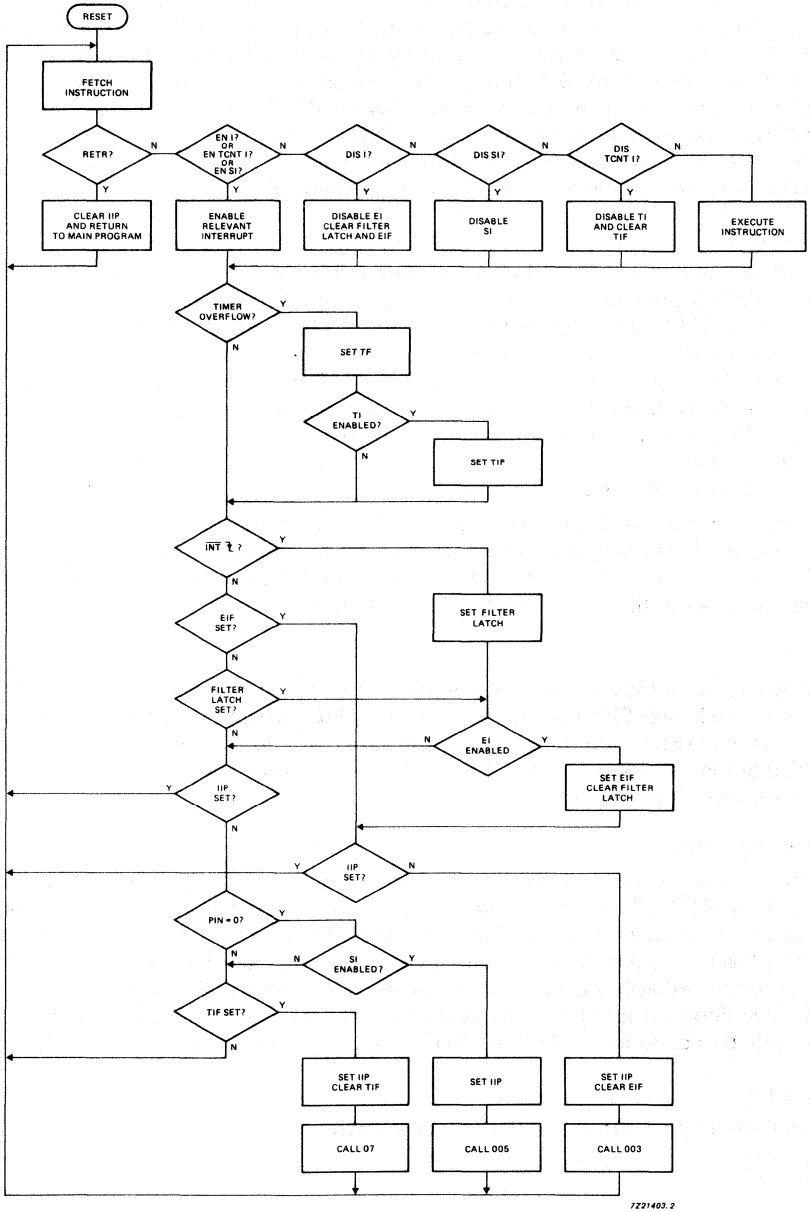
If simultaneous interrupts occur, their priority is as follows:

External (highest)
SIO
Timer/Counter (lowest)

An interrupt routine can only be interrupted by a hardware RESET and cannot be interrupted by other interrupts (which will be latched). When the interrupt routine is terminated by the RETR instruction, at least one instruction of the main program will be executed before another interrupt routine is entered.

* This CALL clears the EIF flag.
** This CALL clears the TIF flag.

FUNCTIONAL DESCRIPTION (continued)



EI	External Interrupt	TF	Timer Flag
SI	Serial Interrupt	TIF	Timer Interrupt Flag
TI	Timer/counter Interrupt	PIN	Pending Interrupt Not (SIO)
EIF	External Interrupt Flag	IIP	Interrupt In Progress Flag

Fig. 13(a) Flow chart illustrating the interrupt handling sequence.

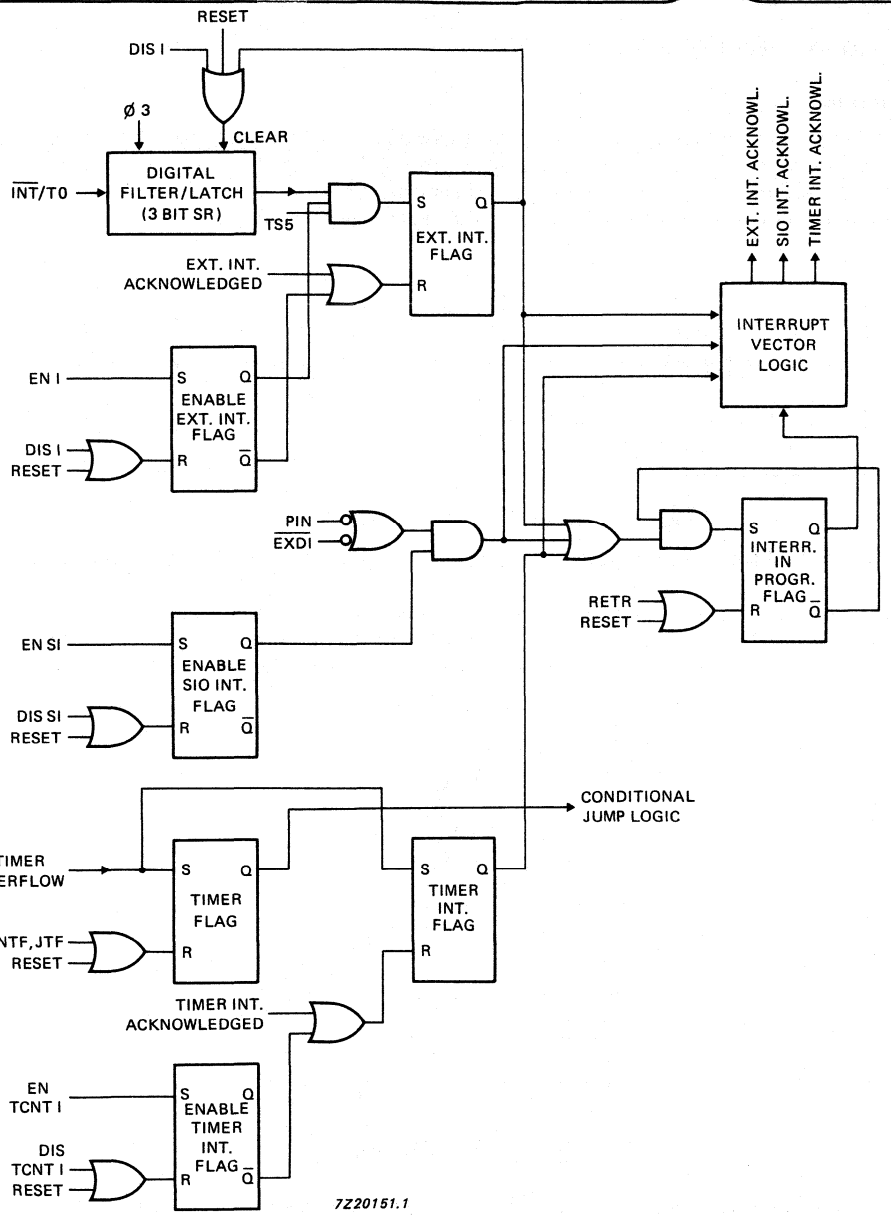


Fig. 13(b) Simplified schematic of the interrupt logic.

Notes to figure 13(b)

1. $\overline{INT}$/T0 negative edge is always latched in the digital filter/latch.
2. Correct interrupt timing is ensured when $\overline{INT}$/T0 is HIGH for > 4 CP and then LOW for > 7 CP.
3. When the interrupt in progress flag is set, further external and timer interrupts are latched but ignored, until RETR is executed.
4. A DIS I instruction always clears a pending external interrupt.
5. For all flip-flops, RESET overrules SET.

FUNCTIONAL DESCRIPTION (continued)

Oscillator (see Fig. 14)

The oscillator can be inhibited by the STOP instruction under software control. It is also inhibited when a low-supply voltage condition is present to prevent discharge of a weak back-up battery. Provided the supply voltage is within the operating range, the oscillator will be restarted after a STOP instruction by a LOW level at the $\overline{INT}$/T0 pin or a HIGH level at the RESET pin.

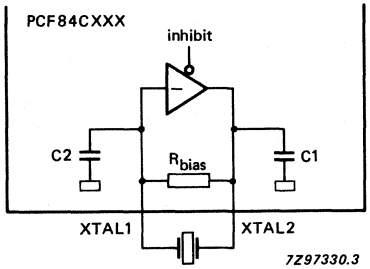


Fig. 14 Oscillator with integrated elements.

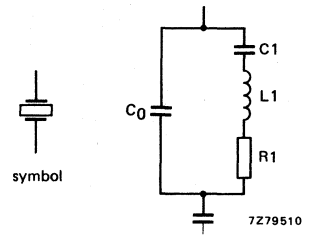


Fig. 15 Crystal unit equivalent circuit.

The values of crystal series resistance R1 and the crystal's total load capacitance C_L (C_0 + wiring + external capacitors) must not be above the curve (Fig. 16) for the corresponding frequency.
Note: if external capacitors are connected to XTAL1 and XTAL2, they must be of equal value.

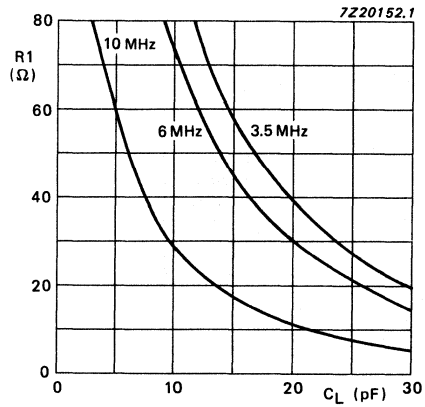


Fig. 16 Crystal circuit criteria.

XTAL2 is the output of the inverting amplifier. An external clock can be applied to XTAL1.
A machine cycle consists of 10 time slots; each time slot is 3 oscillator periods.

Timer/event counter (see Fig. 17)

An internal 8-bit up-counter is provided. This can count external events, modulo-32 machine cycles, or machine cycles directly. Table 3 shows the instructions that control the counter and the prescaler, and the functions performed.

When used as a timer, the input to the counter is either the overflow or input of a 5-bit prescaler. When used as an event counter, LOW-to-HIGH transitions on pin T1 are counted. The counter is incremented during a machine cycle only if the falling edge occurs during the first 7 time slots; otherwise it is incremented during the next cycle. The maximum rate at which the counter may be incremented is once every machine cycle. When the counter overflows, the Timer flag is set. The flag can be tested and reset using the JTF (jump if Timer flag = logic 1) or JNTF (jump if Timer flag = logic 0) instruction. Overflow also generates an interrupt request to the microcontroller by setting the Timer Interrupt Flag when the timer/event counter interrupt is enabled.

Table 3 Timer/event counter control

function	timer mode modulo-1, modulo-32*	counter mode
CLEAR	MOV T,A (A) = 0 or RESET	MOV T,A (A) = 0 or RESET
PRESET	MOV T,A	MOV T,A
START	STRT T	STRT CNT
STOP	STOP TCNT or RESET	STOP TCNT or RESET
TEST	JTF/JNTF	JTF/JNTF
READ**	MOV A,T	MOV A,T

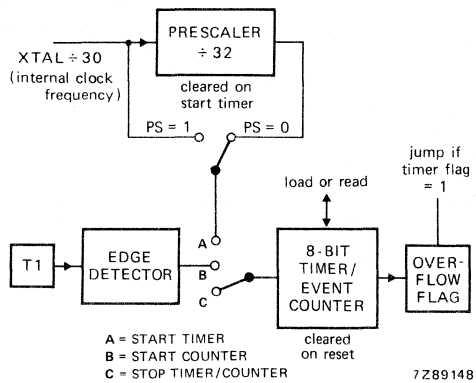


Fig. 17 Timer/event counter.

* With prescaler select (PS) = logic 0, the timer is incremented every 32 machine cycles; with PS = logic 1 the timer is incremented every machine cycle (prescaler not used), the prescaler is cleared by the STRT T instruction and is not readable.

** READ does not disturb the counting process.

FUNCTIONAL DESCRIPTION (continued)

Program status word (see Fig. 18)

The program status word (PSW) is an 8-bit word (1 byte) in the CPU which stores information about the current status of the microcontroller.
The PSW bits are:

- Bits 0 to 2 — stack pointer bits (SP_0, SP_1, SP_2)
- Bit 3 — prescaler select (PS);
  0 = modulo-32; 1 = modulo-1 (no prescaling)
- Bit 4 — working register bank select (RBS);
  0 = register bank 0; 1 = register bank 1
- Bit 5 — not used (1)
- Bit 6 — auxiliary carry (AC); half-carry bit generated by an ADD instruction and used by the decimal adjust instruction DA A
- Bit 7 — carry (CY); the carry flag indicates that the previous operation resulted in an overflow of the accumulator

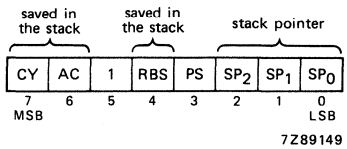


Fig. 18 Program status word.

All bits can be read using the MOV A, PSW instruction. Bits 7 and 6 are set and cleared by CPU operation. Bit 4 can be changed by the SEL RB instructions, bit 3 by the MOV PSW, A instruction, and bits 0, 1 and 2 by the CALL, RET or RETR instructions and in the event of an interrupt. Bits 7, 6 and 4 are stored in the program counter stack during subroutine and interrupt calls. These bits are restored in the PSW with a RETR (return and restore) instruction which must be used at the end of an interrupt service routine and can be used at the end of a normal subroutine. The RET instruction has no restore feature and cannot be used at the end of an interrupt service routine.

Program counter (see Fig. 19)

PCF84CXXX program counters can address up to 8 K bytes of ROM. The arrangement of the bits is shown in Fig. 17. The contents of the program counter are saved in the stack during CALL and interrupt routines.

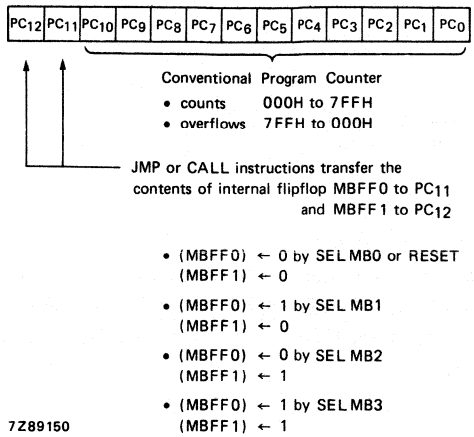


Fig. 19 Program counter.

Central processing unit

The PCF84CXXX has arithmetic, logical and branching capabilities. The DA A, SWAP A and XCHD instructions simplify BCD arithmetic and the handling of nibbles. The MOVP A,@A instruction permits efficient table look-up from the current ROM page.

Conditional branch logic

The conditional branch logic within the microcontroller enables several conditions, internal and external to the microcontroller, to be tested by the user's program. Table 4 lists the conditional jump instructions used to change the program sequence. The DJNZ instruction decrements a designated register or data memory location and branches if the contents are not zero. This instruction is useful for looping control. The JMPP@A instruction allows multiway branches to destinations determined by the contents of the accumulator.

Table 4 Conditional branches

test	jump condition	jump instruction
accumulator	all bits zero	JZ
	any bit non-zero	JNZ
accumulator bit test	1	JB0 to JB7
carry flag	1	JC
	0	JNC
timer overflow flag	1	JTF
	0	JNTF
test input T0	1	JT0
	0	JNT0
test input T1	1	JT1
	0	JNT1
register	non-zero	DJNZ

FUNCTIONAL DESCRIPTION (continued)

Test input T1

The T1 input line can be used as:

- A test input for branch instructions JT1 and JNT1
- An external input to the event counter

When used as a test input:

- JT1 instruction tests for logic 1 level
- JNT1 instruction tests for logic 0 level

When used as an input to the event counter, T1 must be LOW for $>$ 4 CP, and then HIGH for $>$ 4 CP. A transition can be recognized every 30 oscillator clock periods (1 machine cycle).

There is no internal pull-up or pull-down resistor connected to the T1 input. When T1 is not used, it must be tied to V_{DD} or V_{SS}.

Power-on-reset

The internal power-on reset circuit monitors the supply voltage V_{DD}. As long as the supply voltage remains below the internal reference level V_{ref} (typically 1,5 V), the oscillator is inhibited and RESET has an undefined level. When V_{DD} rises above the internal reference level, the oscillator is released and RESET is pulled high to V_{DD} by TR1 for a period t_D (typically 50 μs).

N.B. Because of the narrow bandwidth of the crystal, the start-up time of the oscillator is typically 10 ms.

Three cases of power-on reset are possible:

1. If V_{DD} can be switched with a fast rise time i.e. V_{DD} reaches its minimum operating value (corresponding to the selected oscillator frequency) before the RESET signal has finished (t_D), then no extra components are required (see Figs 20 and 21). Note that the first instruction is executed after the oscillator start-up time plus 1866 clock periods have elapsed.
2. If V_{DD} has a slow rise time then the RESET signal should be stretched by an external RC circuit (see Figs 22 and 23). In the event of a short drop in the supply voltage, the diode path rapidly discharges the capacitor to ensure a reliable power-on-reset. To ensure a correct reset, the RESET signal should reach at least 70% of the final value of V_{DD}. Given that the RESET voltage and V_{DD} rise exponentially, the above requirement is satisfied when the time constant τ of the RESET pulse is $>$ 8 times the time constant of V_{DD}. If V_{DD} rises linearly, then a RESET time constant $>$ 2 times the rise time of V_{DD} is required.

   When a reset is completed (RESET goes LOW) before the oscillator has started up, program execution begins after the oscillator start-up time plus 1866 clock periods have elapsed (see Fig. 23). If the oscillator is started up prior to the completion of RESET, then program execution begins 1866 clock periods after RESET goes LOW.
3. Fig. 22 shows an external reset during power-on. The external reset signal must remain HIGH until V_{DD} has reached its minimum operating value corresponding to the selected oscillator frequency. When a reset is completed (RESET goes LOW) before the oscillator has started up, program execution begins after the oscillator start-up time plus 1866 clock periods have elapsed (see Fig. 25). If the oscillator is started-up prior to the completion of RESET, then program execution begins 1866 clock periods after RESET goes LOW.

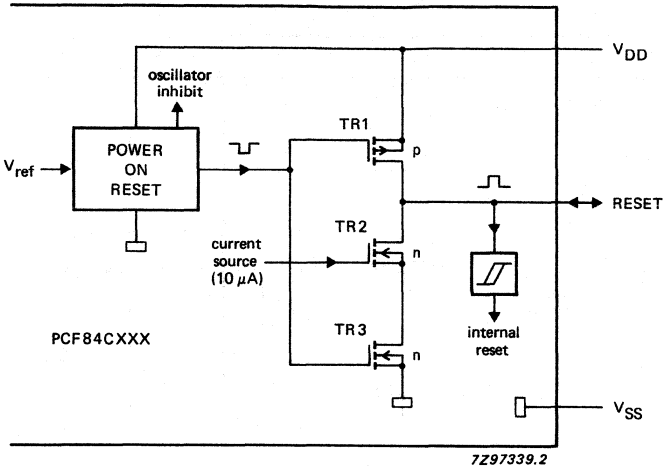


Fig. 20 Power-on-reset configuration.

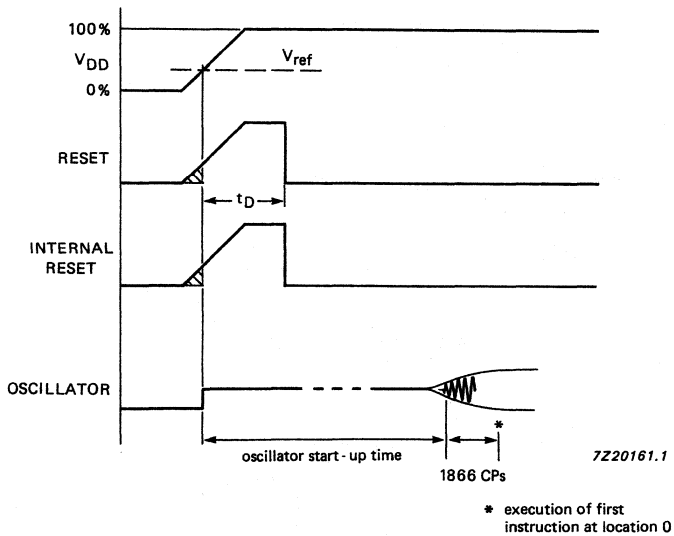


Fig. 21 Timing of power-on-reset with fast rise time.

DEVELOPMENT DATA

FUNCTIONAL DESCRIPTION (continued)

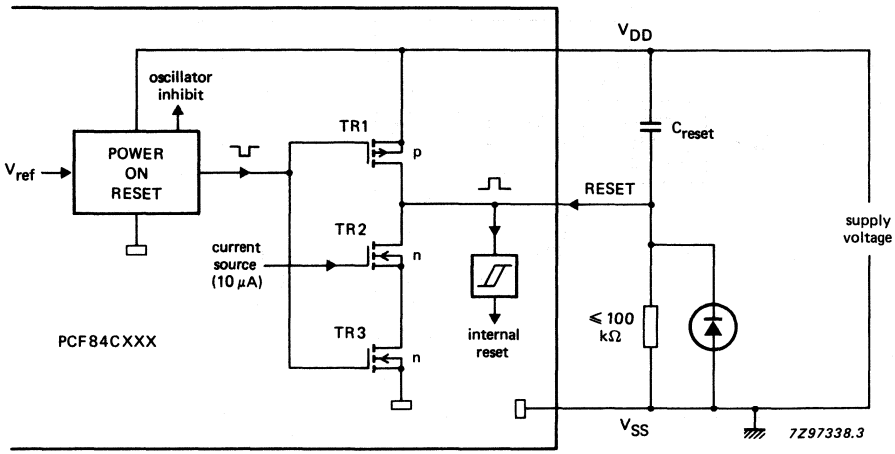


Fig. 22 Stretched power-on-reset with external components.

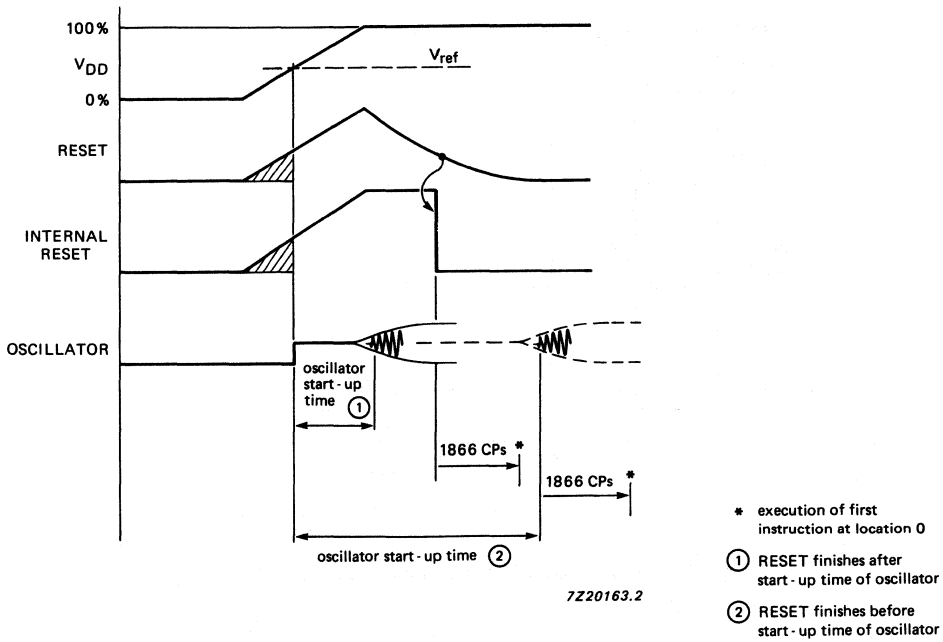


Fig. 23 Timing of power-on-reset with a slowly rising V_{DD} and a stretched RESET pulse.

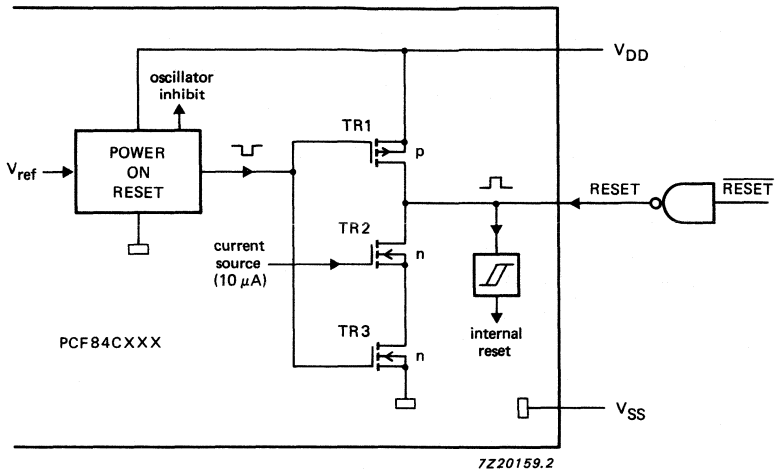


Fig. 24 External power-on-reset configuration.

DEVELOPMENT DATA

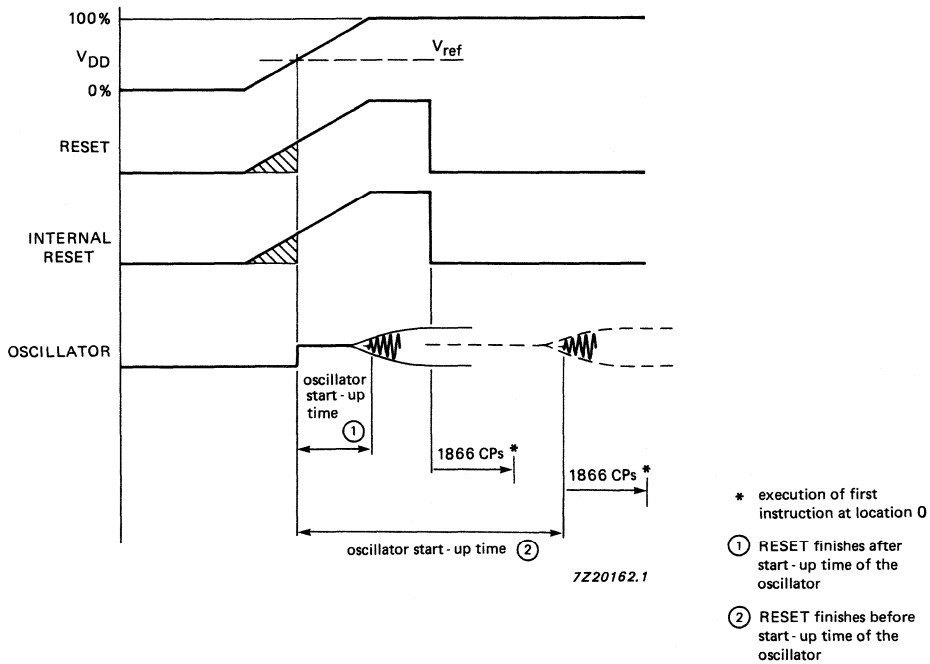


Fig. 25 Timing of external power-on-reset.

SINGLE-CHIP 8-BIT MICROCONTROLLERS

DESCRIPTION

An advanced CMOS process is used to manufacture the PCF84C12, PCF84C22 and PCF84C42 microcontrollers. Each device has 13 quasi-bidirectional I/O port lines, a single-level vectored interrupt structure, an 8-bit timer and on-chip clock oscillator and clock circuits. On-chip RAM and ROM content is as follows:

- PCF84C12 – 64 x 8 RAM, 1 K x 8 ROM
- PCF84C22 – 64 x 8 RAM, 2 K x 8 ROM
- PCF84C42 – 64 x 8 RAM, 4 K x 8 ROM

These efficient microcontrollers also perform well as arithmetic processors. The instruction set is similar to that of the MAB8048. They have bit handling abilities and facilities for both binary and BCD arithmetic.

These microcontrollers are members of the PCF84CXXX family. For detailed information, consult the PCF84CXXX data sheet.

Features

- 8-bit CPU, ROM, RAM, I/O in a single 20-lead DIL or SO package
- 1 K, 2 K or 4 K x 8 ROM
- 64 x 8 RAM
- 2 timers (8-bit programmable)
- 13 quasi-bidirectional I/O port lines
- Two test inputs: one of which is also the external interrupt input
- Single-level, vectored interrupts: external and timer/event counter
- 8-bit programmable timer/event counter
- Clock frequency range: 100 kHz to 10 MHz
- Over 80 instructions (similar to those of the MAB8048) all of 1 or 2 cycles
- Single supply voltage (2.5 V to 5.5 V)
- STOP and IDLE modes
- Power-on-reset circuit
- Operating temperature range: −40 to + 85 °C

Contents	Page	Contents	Page
Block diagram	2	Program counter	20 *
Pinning	3	Central processing unit	21 *
Program memory	4,3 *	Conditional branch logic	21 *
Data memory	4,3 *	Test input T1	22 *
Program counter stack	4 *	Reset	4
IDLE and STOP modes	5 *	Power-on-reset	22 *
I/O facilities	4,7 *	Instruction set	5
Interrupts	15 *	Ratings	12
Oscillator	18 *	DC characteristics	13
Timer/event counters	19 *	Package and soldering details	16
Program status word	20 *		

* See PCF84CXXX Family Data Sheet.

PACKAGE OUTLINES

PCF84C12/22/42P: 20-lead DIL; plastic (SOT146).
PCF84C12/22/42T: 20-lead mini-pack; plastic (SO20, SOT163A).

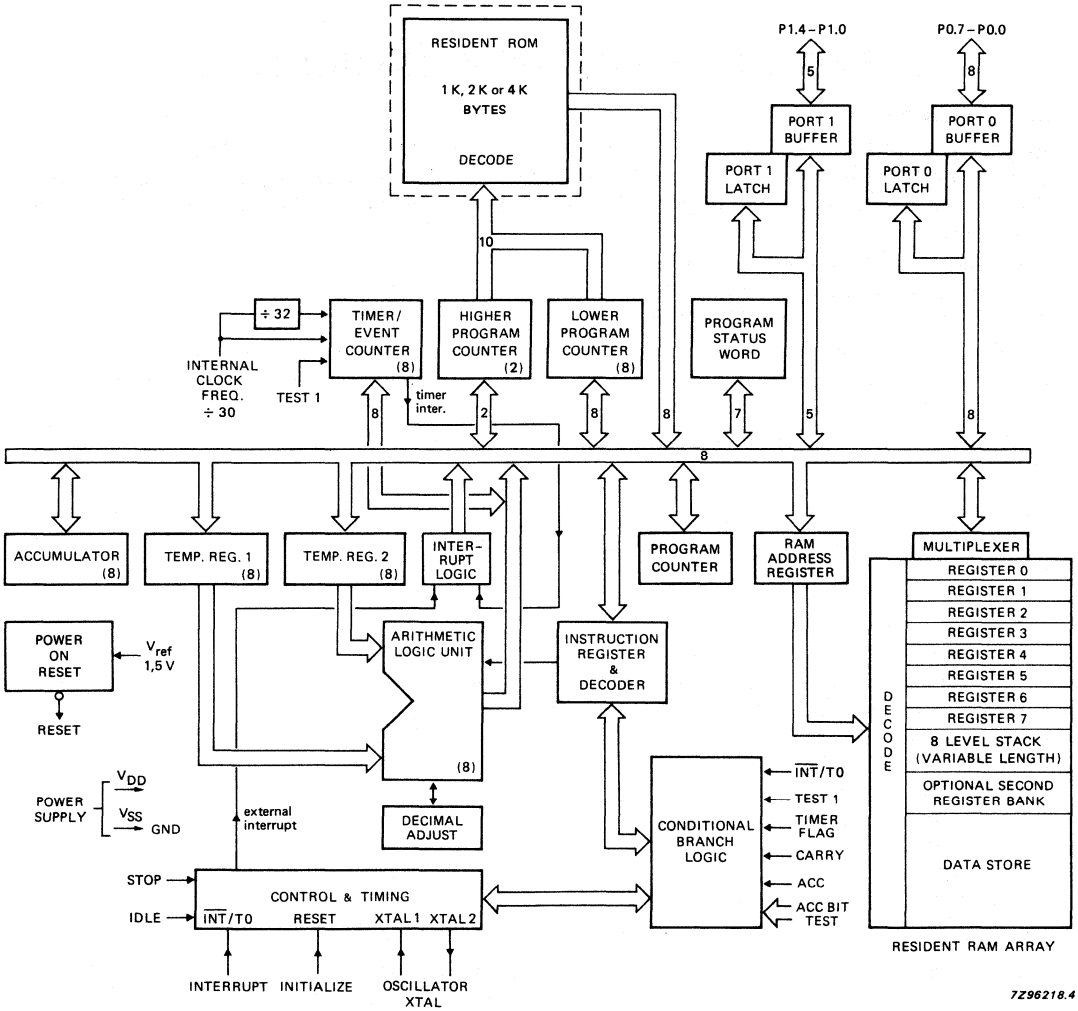


Fig. 1 Block diagram.

PINNING

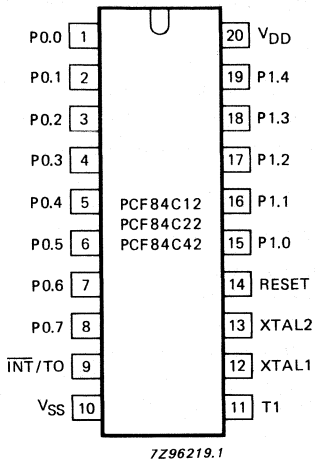


Fig. 2 Pinning diagram.

DEVELOPMENT DATA

PIN DESIGNATION

1-8	P0.0-P0.7	**Port 0:** 8-bit quasi-bidirectional I/O port.
9	$\overline{INT}$/T0	**Interrupt/Test 0**: external interrupt input (negative edge triggered)/test input pin; when used as a test input, this pin is directly tested by conditional branch instructions JT0 and JNT0.
10	V_{SS}	**Ground:** circuit earth potential.
11	T1	**Test 1:** test input pin, directly tested by conditional branch instructions JT1 and JNT1. T1 may also be selected as an input to the 8-bit timer/event counter via the STRT CNT instruction.
12	XTAL 1	**Oscillator input:** input from a crystal which determines the internal oscillator frequency, or an external clock generator.
13	XTAL 2	**Oscillator output:** output of the inverting amplifier.
14	RESET	**Reset input:** used to initialize the microcontroller (active HIGH); also output of power-on-reset circuit.
15-19	P1.0-P1.4	**Port 1:** 8-bit quasi-bidirectional I/O port.
20	V_{DD}	**Power supply:** 2.5 V to 5.5 V.

FUNCTIONAL DESCRIPTION

Program memory

The program memory consists of 1 K, 2 K or 4 K bytes of read-only memory (ROM). Each location is directly addressable by the program counter. The ROM is mask-programmed at the factory.

Data memory

Data memory consists of 64 bytes, random-access data memory (RAM). All locations are indirectly addressable using RAM pointer registers; up to 16 designated locations are directly addressable. Memory also includes an 8-level program counter stack addressed by a 3-bit stack pointer.

I/O facilities

Each device has 13 I/O lines arranged as follows:

- Port 0 — 8-bit parallel port (P0.0 to P0.7)
- Port 1 — 5-bit parallel port (P1.0 to P1.4)
- $\overline{\text{INT}}$/T0 — external interrupt and test input. When used as a test input can be directly tested by conditional branch instructions JT0 and JNT0
- T1 — test input which can alter program sequences when tested by conditional jump instructions JT1 and JNT1. T1 also functions as an input to the 8-bit timer/event counter.

Reset (pin 14)

A positive-going signal on the RESET input:

- Sets the program counter to zero
- Selects location 0 of memory bank 0 and register bank 0
- Sets the stack pointer to zero (000); pointing to RAM address 8
- Disables the interrupts (external and timer)
- Stops the timer/event counter, then sets it to zero
- Sets the timer prescaler to divide by 32
- Resets the timer flag
- Sets all ports to input mode
- Cancels IDLE and STOP mode

INSTRUCTION SET

The instruction set consists of over 80 one and two byte instructions. Program code efficiency is high because all RAM locations and all ROM locations on a 256 byte page require only a single byte address.

Table 3 gives the instruction set. Table 2 shows the instruction map and Table 1 details the symbols that are used.

Table 1 Symbols used in Table 3

symbol	description
A	accumulator
addr	program memory address
Bb	bit designation (b = 0-7)
RBS	register bank select
C	carry bit (bit CY)
CNT	event counter
D	mnemonic for 4-bit digit (nibble)
data	8-bit number or expression
I	interrupt
P	mnemonic for 'in-page' operation
PC	program counter
Pp	port designation (p = 0 or 1)
PSW	program status word
RB	register bank
Rr	register designation (r = 0-7)
SP	stack pointer
T	timer
TF	timer flag
T0, T1	test 0 and 1 inputs
#	immediate data prefix
@	indirect address prefix
(X)	contents of X
((X))	contents of location addressed by X
←	is replaced by
↔	is exchanged with

INSTRUCTION SET (continued)

Table 2 Instruction map

first hexadecimal character of opcode ↓

second hexadecimal character of opcode →

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	NOP	IDLE		ADD A,#data	JMP page 0	EN I	JNTF addr	DEC A	IN A,Pp							
1	INC @Rr		JBO addr	ADDC A,#data	CALL page 0	DIS I	JTF addr	INC A	INC Rr							
2	XCH A,@Rr		STOP	MOV A,#data	JMP page 1	EN TCNTI	JNTO addr	CLR A	XCH A,Rr							
3	XCHD A,@Rr		JB1 addr		CALL page 1	DIS TCNTI	JTO addr	CPL A	OUTL Pp,A							
4	ORL A,@Rr		MOV A,T	ORL A,#data	JMP page 2	STRT CNT	JNT1 addr	SWAP A	ORL A,Rr							
5	ANL A,@Rr		JB2 addr	ANL A,#data	CALL page 2	STRT T	JT1 addr	DA A	ANL A,Rr							
6	ADD A,@Rr		MOV T,A		JMP page 3	STOP TCNT		RRC A	ADD A,Rr							
7	ADDC A,@Rr		JB3 addr		CALL page 3			RR A	ADDC A,Rr							
8				RET	JMP page 4				ORL Pp,#data							
9			JB4 addr	RETR	CALL page 4		JNZ addr	CLR C	ANL Pp,#data							
A	MOV @Rr,A			MOVP A,@A	JMP page 5			CPL C	MOV Rr,A							
B	MOV @Rr,#data		JB5 addr	JMPP @A	CALL page 5				MOV Rr,#data							
C	DEC @Rr				JMP page 6	SEL RB0	JZ addr	MOV A,PSW	DEC Rr							
D	XRL A,@Rr		JB6 addr	XRL A,#data	CALL page 6	SEL RB1		MOV PSW,A	XRL A,Rr							
E	DJNZ @Rr,addr				JMP page 7		JNC addr	RL A	DJNZ Rr, addr							
F	MOV A,@Rr		JB7 addr		CALL page 7		JC addr	RLC A	MOV A,Rr							

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0									0	1						
1	0	1							0	1	2	3	4	5	6	7
2	0	1							0	1	2	3	4	5	6	7
3									0	1						
4	0	1							0	1	2	3	4	5	6	7
5	0	1							0	1	2	3	4	5	6	7
6	0	1							0	1	2	3	4	5	6	7
7	0	1							0	1	2	3	4	5	6	7
8									0	1						
9									0	1						
A	0	1							0	1	2	3	4	5	6	7
B	0	1							0	1	2	3	4	5	6	7
C	0	1							0	1	2	3	4	5	6	7
D	0	1							0	1	2	3	4	5	6	7
E	0	1							0	1	2	3	4	5	6	7
F	0	1							0	1	2	3	4	5	6	7

Table 3 Instruction set

	mnemonic	opcode (hex.)	bytes/ cycles	description	function		notes
ACCUMULATOR	ADD A, Rr	6*	1/1	Add register contents to A	(A)←(A) + (Rr)	r = 0–7	1
	ADD A, @Rr	60 61	1/1	Add RAM data, addressed by Rr, to A	(A)←(A) + ((R0)) (A)←(A) + ((R1))		1
	ADD A, #data	03 data	2/2	Add immediate data to A	(A)←(A) + data		1
	ADDC A, Rr	7*	1/1	Add carry and register contents to A	(A)←(A) + (Rr) + (C)	r = 0–7	1
	ADDC A, @Rr	70 71	1/1	Add carry and RAM data, addressed by Rr, to A	(A)←(A) + ((R0)) + (C) (A)←(A) + ((R1)) + (C)		1
	ADDC A, #data	13 data	2/2	Add carry and immediate data to A	(A)←(A) + data + (C)		1
	ANL A, Rr	5*	1/1	'AND' Rr with A	(A)←(A) AND (Rr)	r = 0–7	
	ANL A, @Rr	50 51	1/1	'AND' RAM data, addressed by Rr, with A	(A)←(A) AND ((R0)) (A)←(A) AND ((R1))		
	ANL A, #data	53 data	2/2	'AND' immediate data with A	(A)←(A) AND data		
	ORL A, Rr	4*	1/1	'OR' Rr with A	(A)←(A) OR (Rr)	r = 0–7	
	ORL A, @Rr	40 41	1/1	'OR' RAM data, addressed by Rr, with A	(A)←(A) OR ((R0)) (A)←(A) OR ((R1))		
	ORL A, #data	43 data	2/2	'OR' immediate data with A	(A)←(A) OR data		
	XRL A, Rr	D*	1/1	'XOR' Rr with A	(A)←(A) XOR (Rr)	r = 0–7	
	XRL A, @Rr	D0 D1	1/1	'XOR' RAM, addressed by Rr, with A	(A)←(A) XOR ((R0)) (A)←(A) XOR ((R1))		
	XRL A, #data	D3 data	2/2	'XOR' immediate data with A	(A)←(A) XOR data		
	INC A	17	1/1	increment A by 1	(A)←(A) + 1		
	DEC A	07	1/1	decrement A by 1	(A)←(A) −1		
	CLR A	27	1/1	clear A to zero	(A)←0		
	CPL A	37	1/1	one's complement A	(A)←NOT(A)		
	RL A	E7	1/1	rotate A left	$(A_{n+1})\leftarrow(A_n)$ $(A_0)\leftarrow(A_7)$	n = 0–6	

INSTRUCTION SET (continued)

ACCUMULATOR (cont.)	RLC A	F7	1/1	rotate A left through carry	$(A_{n+1}) \leftarrow A_n$ $(A_0) \leftarrow (C), (C) \leftarrow (A_7)$	n = 0–6	2
	RR A	77	1/1	rotate A right	$(A_n) \leftarrow (A_{n+1})$ $(A_7) \leftarrow (A_0)$	n = 0–6	
	RRC A	67	1/1	rotate A right through carry	$(A_n) \leftarrow (A_{n+1})$ $(A_7) \leftarrow (C), (C) \leftarrow (A_0)$	n = 0–6	2
	DA A	57	1/1	decimal adjust A			2
	SWAP A	47	1/1	swap nibbles of A	$(A_{4-7}) \leftrightarrow (A_{0-3})$		
DATA MOVES	MOV A, Rr	F*	1/1	move register contents to A	(A)←(Rr)	r = 0–7	
	MOV A, @Rr	F0 F1	1/1	move RAM data, addressed by Rr, to A	(A)←((R0)) (A)←((R1))		
	MOV A, #data	23 data	2/2	move immediate data to A	(A)←data		
	MOV Rr, A	A*	1/1	move accumulator contents to register	(Rr)←(A)	r = 0–7	
	MOV @Rr, A	A0 A1	1/1	move accumulator contents to RAM location addressed by Rr	((R0))←(A) ((R1))←(A)		
	MOV Rr, #data	B* data	2/2	move immediate data to Rr	(Rr)←data		
	MOV @Rr, #data	B0 data B1 data	2/2	move immediate data to RAM location addressed by Rr	((R0))←data ((R1))←data		
	XCH A, Rr	2*	1/1	exchange accumulator contents with Rr	(A)↔(Rr)	r = 0–7	
	XCH A, @Rr	20 21	1/1	exchange accumulator contents with RAM data addressed by Rr	(A)↔((R0)) (A)↔((R1))		
	XCHD A, @Rr	30 31	1/1	exchange lower nibbles of A and RAM data addressed by Rr	$(A_{0-3}) \leftrightarrow ((R0_{0-3}))$ $(A_{0-3}) \leftrightarrow ((R1_{0-3}))$		
	MOV A, PSW	C7	1/1	move PSW contents to accumulator	(A)←(PSW)		
	MOV PSW, A	D7	1/1	move accumulator bit 3 to PSW_3	$(PSW_3) \leftarrow (A_3)$		3
	MOVP A, @A	A3	1/2	move indirectly addressed data in current page to A	$(PC_{0-7}) \leftarrow (A), (A) \leftarrow ((PC))$		
FLAGS	CLR C	97	1/1	clear carry bit	(C)←0		2
	CPL C	A7	1/1	complement carry bit	(C)←NOT(C)		2

DEVELOPMENT DATA

	mnemonic	opcode (hex.)	bytes/ cycles	description	function	notes
REGISTER	INC Rr	1*	1/1	increment register by 1	(Rr)←(Rr) + 1 r = 0–7	
	INC @Rr	10 11	1/1	increment RAM data, addressed by Rr, by 1	((R0))←((R0)) + 1 ((R1))←((R1)) + 1	
	DEC Rr	C*	1/1	decrement register by 1	(Rr)←(Rr) – 1 r = 0–7	
	DEC @Rr	C0 C1	1/1	decrement RAM data, addressed by Rr, by 1	((R0))←((R0)) – 1 ((R1))←((R1)) – 1	
BRANCH	JMP addr	• 4 address	2/2	unconditional jump within a 2 K bank	(PC_{8-10})←$addr_{8-10}$ (PC_{0-7})←$addr_{0-7}$	
	JMPP @A	B3	1/2	indirect jump within a page	(PC_{0-7})←((A))	
	DJNZ Rr, addr	E* address	2/2	decrement Rr by 1 and jump if not zero to addr	(Rr)←(Rr) – 1 r = 0–7 if (Rr) not zero (PC_{0-7})←addr	
	DJNZ @Rr, addr	E0	2/2	decrement RAM data, addressed by Rr by 1 and jump if not zero to addr	((R0))←((R0)) – 1 if ((R0)) not zero (PC_{0-7})←addr	
		E1			((R1))←((R1)) – 1 if ((R1)) not zero (PC_{0-7})←addr	
	JBb addr	▲ 2 address	2/2	jump to addr if Acc. bit b = 1	if b = 1 : (PC_{0-7})←addr b = 0–7	
	JC addr	F6 address	2/2	jump to addr if C = 1	if C = 1 : (PC_{0-7})←addr	
	JNC addr	E6 address	2/2	jump to addr if C = 0	if C = 0 : (PC_{0-7})←addr	
	JZ addr	C6 address	2/2	jump to addr if A = 0	if A = 0 : (PC_{0-7})←addr	
	JNZ addr	96 address	2/2	jump to addr if A is NOT zero	if A≠0 : (PC_{0-7})←addr	
	JT0 addr	36 address	2/2	jump to addr if T0 = 1	if T0 = 1: (PC_{0-7})←addr	
	JNT0 addr	26 address	2/2	jump to addr if T0 = 0	if T0 = 0: (PC_{0-7})←addr	
	JT1 addr	56 address	2/2	jump to addr if T1 = 1	if T1 = 1: (PC_{0-7})←addr	
	JNT1 addr	46 address	2/2	jump to addr if T1 = 0	if T1 = 0: (PC_{0-7})←addr	
	JTF addr	16 address	2/2	jump to addr if Timer Flag = 1	if TF = 1: (PC_{0-7})←addr	4
	JNTF addr	06 address	2/2	jump to addr if Timer Flag = 0	if TF = 0: (PC_{0-7})←addr	

INSTRUCTION SET (continued)

TIMER/EVENT COUNTER	MOV A, T	42	1/1	move timer/event counter contents to accumulator	(A)←(T)	
	MOV T, A	62	1/1	move accumulator contents to timer/event counter	(T)←(A)	
	STRT CNT	45	1/1	start event counter		
	STRT T	55	1/1	start timer		
	STOP TCNT	65	1/1	stop timer/event counter		
	EN TCNTI	25	1/1	enable timer/event counter interrupt		
	DIS TCNTI	35	1/1	disable timer/event counter interrupt		
CONTROL	EN I	05	1/1	enable external interrupt		
	DIS I	15	1/1	disable external interrupt		
	SEL RB0	C5	1/1	select register bank 0	(RBS)←0	5
	SEL RB1	D5	1/1	select register bank 1	(RBS)←1	5
	STOP	22	1/1	enter STOP mode		
	IDLE	01	1/1	enter IDLE mode		
SUBROUTINE	CALL addr	▲ 4 address	2/2	jump to subroutine	((SP))←(PC), ($PSW_{4, 6, 7}$) 6 (SP)←(SP) + 1 (PC_{8-10})←$addr_{8-10}$ (PC_{0-7})←$addr_{0-7}$	6
	RET	83	1/2	return from subroutine	(SP)←(SP) − 1 (PC)←((SP))	6
	RETR	93	1/2	return from interrupt and restore bits 4, 6, 7 of PSW	(SP)←(SP) − 1 ($PSW_{4, 6, 7}$) + (PC)←((SP))	6

	mnemonic	opcode (hex.)	bytes/ cycles	description	function	notes
PARALLEL INPUT/OUTPUT	IN A, Pp	08 09	1/2	input port p data to accumulator	(A)←(P0) (A)←(P1)	
	OUTL Pp, A	38 39	1/2	output accumulator data to port p	(P0)←(A) (P1)←(A)	
	ANL Pp, #data	98 99	2/2	AND port p data with immediate data	(P0)←(P0) AND data (P1)←(P1) AND data	
	ORL Pp, #data	88 89	2/2	OR port p data with immediate data	(P0)←(P0) OR data (P1)←(P1) OR data	
	NOP	00	1/1	no operation		

Notes to Table 3

1. PSW CY, AC affected
2. PSW CY affected
3. PSW PS affected
4. Execution of JTF and JNTF instructions resets the Timer Flag (TF).
5. PSW RBS affected
6. PSW SP_0, SP_1, SP_2 affected

* : 8, 9, A, B, C, D, E, F
● : 0, 2, 4, 6, 8, A, C, E
▲ : 1, 3, 5, 7, 9, B, D, F

RATINGS

Limiting values in accordance with the Absolute Maximum System (IEC 134)

Supply voltage (pin 20)	V_{DD}		−0.8 to + 8	V
All input voltages	V_I		−0.8 to V_{DD} + 0.8	V
DC current into any input or output	$\pm I_I, \pm I_O$	max.	10	mA
Total power dissipation (see note 1)	P_{tot}	max.	500	mW
Power dissipation per output	P_O	max.	50	mW
Storage temperature range	T_{stg}		−65 to + 150	°C
Operating ambient temperature range	T_{amb}		−40 to +85	°C
Operating junction temperature	T_j	max.	125	°C

HANDLING

Inputs and outputs are protected against electrostatic discharge in normal handling. However, it is good practice to take normal precautions appropriate to handling MOS devices (see 'Handling MOS devices').

Notes to the ratings

1. Thermal resistance (junction to ambient)

for SOT146	$R_{th\ j\text{-}a}$	max.	120	K/W
for SOT163A	$R_{th\ j\text{-}a}$	max.	150	K/W

DEVELOPMENT DATA

DC CHARACTERISTICS

V_{DD} = 2.5 to 5.5 V; V_{SS} = 0 V; T_{amb} = −40 to + 85 °C; all voltages with respect to V_{SS}; unless otherwise specified

parameter	symbol	min.	typ.	max.	unit
Supply voltage					
operating (see Fig. 3)	V_{DD}	2.5	–	5.5	V
Supply current					
operating (see Fig. 4 and note 2)					
at V_{DD} = 5 V; f_{XTAL} = 10 MHz	I_{DD}	–	1.6	3.2	mA
at V_{DD} = 5 V; f_{XTAL} = 6 MHz	I_{DD}	–	1	2	mA
at V_{DD} = 3 V; f_{XTAL} = 3.58 MHz	I_{DD}	–	0.3	0.6	mA
IDLE mode (see Fig. 5 and note 2)					
at V_{DD} = 5 V; f_{XTAL} = 10 MHz	I_{DD}	–	0.8	1.6	mA
at V_{DD} = 5 V; f_{XTAL} = 6 MHz	I_{DD}	–	0.5	1	mA
at V_{DD} = 3 V; f_{XTAL} = 3.58 MHz	I_{DD}	–	0.15	0.4	mA
STOP mode (see Fig. 8, note 1 and note 2)					
at V_{DD} = 2.5 V; T_{amb} = 25 °C	I_{DD}	–	1.2	2.5	μA
at V_{DD} = 2.5 V; T_{amb} = 85 °C	I_{DD}	–	–	10	μA
Inputs					
Input voltage LOW	V_{IL}	0	–	$0.3V_{DD}$	V
Input voltage HIGH	V_{IH}	$0.7V_{DD}$	–	V_{DD}	V
Input leakage current					
as $V_{SS} < V_I < V_{DD}$	$\pm I_{IL}$	–	–	1	μA
Outputs					
Output sink current LOW					
at V_{DD} = 5 V ± 10%; V_O = 0.4 V	I_{OL}	1.6	3	–	mA
Pull-up output source current HIGH					
at V_{DD} = 5 V ± 10%; V_O = 0.7 V_{DD}	$-I_{OH}$	40	–	–	μA
at V_{DD} = 5 V ± 10%; V_O = V_{SS}	$-I_{OH}$	–	–	400	μA
Push-pull output source current HIGH					
at V_{DD} = 5 V ± 10%; V_O = V_{DD}−0.4 V	$-I_{OH}$	1.6	3	–	mA

Notes to the DC characteristics

1. Crystal connected between XTAL 1 and XTAL 2; T1 = V_{SS}; $\overline{INT}$ = V_{DD}.
2. V_{IL} = V_{SS}; V_{IH} = V_{DD}; all outputs unloaded; all open drain outputs connected to V_{SS}.

DC CHARACTERISTICS (continued)

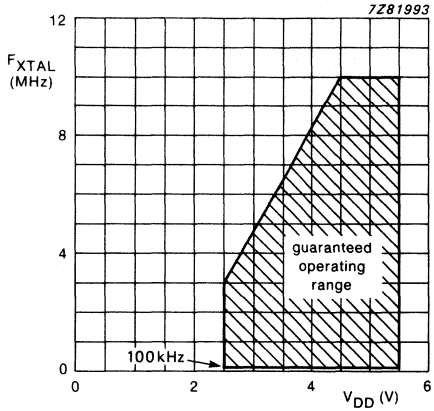
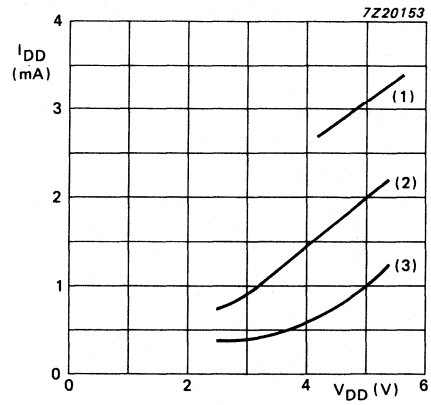
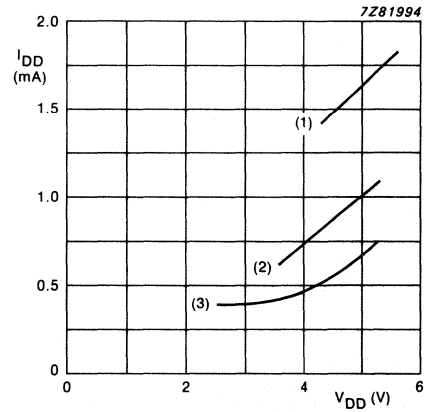


Fig. 3 Maximum clock frequency (f_{XTAL}) as a function of supply voltage (V_{DD}).



(1) f_{OSC} = 10 MHz
(2) f_{OSC} = 6 MHz
(3) f_{OSC} = 3.58 MHz

Fig. 4 Maximum supply current (I_{DD}) in operating mode as a function of supply voltage (V_{DD}).



(1) f_{OSC} = 10 MHz
(2) f_{OSC} = 6 MHz
(3) f_{OSC} = 3.58 MHz

Fig. 5 Maximum supply current (I_{DD}) in IDLE mode as a function of supply voltage (V_{DD}).

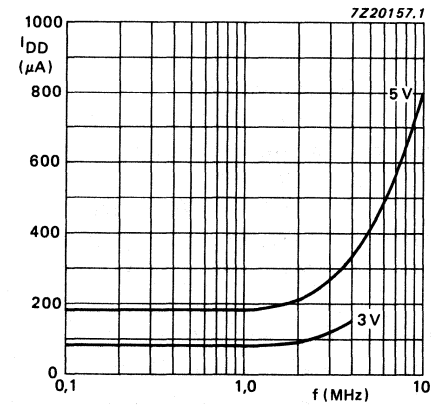


Fig. 6 Typical supply current during IDLE mode as a function of frequency at V_{DD} = 3 V and V_{DD} = 5 V.

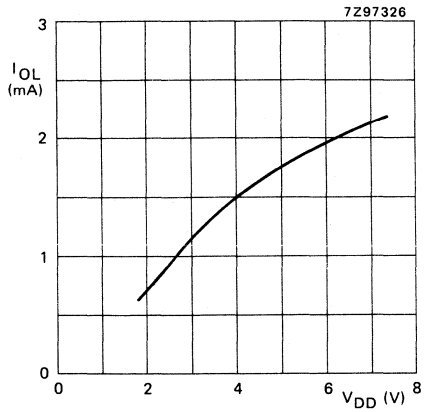
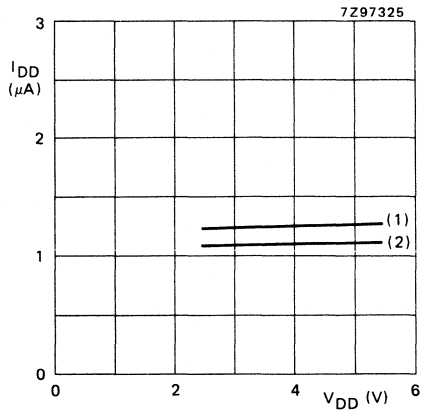


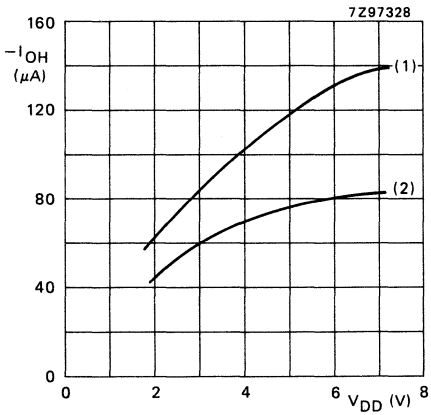
Fig. 7 Typical output sink current (I_{OL}), as a function of supply voltage (V_{DD}); V_O = 0.4 V.



(1) T_{amb} = 85 °C
(2) T_{amb} = 25 °C

Fig. 8 Typical supply current (I_{DD}) in STOP mode as a function of the supply voltage (V_{DD}).

DEVELOPMENT DATA



(1) $V_O = V_{SS}$
(2) $V_O = 0.7\ V_{DD}$

Fig. 9 Typical output source current ($-I_{OH}$) as a function of supply voltage (V_{DD}).

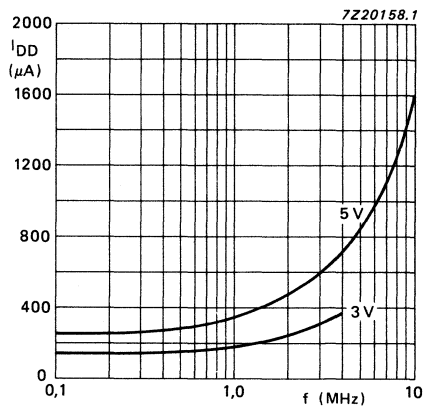


Fig. 10 Typical supply current during operating mode as a function of frequency at V_{DD} = 3 V and V_{DD} = 5 V.

SINGLE-CHIP 8-BIT MICROCONTROLLERS WITH I²C-BUS INTERFACE

DESCRIPTION

An advanced CMOS process is used to manufacture the PCF84C00, PCF84C21/C, PCF84C41/C and PCF84C81/C microcontrollers. The PCF84C21C, PCF84C41C and PCF84C81C operate at a higher clock frequency. Each device has 20 quasi-bidirectional I/O port lines, a serial I/O interface, a single-level vectored interrupt structure, an 8-bit timer/event counter and on-chip clock oscillator and clock circuits. On-chip RAM and ROM content is as follows:

- PCF84C00 – 256 x 8 RAM, external program memory
- PCF84C21 – 64 x 8 RAM, 2 K x 8 ROM
- PCF84C41 – 128 x 8 RAM, 4 K x 8 ROM
- PCF84C81 – 256 x 8 RAM, 8 K x 8 ROM

These efficient controllers also perform well as arithmetic processors. They have facilities for both binary and BCD arithmetic plus bit-handling capabilities. The instruction set is similar to that of the MAB8048.

These microcontrollers are members of the PCF84CXXX family. For detailed information, consult the PCF84CXXX data sheet.

Features

- 8-bit CPU, ROM, RAM, I/O in a single 28-lead DIL or SO package
- 2K, 4 K or 8 K x ROM; also a ROM-less version
- 64, 128 or 256 x 8 RAM
- 20 quasi-bidirectional I/O port lines
- Two test inputs, one of which is also the external interrupt input
- Single-level vectored interrupts: external, timer/event counter and serial I/O
- I²C hardware interface for serial data transfer on two lines (serial I/O data via an existing port line and clock via a dedicated line)
- 8-bit programmable timer/event counter
- Clock frequency range: 100 kHz to 10 MHz ; C versions: 1 MHz to 12 MHz
- Over 80 instructions (similar to those of the MAB8048) all of 1 or 2 cycles
- Single supply voltage (2,5 to 5,5 V)
- STOP and IDLE modes
- Power-on reset circuit
- Operating temperature range: −40 to +85 °C
- High current on Port 1: I_{OL} = 10 mA at V_{OL} = 1,2 V (all versions except the PCF84C00).

Contents	page
Block diagram	2
Pinning	3
Program memory	7,3*
Data memory	7,3*
Program counter stack	4*
IDLE and STOP modes	5*
I/O facilities	7,7*
Serial I/O	10*
Interrupts	15*
Oscillator	18*
Timer/event counter	19*
Program status word	20*

Contents	page
Program counter	20*
Central processing unit	21*
Conditional branch logic	21*
Test input T1	22*
Reset	7
Power-on reset	22*
Instruction set	8*
Ratings	16
DC characteristics	17
AC characteristics	18
Package and soldering details	27

* See PCF84CXXX family Data Sheet.

PACKAGE OUTLINES

PCF84C21/41/81P: 28-lead DIL; plastic (SOT117).
PCF84C21/41/81T: 28-lead mini-pack; plastic (SO28; SOT136A).
PCF84C00B : 28-lead 'piggy-back' package (supports up to 28-pin EPROM).
PCF84C00T : 56-lead mini-pack; plastic (VSO56; SOT190).

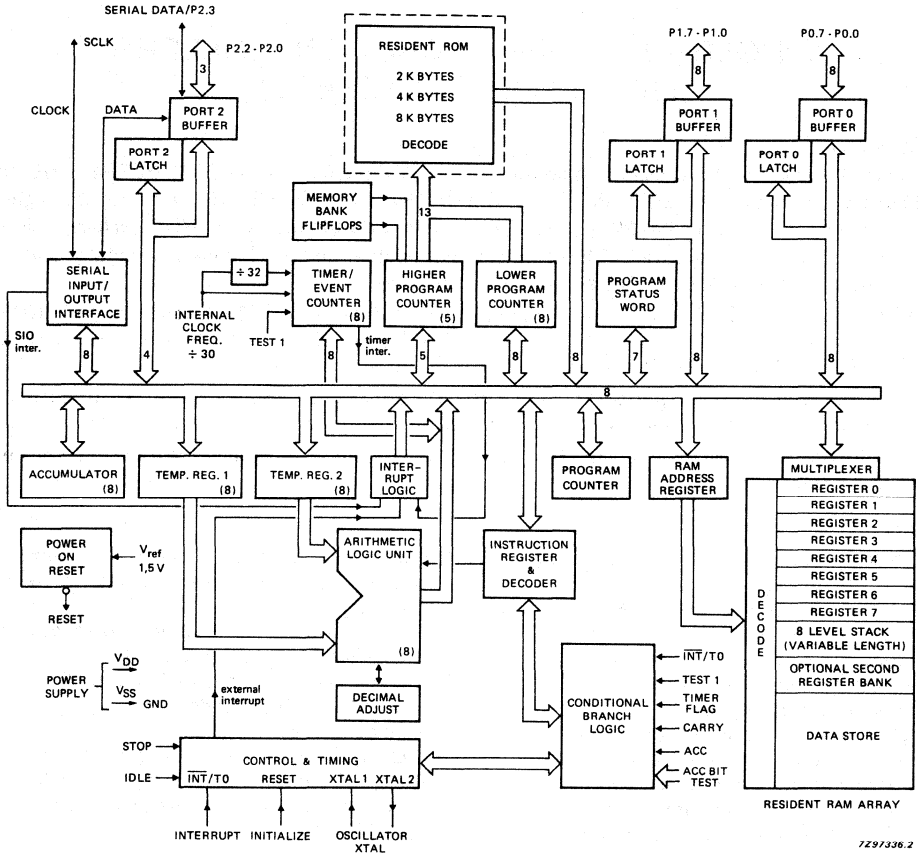


Fig. 1 Block diagram.

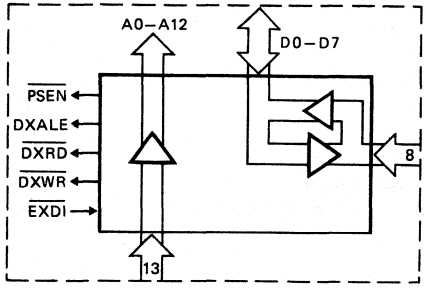


Fig. 1a Replacement of dotted section in Fig. 1, for the PCF84C00T ROM-less version.

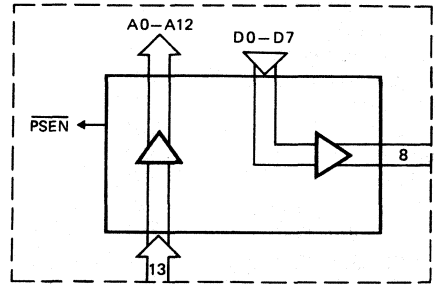


Fig. 1b Replacement of dotted section in Fig. 1, for the PCF84C00B 'piggy-back' version.

PINNING

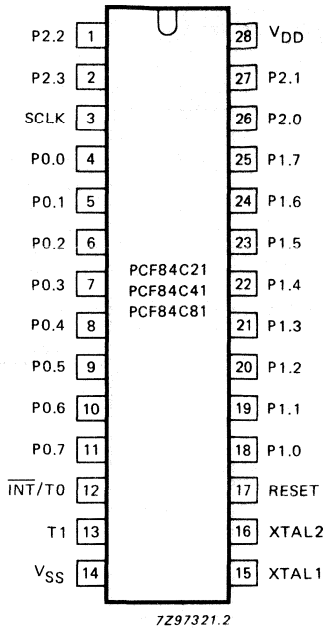


Fig. 2 Pinning diagram.

PIN DESIGNATION

Pin	*Symbol*	*Type*	*Function*
3	SCLK	I/O	**Clock:** bidirectional clock for serial I/O.
4-11	P0.0-P0.7	I/O	**Port 0:** 8-bit quasi-bidirectional I/O port.
12	$\overline{\text{INT}}$/T0	I	**Interrupt/Test 0:** external interrupt input (negative edge triggered)/ test input pin; when used as a test input, this pin is directly tested by conditional branch instructions JT0 and JNT0.
13	T1	I	**Test 1:** test input pin, directly tested by conditional branch instructions JT1 and JNT1. T1 may also be selected as an input to the 8-bit timer/event counter via the STRT CNT instruction.
14	V_{SS}	I	**Ground:** circuit earth potential.
15	XTAL 1	I	**Oscillator input:** input from a crystal which determines the internal oscillator frequency or an external clock generator.
16	XTAL 2	I/O	**Oscillator output:** output of the inverting amplifier.
17	RESET	I/O	**Reset input:** used to initialize the microcontroller (active HIGH); also output of power-on-reset circuit.
18-25	P1.0-P1.7	I/O	**Port 1:** 8-bit quasi-bidirectional I/O port.
26, 27, 1, 2	P2.0-P2.3	I/O	**Port 2:** 4-bit quasi-bidirectional I/O port. P2.3 is the serial data input/output in serial I/O mode.
28	V_{DD}	I	**Power supply:** 2,5 V to 5,5 V.

PINNING (continued)

Pin diagram of the PCF84C00B 'piggy-back' version

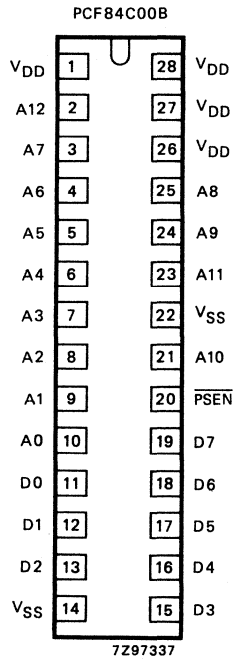


Fig. 3(a) Pinning diagram (top pins) of the PCF84C00B 'piggy-back' version.

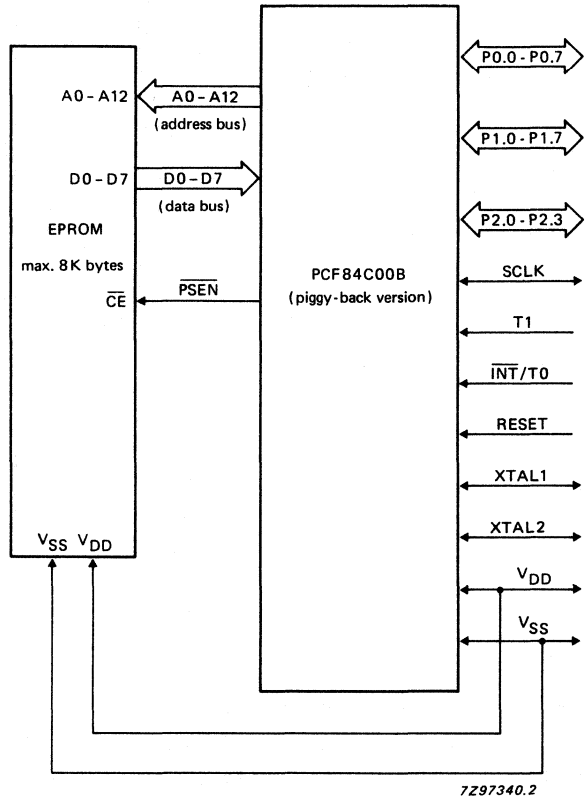


Fig. 3(b) Connection of EPROM to the PCF84C00B 'piggy-back' version.

The PCF84C00B is mainly used for prototyping and for low volume production applications. This device is packaged in a 'piggy-back' package. i.e. a 4 K or 8 K byte EPROM (2732 or 2764) may be mounted in the 24/28 pin socket on top of the package.

Notes

1. The PCF84C00B has 256 bytes of on-chip RAM.
2. Access time for ROMS/EPROMS must be less than $7 \times 1/f_{XTAL}$.
3. Bottom pinning is identical to that of Fig. 2.

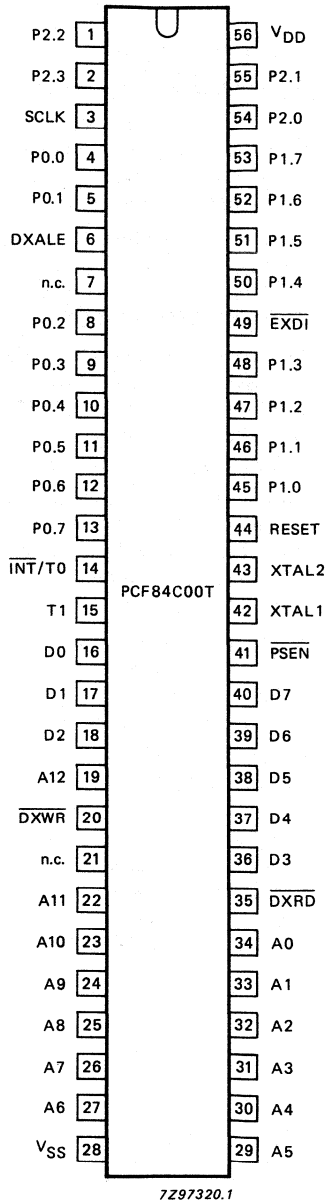


Fig. 4 Pinning diagram; ROM-less version PCF84C00T.

The PCF84C00T may be used for prototyping future derivatives of the PCF84CXX family or for low volume production applications. This device is packaged in a 56-lead VSO outline. Additional signals are available (see pinning information following) to control external program memory and derivative functions.

PIN FUNCTION (continued)

Pin	*Symbol*	*Type*	*Function*
34-29, 27-22, 19	A00-A12	O	**Address bus.** For external memory and pheripherals.
16-18, 36-40	D0-D7	I/O	**Data bus.** For external memory and pheripherals. The specified STOP mode supply current is valid only if external pull-ups are connected to all data lines.
41	$\overline{\text{PSEN}}$	O	**Program store enable** (active LOW). $\overline{\text{PSEN}}$ is used to enable external program memory and is active during TS9 and TS10 of each machine cycle and TS1 of each following cycle. $\overline{\text{PSEN}}$ is HIGH during the STOP mode.
6	DXALE	O	**Address latch enable.** On the falling edge of DXALE, the Dx address can be latched in an external latch. This signal occurs only during execution of the MOV Dx,A, MOV A,Dx, ANL Dx,A and ORL Dx,A instructions, with x = 0 to 255. It is active during TS10 of cycle 1 and the first half of TS1 of cycle 2.
35	$\overline{\text{DXRD}}$	O	**Read strobe** (active LOW). When this signal is active, external registers emulating Dx registers can be read by the data bus. This signal occurs only during execution of MOV A,Dx, ANL Dx,A and ORL Dx,A instructions, with x = 0 to 255. It is active during TS3 and TS4 of cycle 2.
20	$\overline{\text{DXWR}}$	O	**Write strobe** (active LOW). On the rising edge, data on D0-D7 may be written to external registers. This signal occurs only during MOV Dx,A, ANL Dx,A and ORL Dx,A instructions, with x = 0 to 255. It is active during TS7 of cycle 2.
49	$\overline{\text{EXDI}}$	I	**External derivative interrupt** (active LOW). $\overline{\text{EXDI}}$ is 'OR-ed' with the internal serial interrupt and can be used to initiate an interrupt from external hardware emulating derivative functions. $\overline{\text{EXDI}}$ is pulled HIGH internally. The derivative interrupt is polled during time slot TS6*, and is only accepted if an EN SI instruction has been executed and the device is not already executing an interrupt routine. Derivative interrupts are not latched in the PCF84C00.

* The interrupt signal must remain active until the vector address (05 H) is present on the address bus.

FUNCTIONAL DESCRIPTION

ROM-less version PCF84C00T

The PCF84C00T microcontroller contains no on-chip ROM, but has all address and data lines brought out to access an external ROM or EPROM. This version has more pins than the PCF84CXXX with on-chip ROM (see Fig. 1a). The PCF84C00T can address up to 8 K bytes of external program memory, and has 256 bytes of internal data RAM.

'Piggy-back' version PCF84C00B

The PCF84C00B package has standard pinning on the bottom to facilitate insertion as a mask-programmed device. An EPROM can be mounted on top in an additional socket. The total package height is greater than the height of a standard DIL package. The PCF84C00B can address up to 8 K bytes of external ROM/RAM, and has 256 bytes of internal data RAM.

Program memory

The program memory consists of 2, 4 or 8 K bytes of read-only memory (ROM). Each location is directly addressable by the program counter. The ROM is mask-programmed at the factory.

Data memory

Data memory consists of 64, 128 or 256 bytes of random-access memory (RAM). All locations are indirectly addressable using RAM pointer registers; up to 16 designated locations are directly addressable. Memory also includes an 8-level program counter stack addressed by a 3-bit stack pointer.

I/O facilities

Each device has 23 I/O lines arranged as follows:

- Port 0 — 8-bit parallel port (P0.0-P0.7)
- Port 1 — 8-bit parallel port (P1.0-P1.7)
- Port 2 — 4-bit parallel port (P2.0-P2.3)
- SCLK — serial I/O clock line
- $\overline{\text{INT}}$/T0 — external interrupt and test input. When used as a test input, it can be directly tested by conditional branch instructions JT0 and JNT0
- T1 — test input which can alter program sequences when tested by conditional jump instructions JT1 and JNT1. T1 also functions as an input to the 8-bit timer/event counter

Reset

A positive-going signal on the RESET input/output:

- Sets the program counter to zero
- Selects location 0 of memory bank 0 and register bank 0
- Sets the stack pointer to zero (000); pointing to RAM address 8
- Disables the interrupts (external, timer and serial I/O)
- Stops the timer/event counter, then sets it to zero
- Sets the timer prescaler to divide by 32
- Resets the timer flag
- Sets all ports except P2.3 to input mode
- Sets the serial I/O to slave receiver mode and disables the serial I/O
- Cancels IDLE and STOP mode

A negative-going signal on the RESET input/output:

- Sets P2.3/SDA and SCLK to HIGH after a maximum of 30 clock pulses
- Sets the serial I/O to slave receiver mode and disables the serial I/O after a maximum of 30 clock pulses
- Starts program execution after 1866 clock pulses

INSTRUCTION SET

The instruction set consists of over 80 one and two byte instructions. Program code efficiency is high because all RAM locations and all ROM locations on a 256 byte page require only a single byte address.

Table 3 contains the instruction set. Table 2 shows the instruction map and Table 1 details the symbols that are used.

Table 1 Symbols and definitions used in Table 3

symbol	description
A	accumulator
addr	program memory address
Bb	bit designation (b = 0-7)
RBS	register bank select
C	carry bit (bit CY)
CNT	event counter
Dx	mnemonic derivative register (x = 0 ... 255)
direct	8-bit derivative register address
data	8-bit number or expression
I	interrupt
MB	memory bank
MBFF	memory bank flip-flop
P	mnemonic for 'in-page' operation
PC	program counter
Pp	port designation (p = 0, 1 or 2)
PSW	program status word
RB	register bank
Rr	register designation (r = 0-7)
Sn	serial I/O register
SP	stack pointer
T	timer
TF	timer flag
T0, T1	test 0 and 1 inputs
#	immediate data prefix
@	indirect address prefix
(X)	contents of X
((X))	contents of location addressed by X
←	is replaced by
↔	is exchanged with

INSTRUCTION SET (continued)

Table 2 Instruction map

first hexadecimal character of opcode ↓ / second hexadecimal character of opcode →	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	NOP	IDLE		ADD A, # data	JMP page 0	EN I	JNTF addr	DEC A	IN A,Pp 0	IN A,Pp 1	IN A,Pp 2		MOV A,Sn 0	MOV A,Sn 1		
1	INC @Rr 0	INC @Rr 1	JB0 addr	ADDC A, # data	CALL page 0	DIS I	JTF addr	INC A	INC Rr 0	INC Rr 1	INC Rr 2	INC Rr 3	INC Rr 4	INC Rr 5	INC Rr 6	INC Rr 7
2	XCH A,@Rr 0	XCH A,@Rr 1	STOP	MOV A, # data	JMP page 1	EN TCNTI	JNT0 addr	CLR A	XCH A,Rr 0	XCH A,Rr 1	XCH A,Rr 2	XCH A,Rr 3	XCH A,Rr 4	XCH A,Rr 5	XCH A,Rr 6	XCH A,Rr 7
3	XCHD A,@Rr 0	XCHD A,@Rr 1	JB1 addr		CALL page 1	DIS TCNTI	JT0 addr	CPL A	OUTL Pp,A 0	OUTL Pp,A 1	OUTL Pp,A 2		MOV Sn,A 0	MOV Sn,A 1	MOV Sn,A 2	
4	ORL A,@Rr 0	ORL A,@Rr 1	MOV A, T	ORL A, # data	JMP page 2	STRT CNT	JNT1 addr	SWAP A	ORL A,Rr 0	ORL A,Rr 1	ORL A,Rr 2	ORL A,Rr 3	ORL A,Rr 4	ORL A,Rr 5	ORL A,Rr 6	ORL A,Rr 7
5	ANL A,@Rr 0	ANL A,@Rr 1	JB2 addr	ANL A, # data	CALL page 2	STRT T	JT1 addr	DA A	ANL A,Rr 0	ANL A,Rr 1	ANL A,Rr 2	ANL A,Rr 3	ANL A,Rr 4	ANL A,Rr 5	ANL A,Rr 6	ANL A,Rr 7
6	ADD A,@Rr 0	ADD A,@Rr 1	MOV T, A		JMP page 3	STOP TCNT		RRC A	ADD A,Rr 0	ADD A,Rr 1	ADD A,Rr 2	ADD A,Rr 3	ADD A,Rr 4	ADD A,Rr 5	ADD A,Rr 6	ADD A,Rr 7
7	ADDC A,@Rr 0	ADDC A,@Rr 1	JB3 addr		CALL page 3			RR A	ADDC A,Rr 0	ADDC A,Rr 1	ADDC A,Rr 2	ADDC A,Rr 3	ADDC A,Rr 4	ADDC A,Rr 5	ADDC A,Rr 6	ADDC A,Rr 7
8				RET	JMP page 4	EN SI			ORL Pp, # data 0	ORL Pp, # data 1	ORL Pp, # data 2		MOV A, Dx	MOV Dx, A	ANL Dx, A	ORL Dx, A
9			JB4 addr	RETR	CALL page 4	DIS SI	JNZ addr	CLR C	ANL Pp,# data 0	ANL Pp,# data 1	ANL Pp,# data 2		MOV Sn,# data 0	MOV Sn,# data 1	MOV Sn,# data 2	
A	MOV @Rr,A 0	MOV @Rr,A 1		MOVP A,@A	JMP page 5	SEL MB2		CPL C	MOV Rr,A 0	MOV Rr,A 1	MOV Rr,A 2	MOV Rr,A 3	MOV Rr,A 4	MOV Rr,A 5	MOV Rr,A 6	MOV Rr,A 7
B	MOV @Rr, # data 0	MOV @Rr, # data 1	JB5 addr	JMPP @A	CALL page 5	SEL MB3			MOV Rr, # data 0	MOV Rr, # data 1	MOV Rr, # data 2	MOV Rr, # data 3	MOV Rr, # data 4	MOV Rr, # data 5	MOV Rr, # data 6	MOV Rr, # data 7
C	DEC @Rr 0	DEC @Rr 1			JMP page 6	SEL RB0	JZ addr	MOV A, PSW	DEC Rr 0	DEC Rr 1	DEC Rr 2	DEC Rr 3	DEC Rr 4	DEC Rr 5	DEC Rr 6	DEC Rr 7
D	XRL A,@Rr 0	XRL A,@Rr 1	JB6 addr	XRL A,# data	CALL page 6	SEL RB1		MOV PSW, A	XRL A,Rr 0	XRL A,Rr 1	XRL A,Rr 2	XRL A,Rr 3	XRL A,Rr 4	XRL A,Rr 5	XRL A,Rr 6	XRL A,Rr 7
E	DJNX @Rr, addr 0	DJNX @Rr, addr 1			JMP page 7	SEL MB0	JNC addr	RL A	DJNZ Rr, addr 0	DJNZ Rr, addr 1	DJNZ Rr, addr 2	DJNZ Rr, addr 3	DJNZ Rr, addr 4	DJNZ Rr, addr 5	DJNZ Rr, addr 6	DJNZ Rr, addr 7
F	MOV A,@Rr 0	MOV A,@Rr 1	JB7 addr		CALL page 7	SEL MB1	JC addr	RLC A	MOV A,Rr 0	MOV A,Rr 1	MOV A,Rr 2	MOV A,Rr 3	MOV A,Rr 4	MOV A,Rr 5	MOV A,Rr 6	MOV A,Rr 7

Table 3 Instruction set

	mnemonic	opcode (hex.)	bytes/ cycles	description	function		notes
ACCUMULATOR	ADD A, Rr	6*	1/1	Add register contents to A	(A)←(A) + (Rr)	r = 0–7	1
	ADD A, @Rr	60 61	1/1	Add RAM data, addressed by Rr, to A	(A)←(A) + ((R0)) (A)←(A) + ((R1))		1
	ADD A, #data	03 data	2/2	Add immediate data to A	(A)←(A) + data		1
	ADDC A, Rr	7*	1/1	Add carry and register contents to A	(A)←(A) + (Rr) + (C)	r = 0–7	1
	ADDC A, @Rr	70 71	1/1	Add carry and RAM data, addressed by Rr, to A	(A)←(A) + ((R0)) + (C) (A)←(A) + ((R1)) + (C)		1
	ADDC A, #data	13 data	2/2	Add carry and immediate data to A	(A)←(A) + data + (C)		1
	ANL A, Rr	5*	1/1	'AND' Rr with A	(A)←(A) AND (Rr)	r = 0–7	
	ANL A, @Rr	50 51	1/1	'AND' RAM data, addressed by Rr, with A	(A)←(A) AND ((R0)) (A)←(A) AND ((R1))		
	ANL A, #data	53 data	2/2	'AND' immediate data with A	(A)←(A) AND data		
	ORL A, Rr	4*	1/1	'OR' Rr with A	(A)←(A) OR (Rr)	r = 0–7	
	ORL A, @Rr	40 41	1/1	'OR' RAM data, addressed by Rr, with A	(A)←(A) OR ((R0)) (A)←(A) OR ((R1))		
	ORL A, #data	43 data	2/2	'OR' immediate data with A	(A)←(A) OR data		
	XRL A, Rr	D*	1/1	'XOR' Rr with A	(A)←(A) XOR (Rr)	r = 0–7	
	XRL A, @Rr	D0 D1	1/1	'XOR' RAM, addressed by Rr, with A	(A)←(A) XOR ((R0)) (A)←(A) XOR ((R1))		
	XRL A, #data	D3 data	2/2	'XOR' immediate data with A	(A)←(A) XOR data		
	INC A	17	1/1	increment A by 1	(A)←(A) + 1		
	DEC A	07	1/1	decrement A by 1	(A)←(A) −1		
	CLR A	27	1/1	clear A to zero	(A)←0		
	CPL A	37	1/1	one's complement A	(A)←NOT(A)		
	RL A	E7	1/1	rotate A left	$(A_{n+1})\leftarrow(A_n)$ $(A_0)\leftarrow(A_7)$	n = 0–6	

ACCUMULATOR (cont.)	RLC A	F7	1/1	rotate A left through carry	$(A_{n+1}) \leftarrow A_n$ $(A_0) \leftarrow (C)$, $(C) \leftarrow (A_7)$	n = 0–6	2
	RR A	77	1/1	rotate A right	$(A_n) \leftarrow (A_{n+1})$ $(A_7) \leftarrow (A_0)$	n = 0–6	
	RRC A	67	1/1	rotate A right through carry	$(A_n) \leftarrow (A_{n+1})$ $(A_7) \leftarrow (C)$, $(C) \leftarrow (A_0)$	n = 0–6	2
	DA A	57	1/1	decimal adjust A			2
	SWAP A	47	1/1	swap nibbles of A	$(A_{4-7}) \leftrightarrow (A_{0-3})$		
DATA MOVES	MOV A, Rr	F*	1/1	move register contents to A	(A)←(Rr)	r = 0–7	
	MOV A, @Rr	F0 F1	1/1	move RAM data, addressed by Rr, to A	(A)←((R0)) (A)←((R1))		
	MOV A, #data	23 data	2/2	move immediate data to A	(A)←data		
	MOV Rr, A	A*	1/1	move accumulator contents to register	(Rr)←(A)	r = 0–7	
	MOV @Rr, A	A0 A1	1/1	move accumulator contents to RAM location addressed by Rr	((R0))←(A) ((R1))←(A)		
	MOV Rr, #data	B* data	2/2	move immediate data to Rr	(Rr)←data		
	MOV @Rr, #data	B0 data B1 data	2/2	move immediate data to RAM location addressed by Rr	((R0))←data ((R1))←data		
	XCH A, Rr	2*	1/1	exchange accumulator contents with Rr	(A)↔(Rr)	r = 0–7	
	XCH A, @Rr	20 21	1/1	exchange accumulator contents with RAM data addressed by Rr	(A)↔((R0)) (A)↔((R1))		
	XCHD A, @Rr	30 31	1/1	exchange lower nibbles of A and RAM data addressed by Rr	$(A_{0-3}) \leftrightarrow ((R0_{0-3}))$ $(A_{0-3}) \leftrightarrow ((R1_{0-3}))$		
	MOV A, PSW	C7	1/1	move PSW contents to accumulator	(A)←(PSW)		
	MOV PSW, A	D7	1/1	move accumulator bit 3 to PSW_3	$(PSW_3) \leftarrow (A_3)$		3
	MOVP A, @A	A3	1/2	move indirectly addressed data in current page to A	$(PC_{0-7}) \leftarrow (A)$, (A)←((PC))		
FLAGS	CLR C	97	1/1	clear carry bit	(C)←0		2
	CPL C	A7	1/1	complement carry bit	(C)←NOT(C)		2

INSTRUCTION SET (continued)

	mnemonic	opcode (hex.)	bytes/ cycles	description	function	notes
REGISTER	INC Rr	1*	1/1	increment register by 1	(Rr)←(Rr) + 1 r = 0–7	
	INC @Rr	10 11	1/1	increment RAM data, addressed by Rr, by 1	((R0))←((R0)) + 1 ((R1))←((R1)) + 1	
	DEC Rr	C*	1/1	decrement register by 1	(Rr)←(Rr) – 1 r = 0–7	
	DEC @Rr	C0 C1	1/1	decrement RAM data, addressed by Rr, by 1	((R0))←((R0)) – 1 ((R1))←((R1)) – 1	
BRANCH	JMP addr	● 4 addr	2/2	unconditional jump within a 2 K bank	(PC_{8-10})←$addr_{8-10}$ (PC_{0-7})←$addr_{0-7}$ (PC_{11-12})←MBFF 0–1	
	JMPP @A	B3	1/2	indirect jump within a page	(PC_{0-7})←((A))	
	DJNZ Rr, addr	E* addr	2/2	decrement Rr by 1 and jump if not zero to addr	(Rr)←(Rr) – 1 r = 0–7 if (Rr) not zero (PC_{0-7})←addr	
	DJNZ @Rr, addr	E0	2/2	decrement RAM data, addressed by Rr by 1 and jump if not zero to addr	((R0))←((R0)) – 1 if ((R0)) not zero (PC_{0-7})←addr	
		E1			((R1))←((R1)) – 1 if ((R1)) not zero (PC_{0-7})←addr	
	JBb addr	▲ 2 addr	2/2	jump to addr if Acc. bit b = 1	if b = 1 : (PC_{0-7})←addr b = 0–7	
	JC addr	F6 addr	2/2	jump to addr if C = 1	if C = 1 : (PC_{0-7})←addr	
	JNC addr	E6 addr	2/2	jump to addr if C = 0	if C = 0 : (PC_{0-7})←addr	
	JZ addr	C6 addr	2/2	jump to addr if A = 0	if A = 0 : (PC_{0-7})←addr	
	JNZ addr	96 addr	2/2	jump to addr if A is NOT zero	if A≠0 : (PC_{0-7})←addr	
	JT0 addr	36 addr	2/2	jump to addr if T0 = 1	if T0 = 1: (PC_{0-7})←addr	
	JNT0 addr	26 addr	2/2	jump to addr if T0 = 0	if T0 = 0: (PC_{0-7})←addr	
	JT1 addr	56 addr	2/2	jump to addr if T1 = 1	if T1 = 1: (PC_{0-7})←addr	
	JNT1 addr	46 addr	2/2	jump to addr if T1 = 0	if T1 = 0: (PC_{0-7})←addr	
	JTF addr	16 addr	2/2	jump to addr if Timer Flag = 1	if TF = 1: (PC_{0-7})←addr	4
	JNTF addr	06 addr	2/2	jump to addr if Timer Flag = 0	if TF = 0: (PC_{0-7})←addr	

TIMER/EVENT COUNTER	MOV A, T	42	1/1	move timer/event counter contents to accumulator	(A)←(T)	
	MOV T, A	62	1/1	move accumulator contents to timer/event counter	(T)←(A)	
	STRT CNT	45	1/1	start event counter		
	STRT T	55	1/1	start timer		
	STOP TCNT	65	1/1	stop timer/event counter		
	EN TCNTI	25	1/1	enable timer/event counter interrupt		
	DIS TCNTI	35	1/1	disable timer/event counter interrupt		
CONTROL	EN I	05	1/1	enable external interrupt		
	DIS I	15	1/1	disable external interrupt		
	SEL RB0	C5	1/1	select register bank 0	(RBS)←0	5
	SEL RB1	D5	1/1	select register bank 1	(RBS)←1	5
	SEL MB0	E5	1/1	select program memory bank 0	(MBFF0)←0, (MBFF1)←0	
	SEL MB1	F5	1/1	select program memory bank 1	(MBFF0)←1, (MBFF1)←0	10
	SEL MB2	A5	1/1	select program memory bank 2	(MBFF0)←0, (MBFF1)←1	10
	SEL MB3	B5	1/1	select program memory bank 3	(MBFF0)←1, (MBFF1)←1	10
	STOP	22	1/1	enter STOP mode		
	IDLE	01	1/1	enter IDLE mode		
SUBROUTINE	CALL addr	▲ 4 addr	2/2	jump to subroutine	((SP))←(PC), ($PSW_{4, 6, 7}$) (SP)←(SP) + 1 (PC_{8-10})←$addr_{8-10}$ (PC_{0-7})←$addr_{0-7}$ (PC_{11-12})←MBFF 0–1	6
	RET	83	1/2	return from subroutine	(SP)←(SP) – 1 (PC)←((SP))	6
	RETR	93	1/2	return from interrupt and restore bits 4, 6, 7 of PSW	(SP)←(SP) – 1 ($PSW_{4, 6, 7}$) + (PC)←((SP))	6

INSTRUCTION SET (continued)

	mnemonic	opcode (hex.)	bytes/ cycles	description	function	notes
PARALLEL INPUT/OUTPUT	IN A, Pp	08 09 0A	1/2	input port p data to accumulator	(A)←(P0) (A)←(P1) (A)←(P2)	 7
	OUTL Pp, A	38 39 3A	1/2	output accumulator data to port p	(P0)←(A) (P1)←(A) (P2)←(A)	
	ANL Pp, #data	98 data 99 data 9A data	2/2	AND port p data with immediate data	(P0)←(P0) AND data (P1)←(P1) AND data (P2)←(P2) AND data	
	ORL Pp, #data	88 data 89 data 8A data	2/2	OR port p data with immediate data	(P0)←(P0) OR data (P1)←(P1) OR data (P2)←(P2) OR data	
DERIVATIVE INPUT/OUTPUT	MOV A, Dx	8C direct	2/2	move derivative register contents to accumulator	(A)←(Dx) x = 0 to 255	8
	MOV Dx, A	8D direct	2/2	move accumulator contents to derivative register	(Dx)←(A) x = 0 to 255	8
	ANL Dx, A	8E direct	2/2	AND derivative register with accumulator	(Dx)←(Dx) AND (A) x = 0 to 255	8
	ORL Dx, A	8F direct	2/2	OR derivative register with accumulator	(Dx)←(Dx) OR (A) x = 0 to 255	8

	mnemonic	opcode (hex.)	bytes/ cycles	description	function	notes
SERIAL INPUT/OUTPUT	MOV A, S_n	0C 0D	1/2	move serial I/O register contents to accumulator	(A)←(S0) (A)←(S1)	9
	MOV S_n, A	3C 3D 3E	1/2	move accumulator contents to serial I/O register	(S0)←(A) (S1)←(A) (S2)←(A)	
	MOV S_n, #data	9C data 9D data 9E data	2/2	move immediate data to serial I/O register	(S0)←data (S1)←data (S2)←data	
	EN SI	85	1/1	enable serial I/O interrupt		
	DIS SI	95	1/1	disable serial I/O interrupt		
	NOP	00	1/1	no operation		

Notes to Table 3

1. PSW CY, AC affected
2. PSW CY affected
3. PSW PS affected
4. Execution of a JTF or JNTF instruction resets the Timer Flag (TF).

* : 8, 9, A, B, C, D, E, F
● : 0, 2, 4, 5, 6, A, C, E
▲ : 1, 3, 5, 7, 9, B, D, F

5. PSW RBS affected
6. PSW SP_0, SP_1, SP_2 affected
7. (A) = 0000, P2.3, P2.2, P2.1, P2.0.
8. Instructions for PCF84C00T only.
9. (S1) has a different meaning for read and write operation, see serial I/O interface.
10. SEL MB1, SEL MB2 and SEL MB3 may not be used within interrupt routines.

RATINGS

Limiting values in accordance with the Absolute Maximum System (IEC 134)

Supply voltage	V_{DD}	−0,8 to +8	V
All input voltages	V_I	−0,5 to V_{DD} +0,5	V
DC current into any input or output	$\pm I_I$, $\pm I_O$	max. 10	mA
Total power dissipation (see note)	P_{tot}	max. 125	mW
Storage temperature range	T_{stg}	−65 to +150	°C
Operating ambient temperature range (if P_{tot} max. = 100 mW)	T_{amb}	−40 to +70	°C
Operating ambient temperature range (if P_{tot} max. = 30 mW)	T_{amb}	−40 to +85	°C
Operating junction temperature	T_j	max. 90	°C

HANDLING

Inputs and outputs are protected against electrostatic discharge in normal handling. However, it is good practice to take normal precautions appropriate to handling MOS devices (see 'Handling MOS devices').

Note

Thermal resistance (junction to ambient)

for SOT-117	$R_{th\ j\text{-}a}$	max. 120	K/W
for SOT-136A	$R_{th\ j\text{-}a}$	max. 150	K/W
for SOT-190	$R_{th\ j\text{-}a}$	max. 110	K/W

DC CHARACTERISTICS

V_{DD} = 2,5 to 5,5 V; V_{SS} = 0 V; T_{amb} = −40 to +85 °C; all voltages with respect to V_{SS}; unless otherwise specified.

parameter	symbol	min.	typ.	max.	unit
Supply voltage					
operating (see Fig. 9)	V_{DD}	2,5	–	5,5	V
Supply current					
operating (see Fig. 10; not valid for PCF84C00)					
at V_{DD} = 5 V; f_{XTAL} = 10 MHz (note 2)	I_{DD}	–	1,6	3,2	mA
at V_{DD} = 5 V; f_{XTAL} = 6 MHz (note 2)	I_{DD}	–	1	2	mA
at V_{DD} = 3 V; f_{XTAL} = 3,58 MHz (note 2)	I_{DD}	–	0,3	0,6	mA
IDLE mode (see Fig. 11; not valid for PCF84C00)					
at V_{DD} = 5 V; f_{XTAL} = 10 MHz (note 2)	I_{DD}	–	0,8	1,6	mA
at V_{DD} = 5 V; f_{XTAL} = 6 MHz (note 2)	I_{DD}	–	0,5	1	mA
at V_{DD} = 3 V; f_{XTAL} = 3,58 MHz (note 2)	I_{DD}	–	0,15	0,4	mA
STOP mode (see Fig. 17, note 1 and note 2)					
at V_{DD} = 2,5 V; T_{amb} = 25 °C	I_{DD}	–	1,2	2,5	μA
at V_{DD} = 2,5 V; T_{amb} = 85 °C	I_{DD}	–	–	10	μA
Inputs					
Input voltage LOW	V_{IL}	0	–	$0{,}3V_{DD}$	V
Input voltage HIGH	V_{IH}	$0{,}7V_{DD}$	–	V_{DD}	V
Input leakage current					
at $V_{SS} < V_I < V_{DD}$	$\pm I_{IL}$	–	–	1	μA
Outputs					
Output sink current LOW					
at V_{DD} = 5 V ± 10%; V_O = 0,4 V					
except P2.3/SDA, SCLK (see Fig. 13) and port 1	I_{OL}	1,6	3	–	mA
P2.3/SDA, SCLK (see Fig. 14)	I_{OL}	3	–	–	mA
P1.0-P1.7 (not PCF84C00) at V_{OL} = 1,2 V	I_{OL}	10	–	–	mA
Pull-up output source current HIGH (see Fig. 15)					
at V_{DD} = 5 V ± 10%; V_O = $0{,}7V_{DD}$	$-I_{OH}$	40	–	–	μA
at V_{DD} = 5 V ± 10%; V_O = V_{SS}	$-I_{OH}$	–	–	400	μA
Push-pull output source current HIGH					
at V_{DD} = 5 V ± 10%; V_O = V_{DD}−0,4 V	$-I_{OH}$	1,6	3	–	mA

Note 1: Crystal connected between XTAL1 and XTAL2; T1 = V_{SS}; $\overline{INT}$ = V_{DD}.

Note 2: V_{IL} = V_{SS}; V_{IH} = V_{DD}; all outputs unloaded; all open drain outputs connected to V_{SS}.

AC CHARACTERISTICS

V_{DD} = 2,5 to 5,5 V; V_{SS} = 0 V; T_{amb} = −40 to +85 °C. All voltages with respect to V_{SS} unless otherwise specified.

parameter	symbol	min.	typ.	max.	unit
Rise time all outputs (note 1)	t_R	–	30	–	ns
Fall time all outputs (note 1)	t_F	–	30	–	ns
Cycle time (= 30 CP; note 2)	t_{CY}	3	–	30	μs
PCF84C00/non-standard pins:					
Control pulse width	t_{CC}	–	9	–	CP
Address to $\overline{PSEN}$ set-up	t_{AS}	–	1,5	–	CP
Data to $\overline{PSEN}$ set-up	t_{DS}	–	2	–	CP
Data hold time	t_{DR}	0	–	–	ns
Data out to $\overline{DWXR}$ set-up	t_{SDO}	–	2	–	CP
Data out to $\overline{DXWR}$ hold	t_{HDO}	–	1	–	CP
Time from DXALE to $\overline{PSEN}$	t_{SLPH}	–	1,5	–	CP
Data-in to $\overline{DXRD}$ set-up	t_{DS1}	–	2,5	–	CP
Data-in to $\overline{DXRD}$ hold	t_{DR1}	0	–	–	ns
HIGH time of DXALE	t_{DXALE}	–	4,5	–	CP
LOW time of $\overline{DXRD}$	t_{DXRD}	–	6	–	CP
LOW time of $\overline{DXWR}$	t_{DXWR}	–	3	–	CP

Notes:

1. At V_{DD} = 5 V; T_{amb} = + 25 °C; C_L = 50 pF.
2. 1 Time slot (TS) = 3 CP, 1 clock pulse (CP) = $1/f_{XTAL}$.

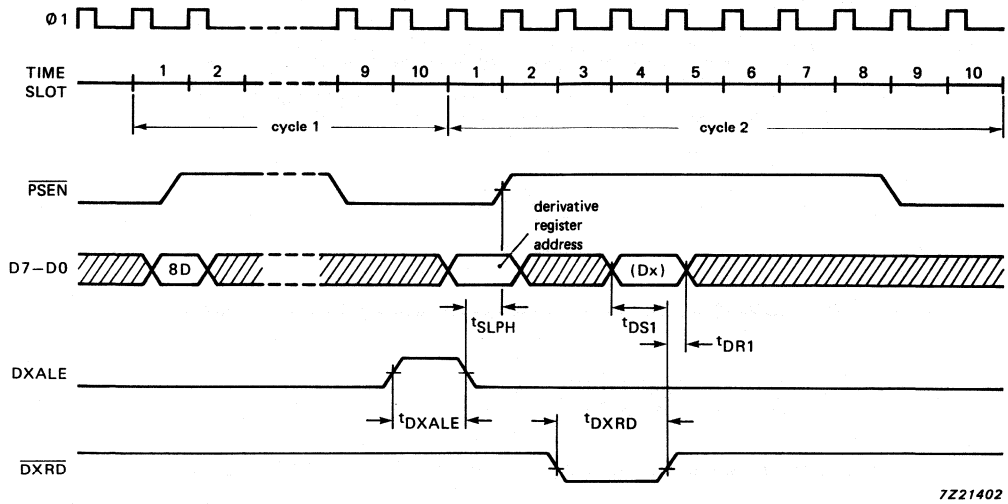


Fig. 5 MOV A,Dx timing (PCF84C00T only).

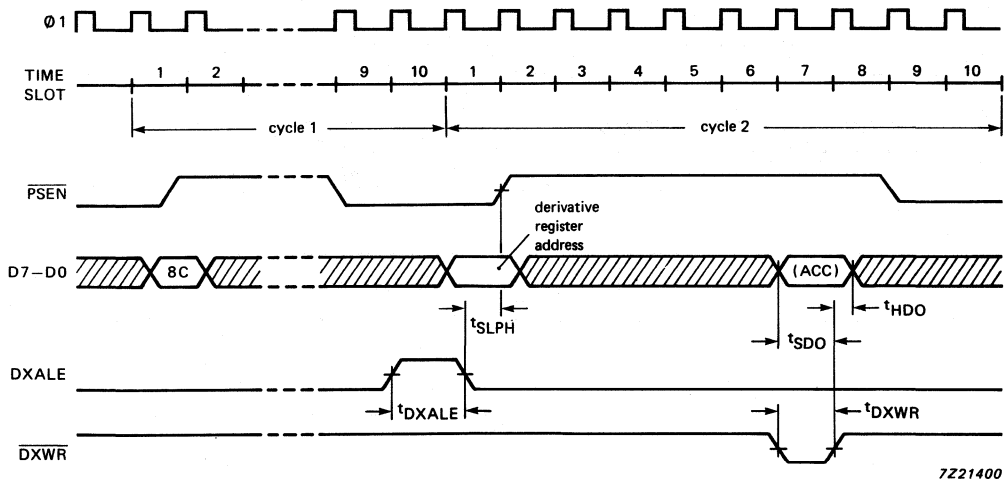


Fig. 6 MOV Dx,A timing (PCF84C00T only).

AC CHARACTERISTICS (continued)

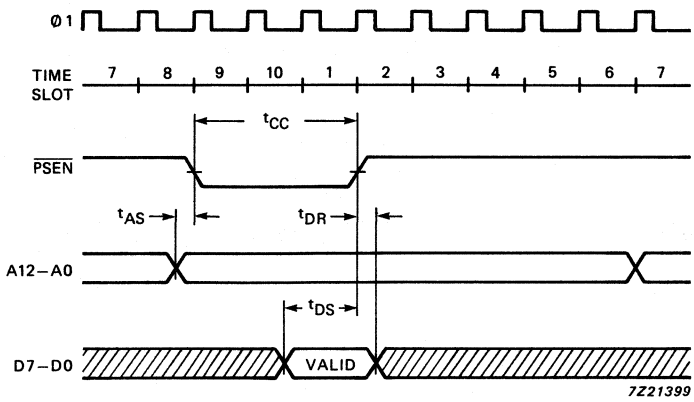


Fig. 7 External memory access timing (PCF84C00T and PCF8400B).

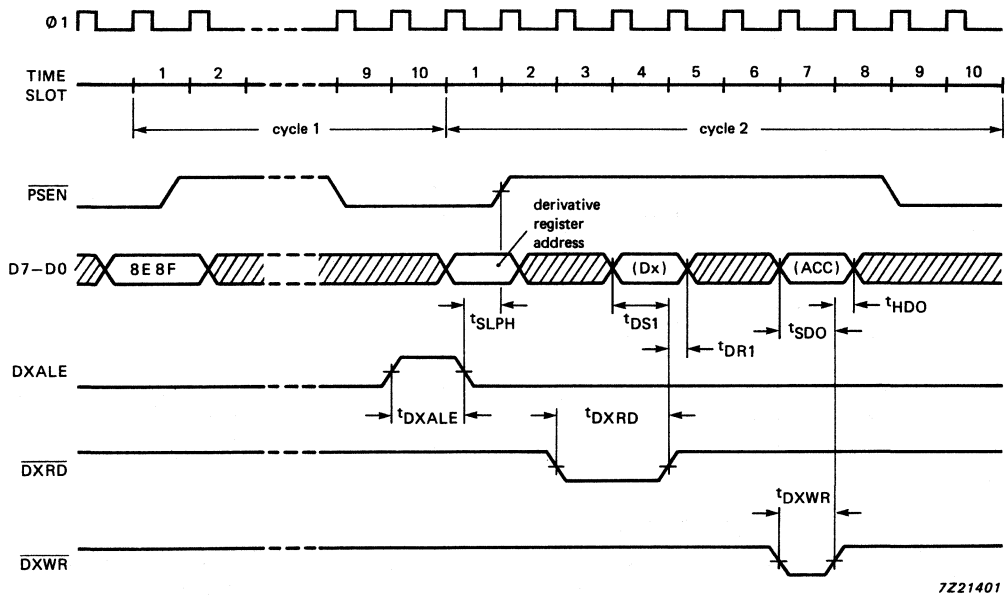


Fig. 8 ANL/ORL derivative interface timing (PCF84C00T only).

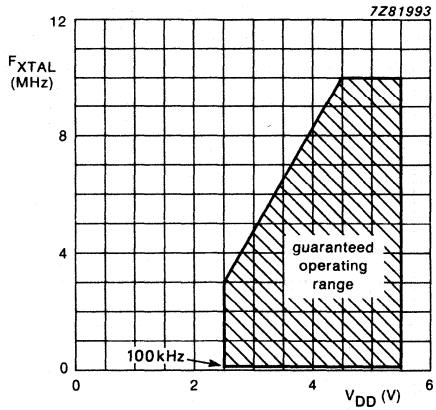


Fig. 9(a) Maximum clock frequency (f_{XTAL}) as a function of the supply voltage (PCF84C00, PCF84C21, PCF84C41 and PCF84C81).

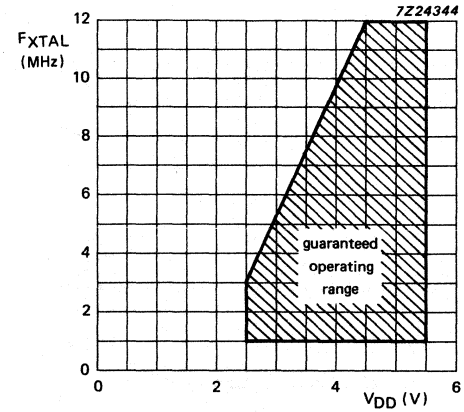
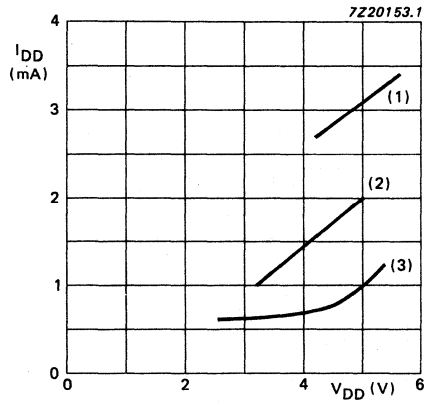
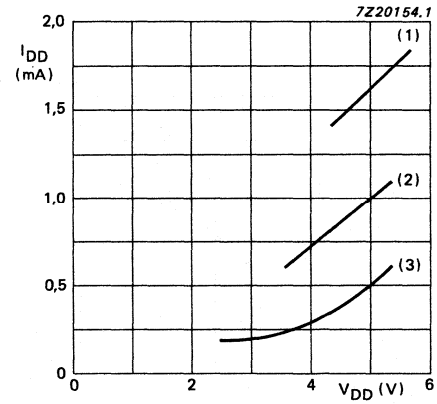


Fig. 9(b) Maximum clock frequency (f_{XTAL}) as a function of the supply voltage (PCF84C21C, PCF84C41C and PCF84C81C).



(1) clock frequency = 10 MHz
(2) clock frequency = 6 MHz
(3) clock frequency = 3,58 MHz

Fig. 10 Maximum supply current (I_{DD}) in operation mode as a function of the supply voltage (V_{DD}).



(1) clock frequency = 10 MHz
(2) clock frequency = 6 MHz
(3) clock frequency = 3,58 MHz

Fig. 11 Maximum supply current (I_{DD}) in IDLE mode as a function of the supply voltage (V_{DD}).

AC CHARACTERISTICS (continued)

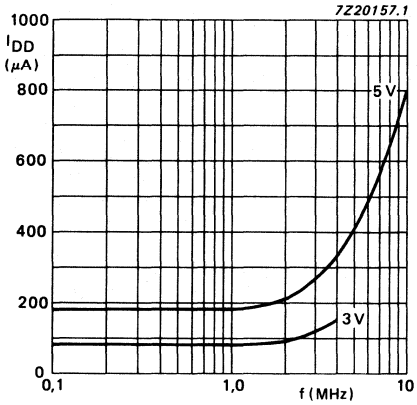


Fig. 12 Typical supply current during IDLE mode as a function of frequency at V_{DD} = 3 V and V_{DD} = 5 V.

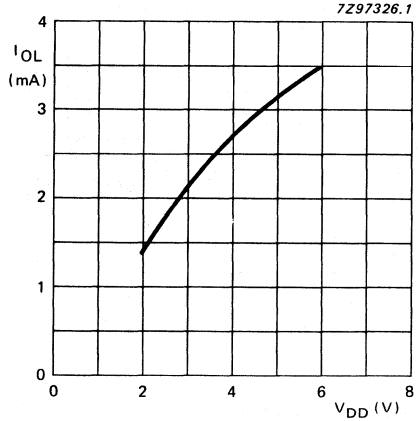


Fig. 13 Typical output sink current (I_{OL}), outputs P0.0 to P0.7 and P2.0 to P2.2, as a function of the supply voltage (V_{DD}); V_O = 0,4 V.

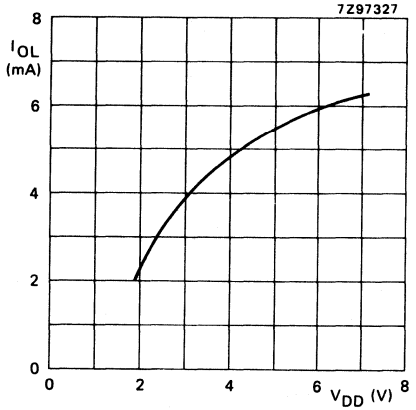


Fig. 14(a) Typical output sink current (I_{OL}), outputs P2.3/SDA and SCLK, as a function of the supply voltage (V_{DD}); V_O = 0,4 V.

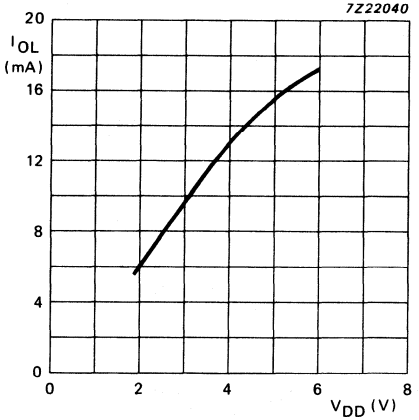
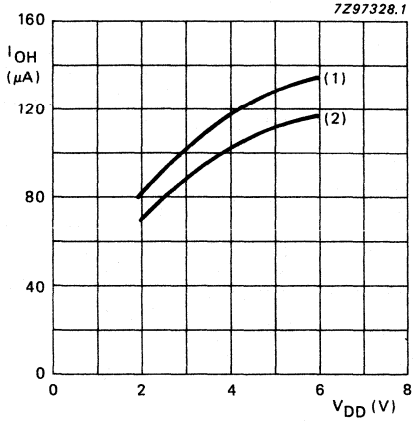


Fig. 14(b) Typical output sink current (I_{OL}), outputs P1.0 to P1.7, as a function of the supply voltage (V_{DD}); V_O = 1,2 V.



(1) $V_O = V_{SS}$
(2) $V_O = 0{,}7\ V_{DD}$

Fig. 15 Typical output source current ($-I_{OH}$) as a function of the supply voltage (V_{DD}).

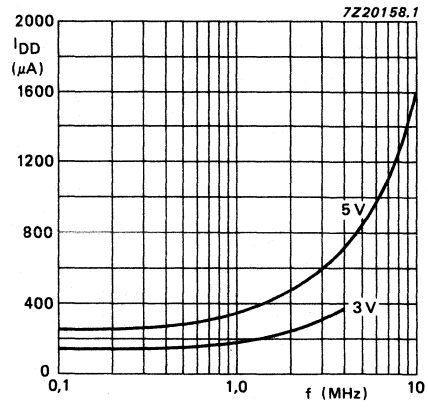
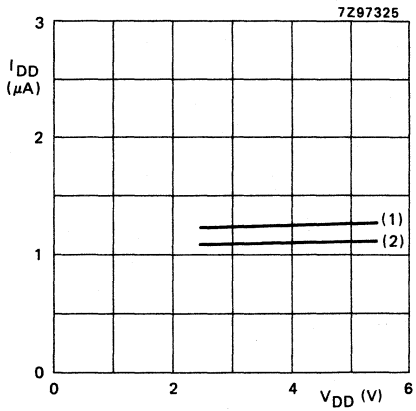


Fig. 16 Typical supply current during operating mode as a function of frequency at V_{DD} = 3 V and V_{DD} = 5 V.



(1) $T_{amb} = 85\ ^oC$
(2) $T_{amb} = 25\ ^oC$

Fig. 17 Typical supply current (I_{DD}) in STOP mode as a function of the supply voltage (V_{DD}).

Table 4 Input timing shown in Fig. 18

symbol	timing
t_{BUF}	$\geqslant 14t_{XTAL}$
$t_{HD;STA}$	$\geqslant 14t_{XTAL}$
t_{HIGH}	$\geqslant 17t_{XTAL}$
t_{LOW}	$\geqslant 17t_{XTAL}$
$t_{SU;STO}$	$\geqslant 14t_{XTAL}$
$t_{HD;DAT}$	> 0
$t_{SU;DAT}$	$\geqslant 250$ ns
t_{RD}	$\leqslant 1\ \mu s$
t_{RC}	$\leqslant 1\ \mu s$
t_{FD}	$\leqslant 1\ \mu s$
t_{FC}	$\leqslant 0{,}3\ \mu s$

Notes to Table 4

t_{XTAL} = one period of the XTAL input frequency (f_{XTAL})
= 167 ns for f_{XTAL} = 6 MHz
These figures apply to all modes.

AC CHARACTERISTICS (continued)

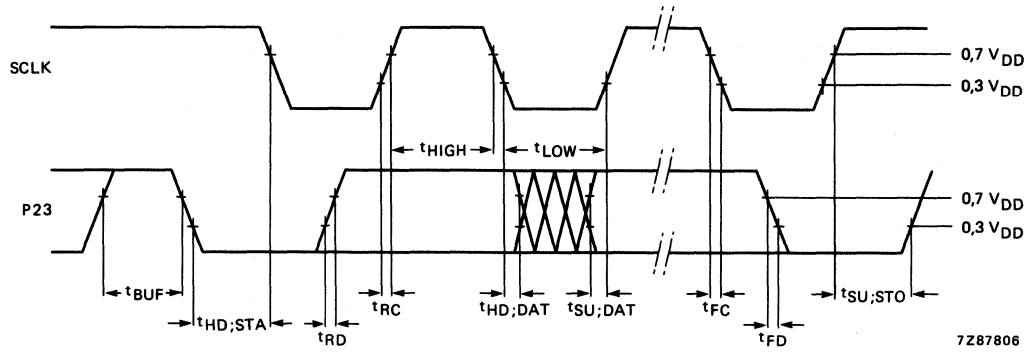


Fig. 18 Timing requirements for the P2.3 and SCLK *input* signals.

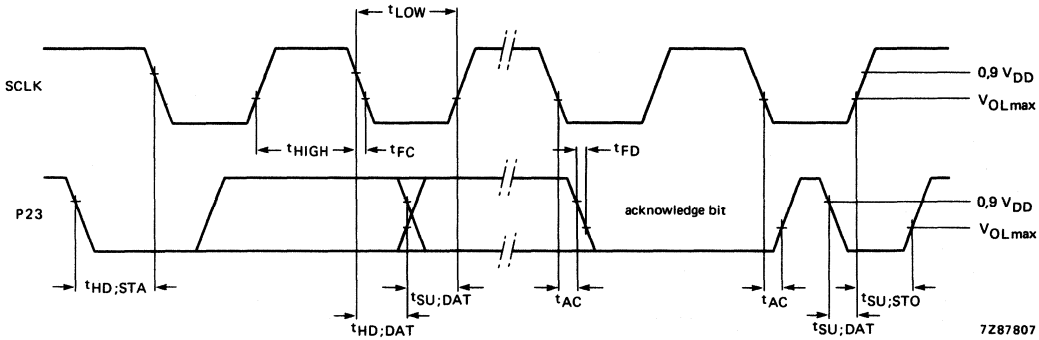


Fig. 19 Timing requirements for the P2.3 and SCLK *output* signals.

Table 5 Output timing shown in Fig. 19

symbol	timing	
	normal mode (ASC in S2 = 0)	low-speed mode (ASC in S2 = 1)
$t_{HD;STA}$	$\frac{1}{2}$ (DF + 9) t_{XTAL}	$\frac{3}{4}$ (DF + 9) t_{XTAL}
t_{HIGH}	$\frac{1}{2}$ (DF) t_{XTAL}	$\frac{3}{4}$ (DF) t_{XTAL}
t_{LOW}	$\frac{1}{2}$ (DF) t_{XTAL}	$\frac{1}{4}$ (DF) t_{XTAL}
$t_{SU;STO}$	$\frac{1}{2}$ (DF − 3) t_{XTAL}	$\frac{1}{4}$ (DF − 3) t_{XTAL}
$t_{HD;DAT}$ (slave transmitter)		
any DF	$\geqslant 9t_{XTAL}$ $\leqslant 12t_{XTAL}$	$\geqslant 9t_{XTAL}$ $\leqslant 12t_{XTAL}$
$t_{HD;DAT}$ (master transmitter)		
for DF $\leqslant$ 51	$\geqslant 9t_{XTAL}$ $\leqslant 12t_{XTAL}$	– –
for DF $\leqslant$ 99	– –	$\geqslant 9t_{XTAL}$ $\leqslant 12t_{XTAL}$
$t_{SU;DAT}$ (master transmitter)		
for DF > 51	$\geqslant 15t_{XTAL}$ $\leqslant 24t_{XTAL}$	– –
for DF > 99	– –	$\geqslant 15t_{XTAL}$ $\leqslant 24t_{XTAL}$
t_{AC}	$\geqslant 9t_{XTAL}$ $\leqslant 12t_{XTAL}$	$\geqslant 9t_{XTAL}$ $\leqslant 12t_{XTAL}$
t_{FD}, t_{FC}	$\leqslant$ 100 ns at C_b = 400 pF	$\leqslant$ 100 ns at C_b = 400 pF

Notes to Table 5

t_{XTAL} = one period of the XTAL input frequency (f_{XTAL})
= 167 ns for f_{XTAL} = 6 MHz
DF = divisor (see Table 2 Serial I/O section).
C_b = the maximum bus capacitance for each line.

AC CHARACTERISTICS (continued)

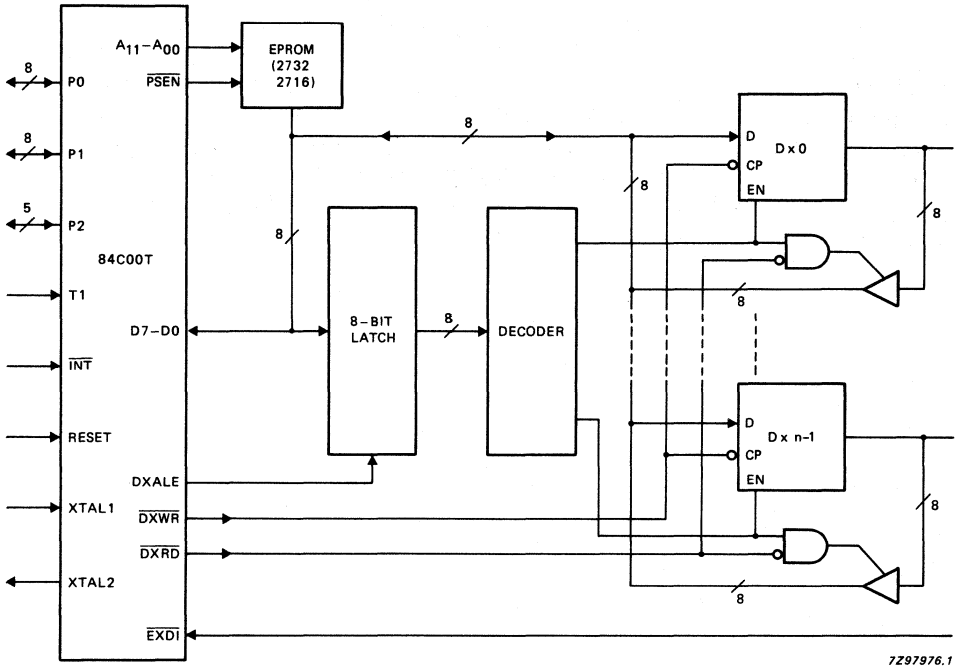


Fig. 20 Block diagram of the external Dx register interface.
The Dx interface can only be used with the PCF84C00T.

Purchase of Philips' I²C components conveys a license under the Philips' I²C patent to use the components in the I²C-system provided the system conforms to the I²C specifications defined by Philips.

SINGLE-CHIP 8-BIT MICROCONTROLLER WITH 32 I/O LINES

DESCRIPTION

The PCF84C85 microcontroller is manufactured in CMOS, and is designed to be an efficient controller as well as an arithmetic processor. The instruction set is based on that of the MAB8048 and is software compatible with the PCF84CXX family. The PCF84C85 has two additional derivative ports and the microcontroller has bit handling abilities and facilities for both binary and BCD arithmetic.

For detailed information on the PCF84CXX see the "Single-chip 8-bit Microcontrollers" user manual.

Features

- 8-bit CPU, ROM, RAM, I/O in a single 40-lead DIL or mini-pack package
- 8 K ROM
- 256 RAM bytes
- 32 quasi-bidirectional I/O port lines
- Two test inputs: one of which is also the external interrupt input
- Single-level vectored interrupts: external, timer/event counter, serial I/O
- I^2C hardware interface for two-line serial data transfer
  (serial I/O data via an existing port line and clock via a dedicated line)
- 8-bit programmable timer/event counter
- Clock frequency 100 kHz to 10 MHz
- Over 80 instructions (based on MAB8048) all of 1 or 2 cycles
- Single supply voltage from 2,5 V to 5,5 V
- STOP and IDLE mode
- Power-on-reset circuit
- Operating temperature range: −40 to +85 °C

PACKAGE OUTLINES

PCF84C85P: 40-lead DIL; plastic (SOT-129)
PCF84C85T: 40-lead; mini-pack (VSO-40; SOT-158)

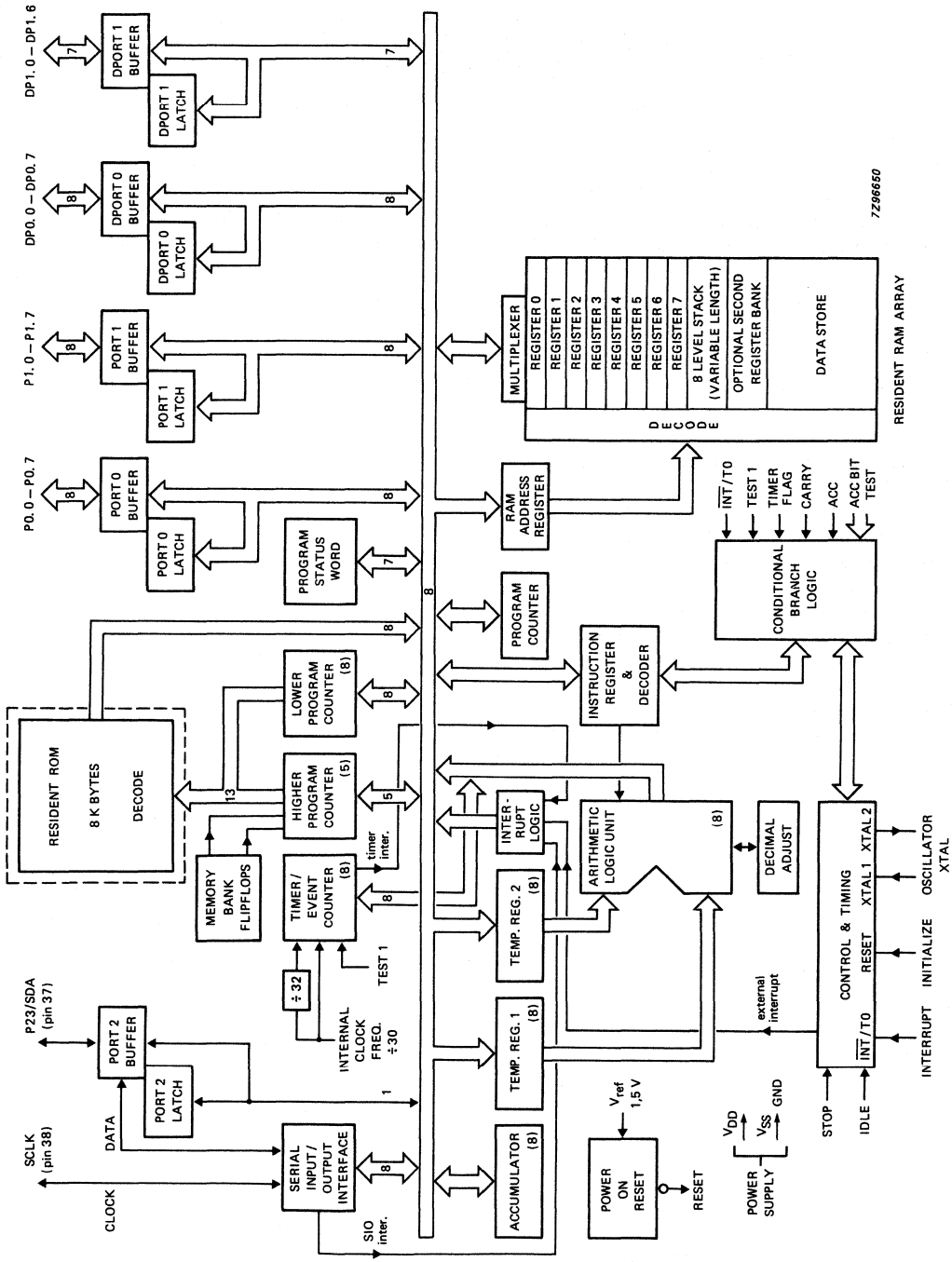


Fig. 1 Block diagram.

DEVELOPMENT DATA

PINNING

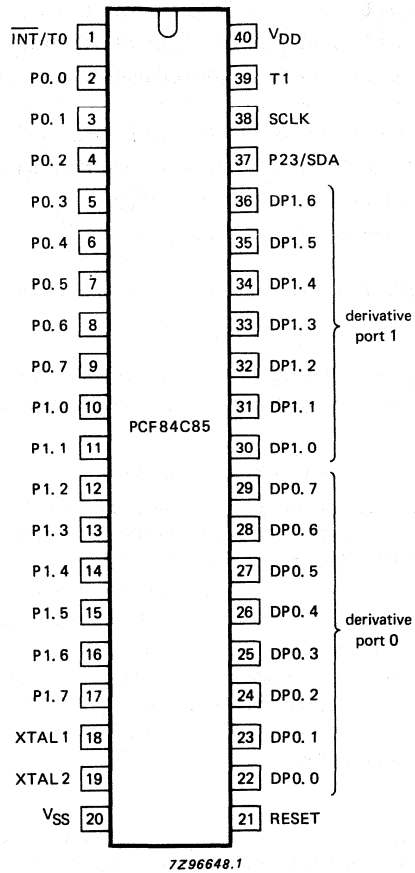


Fig. 2 Pinning diagram.

PIN DESIGNATION

38	SCLK	**Clock:** bidirectional clock for serial I/O.
1	$\overline{\text{INT}}$/T0	**Interrupt/Test 0:** external interrupt input (sensitive to negative-going edge)/test input pin; when used as a test input directly tested by conditional branch instructions JT0 and JNT0.
39	T1	**Test 1:** test input pin, directly tested by conditional branch instructions JT1 and JNT1. T1 also functions as an input to the 8-bit timer/event counter, using the STRT CNT instruction.
18	XTAL 1	**Crystal input:** connection to timing component (crystal) which determines the frequency of the internal oscillator; also the input for an external clock source.
19	XTAL 2	Connection to the other side of the timing component.
21	RESET	**Reset input:** used to initialize the processor (active HIGH), or output of power-on-reset circuit.

PIN DESIGNATION (continued)

2-9	P0.0-P0.7	**Port 0:** 8-bit quasi-bidirectional I/O port.
10-17	P1.0-P1.7	**Port 1:** 8-bit quasi-bidirectional I/O port.
37	P23/SDA	**Port 2:** 1-bit quasi-bidirectional I/O port or serial data input/output in serial I/O mode.
22-29	DP0.0-DP0.7	**Derivative port 0:** 8-bit quasi-bidirectional I/O port.
30-36	DP1.0-DP1.6	**Derivative port 1:** 7-bit quasi-bidirectional I/O port.
20	V_{SS}	**Ground:** circuit earth potentional.
40	V_{DD}	**Power supply:** 2,5 V to 5,5 V.

FUNCTIONAL DESCRIPTION

Program memory

The program memory consists of 8 K bytes, in a read-only memory (ROM). Each location is directly addressable by the program counter. The memory is mask-programmed at the factory. Figure 3 shows the program memory map.

Four program memory locations are of special importance:

- Location 0; contains the first instruction to be executed after the processor is initialized (RESET),
- Location 3; contains the first byte of an external interrupt service subroutine;
- Location 5; contains the first byte of a serial I/O interrupt service subroutine.
- Location 7; contains the first byte of a timer/event counter interrupt service subroutine.

Program memory is arranged in banks of 2 K bytes, which are selected by SEL MB instructions. The program memory is further divided into location 'pages', each of 256 bytes. This latter division applies only for conditional branches. Memory bank boundaries can be crossed only by using unconditional branch instructions after the appropriate memory bank has been selected. A CALL instruction can transfer control to a subroutine on any 'page'; RET and RETR instructions can transfer control from a subroutine back to the main program.

Data memory

Data memory consists of 256 bytes, random-access data memory (RAM). All locations are indirectly addressable using RAM pointer registers; up to 16 designated locations are directly addressable. Memory also includes an 8-level program counter stack addressed by a 3-bit stack pointer. Figure 4 shows the data memory map.

Working registers

Locations 0 to 7 are designated as working registers, directly addressable by the direct register instructions. Ease of addressing, and a minimum requirement of instruction bytes to manipulate their contents, makes these locations suitable for storing frequently addressed intermediate results. This bank of registers can be selected by the SEL RB0 instruction.

Executing the select register bank instruction SEL RB1, designates locations 24 to 31 as working registers, instead of locations 0 to 7, and these are then directly addressable. This second bank of working registers may be used as an extension of the first or reserved for use during interrupt service subroutines saving the first bank for use in the main program. If the second bank is not used, locations 24 to 31 may serve as general purpose RAM.

The first locations of each bank contain the RAM pointer registers R0, R1, R0' and R1', which indirectly address all RAM locations.

All RAM locations make efficient program loop counters when used with the decrement register and test instruction DJNZ.

DEVELOPMENT DATA

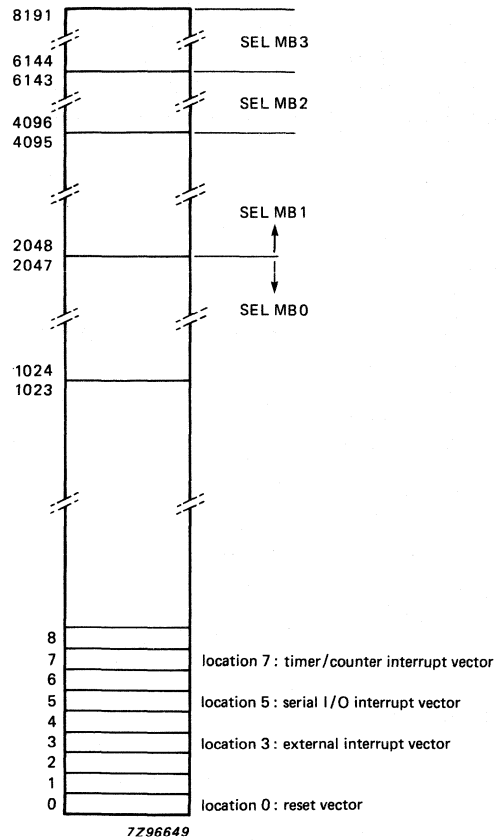


Fig. 3 Program memory map.

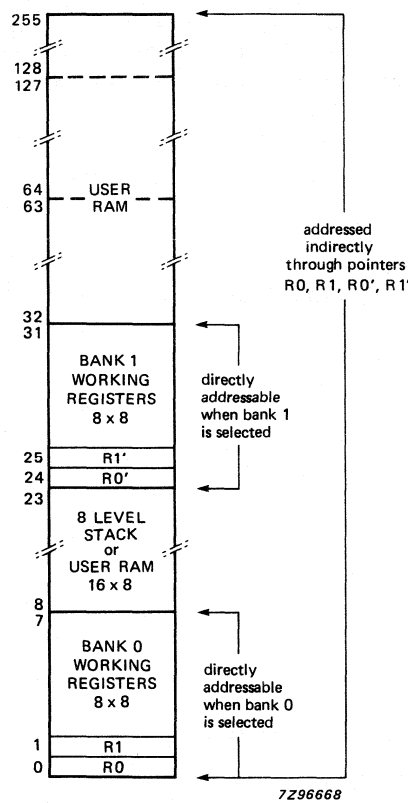


Fig. 4 Data memory map.

Program counter stack

Locations 8 to 23 may be designated as an 8-level program counter stack (2 locations per level), or as general purpose RAM. The program counter stack (Fig.5) enables the processor to keep track of the return addresses and status generated by interrupts or CALL instructions by storing the contents of the program counter prior to servicing the subroutine. A 3-bit stack pointer determines which of the eight register pairs of the program counter stack will be loaded with the next generated return address.

The stack pointer, when initialized to 000 by RESET, points to RAM locations 8 and 9. On the first subroutine CALL or interrupt, the contents of the program counter and bits 4, 6 and 7 of the program status word (PSW) are transferred to locations 8 and 9. The stack pointer increments by one and points to locations 10 and 11 ready for another CALL. Because an address may be up to 13 bits long, two bytes must be used to store each address.

At the end of a subroutine, which is signalled by a return instruction (RET or RETR), the stack pointer decrements by one and the contents of the register pair on top of the stack are transferred to the program counter. The saved PSW bits are transferred to the PSW only by the RETR instruction.

If not all 8 levels of subroutine and interrupt nesting are used, the unused portion of the stack may be used as any other indirectly addressable RAM locations.

Nesting of subroutines within subroutines can continue up to 8 times without overflowing the stack. If overflow does occur the deepest address stored (locations 8 and 9) will be overwritten and lost since the stack pointer overflows from 111 to 000. It also underflows from 000 to 111.

FUNCTIONAL DESCRIPTION (continued)

The value of the saved contents of the program counter is different for an interrupt CALL compared to a normal CALL to subroutine. With an interrupt CALL, the program counter return address is saved; with a subroutine CALL, the saved program counter value is one less than the program counter return address.

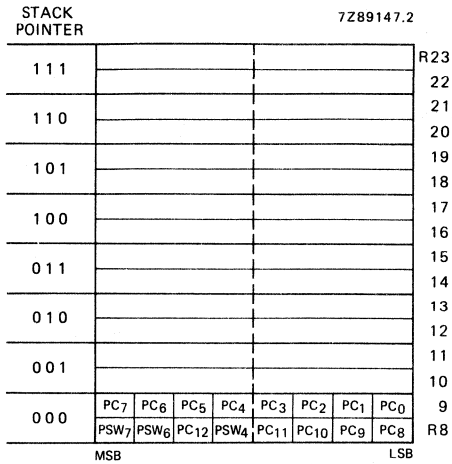


Fig. 5 Program counter stack.

IDLE and STOP modes

IDLE mode

When the microcontroller enters the IDLE mode via the IDLE instruction (01 H) the oscillator, timer/counter and serial I/O are kept running. The microcontroller exits from the IDLE mode by one of three interrupts if they are enabled or by activating a RESET. If the interrupt is not enabled the processor will remain in the IDLE mode. An active signal on the RESET pin restarts the microcontroller and a normal RESET sequence is executed (see Fig. 6).

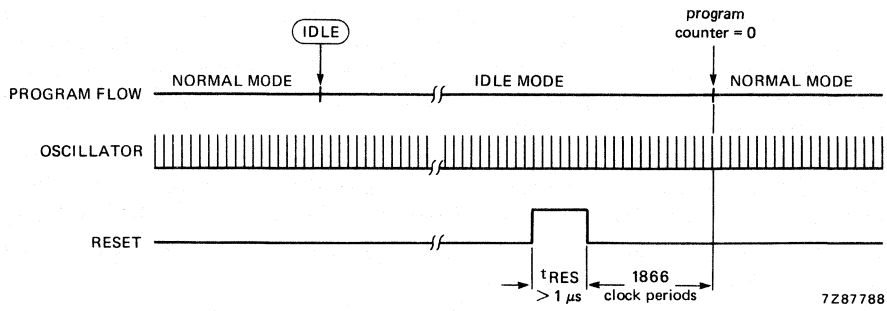


Fig. 6 Exit from IDLE mode via a RESET.

An active signal coming from an enabled interrupt causes the execution of the normal interrupt routine since normal interrupt scanning is still being carried out. A HIGH-to-LOW transition on the external interrupt pin ($\overline{INT}$/T0) reactivates the microcontroller. A LOW level applied to $\overline{INT}$/T0 will reactivate the microcontroller only in the STOP mode. Thus,if $\overline{INT}$/T0 was LOW before the microcontroller entered the IDLE mode, it must go HIGH before the microcontroller can be reactivated (see Fig. 7).

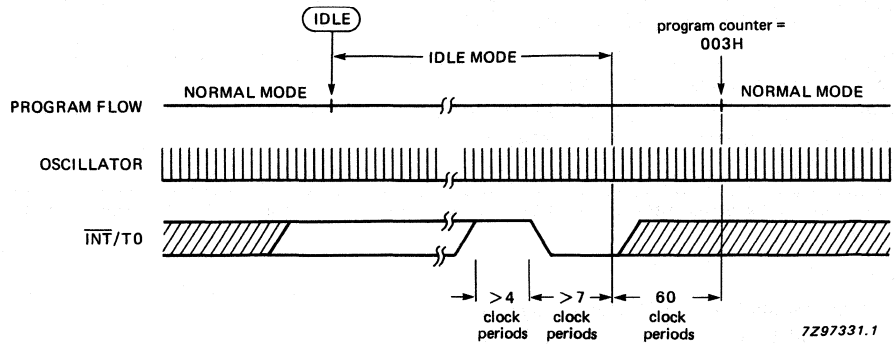


Fig. 7 Exit from IDLE mode via an interrupt.

Wake-up from the IDLE mode is ensured when $\overline{INT}$/T0 is HIGH for at least 4 CP (clock periods) followed by a LOW for 7 CP. After the initial forced CALL (*4H) operation (60 CP) the program continues with the external interrupt service routine.

* 1, 3, 5, 7, 9, B, D and F.

STOP mode

The microcontroller enters the STOP mode by the STOP instruction (22 H). The oscillator is switched off. The internal status of the CPU, RAM contents and the state of I/O ports are not affected. The microcontroller can be brought-out of the STOP mode by an active signal at the external interrupt input or by an external RESET signal. When one of these two signals is applied an internal delay of 1866 CP is provided to ensure that all internal clocks are operating correctly before restart (see Fig. 8). Note; the start-up time of a crystal oscillator is measured in milliseconds, and the 1866 CP count begins after this start-up time.

If the microcontroller exits from the STOP mode by activating RESET, a normal RESET sequence is executed.

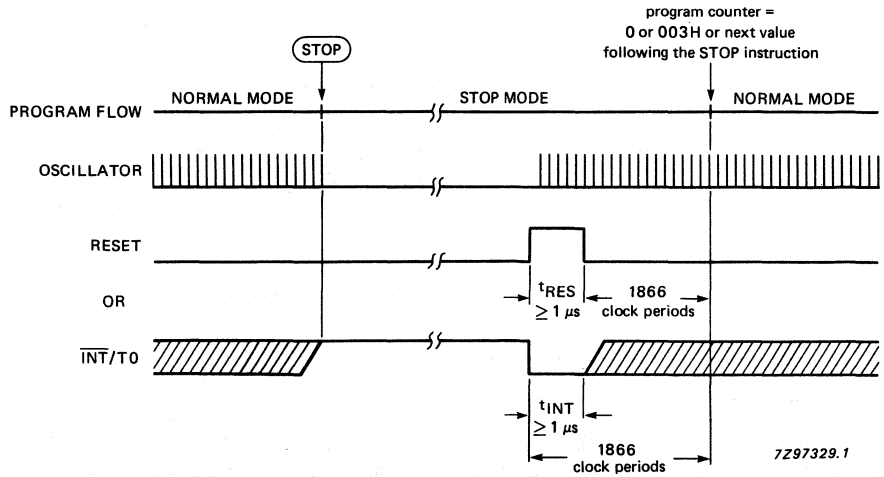


Fig. 8 Entering and exiting the STOP mode.

FUNCTIONAL DESCRIPTION (continued)

If the microcontroller exits the STOP mode by pulling the external interrupt input pin LOW, an interrupt sequence is executed only if the external interrupt is enabled. In this event the microcontroller resumes the normal program sequence after returning from the interrupt routine, as in the normal mode. If the interrupt is not enabled, it continues the normal program sequence, executing the instruction following the STOP instruction.

The microcontroller is restarted by a LOW level applied at the $\overline{INT}$/T0 pin, and not by a HIGH-to-LOW transition as in a normal interrupt mechanism.
Note: when leaving the STOP mode with an interrupt, a further instruction in the main program series is executed prior to entering the interrupt routine.

When the $\overline{INT}$/T0 level is active during the STOP instruction then no STOP is executed.

A LOW level on the external interrupt input of at least 1 μs will cause the microcontroller to exit the STOP mode.

I/O facilities

The PCF84C85 family has 32 I/O lines arranged as:

- Port 0 parallel port of 8 lines (P0.0 to P0.7)
- Port 1 parallel port of 8 lines (P1.0 to P1.7)
- Port 2 parallel port of 1 line (P2.3)
- D port 0 parallel port of 8 lines (DP0.0-DP.7)
- D port 1 parallel port of 7 lines (DP1.0-DP1.6)

In addition to these the PCF84C43 also comprises four specialized I/O lines:

- SCLK I^2C-bus serial clock line
- SDA I^2C-bus serial data line (shared with P2.3)
- $\overline{INT}$/T0 external interrupt and test input. When used as a test input T0 can be directly tested by conditional branch instructions JT0 and JNT0
- T1 test input which can alter program sequences when tested by conditional jump instructions JT1 and JTN1. T1 also functions as an input to the 8-bit timer/event counter.

Parallel ports

All parallel ports can be used as outputs or inputs, their structure is quasi-bidirectional.
Output data written to a port is latched and remains unchanged until rewritten.
Input data is not latched and so must be present until read by an input instruction.
Input lines are fully CMOS compatible, output lines can drive one TTL or CMOS load.

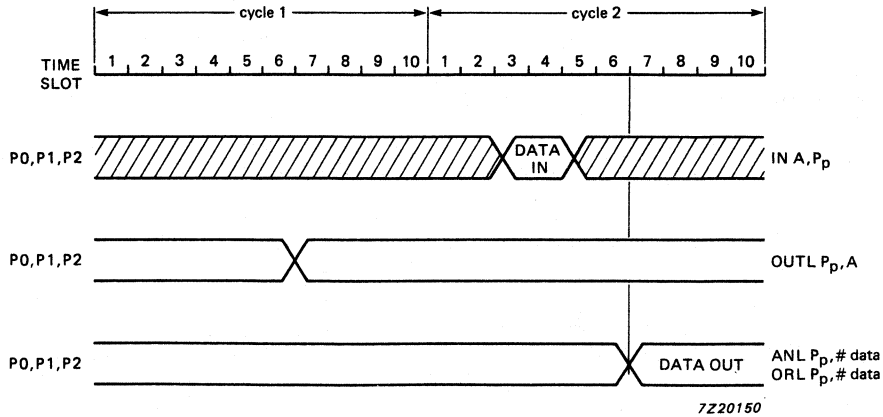


Fig. 9 Shows the timing diagram for all ports using IN, OUTL, ANL and ORL instructions. For the OUTL instruction data changes on time slot 7 of cycle 1. For the MOV, ANL and ORL instructions, the ports change on time slot 7 of cycle 2.

Fig. 10 shows the quasi-bidirectional I/O interface with push-pull output and switched pull-up current source. Each line is pulled up to V_{DD} via a constant current source (TR4), which is enabled via TR3 whenever one of the two output latches contains a logic 1. This current source is sufficient for a TTL HIGH level, yet can be pulled LOW by an external CMOS device, thus allowing the same pin to be used for both input and output.

When a logic 1 is written to the line for the first time (MQ = 1, SQ = 0), TR2 is switched on for the duration of the internal write pulse (one oscillator period), to provide a fast transition from logic 0 to logic 1. Subsequent writing of a logic 1 to the port lines will not switch TR2 on. This prevents unnecessary current through external components connected to the port lines of the same port which might be in the input mode and also connected to ground.

When a logic 0 is written to the line, TR3 switches off the current source. Current sinking capability is provided by TR1, which is now switched on. When used as an input, a logic 1 must first be written to the line, otherwise TR1 will remain low impedance.

The PCF84C85 family offers the possibility to select individually 31 of the 32 parallel port pins (not P23), by the following mask options:

Option 1 – STANDARD PORT; quasi-bidirectional I/O with switched pull-up current source of 100 μA (typ.) and P-channel booster transistor TR2.
TR2 is only active during 1 clock cycle (Fig. 10).

Option 2 – OPEN DRAIN; quasi-bidirectional I/O with only an N-channel open drain output. Application as an output requires connection of an external pull-up resistor (Fig. 11).

Option 3 – PUSH-PULL OUTPUT; drive capability of the output will be 1,6 mA (min.) at V_{DD} = 5 V in both polarities. To avoid a large current flowing through the output transistors during the input mode, these push-pull pins must only be used as outputs (Fig. 12).

DEVELOPMENT DATA

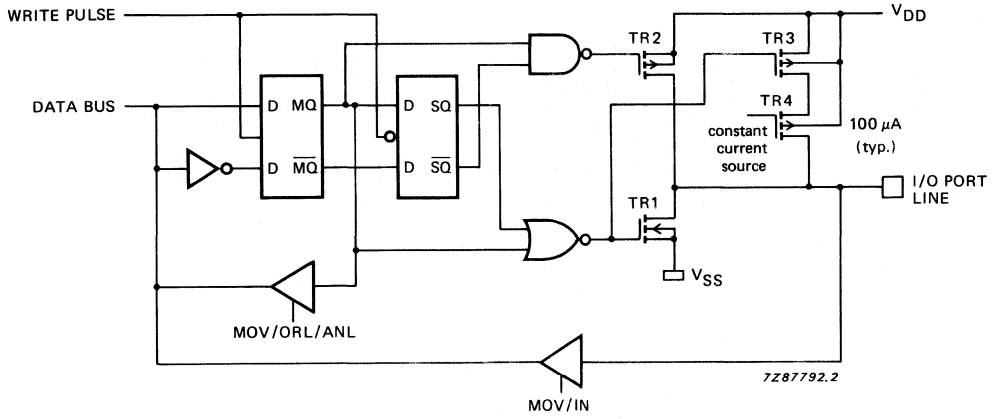


Fig. 10 Standard output with switched pull-up current source.

FUNCTIONAL DESCRIPTION (continued)

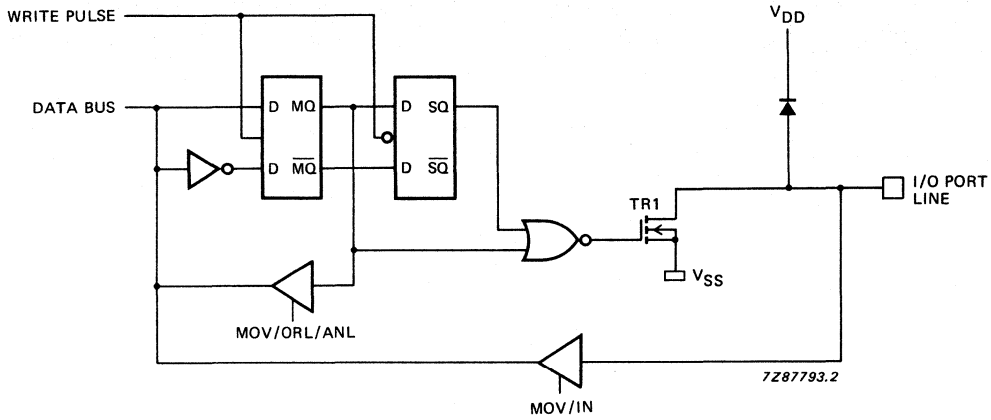


Fig. 11 Open drain output.

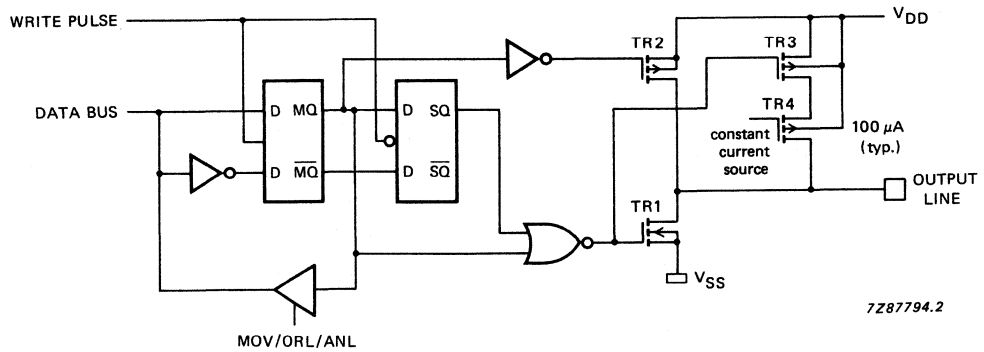


Fig. 12 Push-pull output.

Serial I/O (SIO)

The PCF84C85 has an on-chip serial I/O interface.

Whereas a normal microcontroller must regularly monitor the serial data bus for the presence of data, the serial I/O interface detects, receives and converts the serial data stream into parallel format without interrupting the execution of the current program. An interrupt is sent to the PCF84C85 only when a complete byte is received. It then reads the data byte in one instruction. Likewise during transmission the serial I/O interface performs parallel to serial conversion and subsequent serial output of the data. The microcontroller is only interrupted in the execution of its programmed tasks when a complete byte has been transmitted.

The design of the PCF84C85 serial I/O system allows any number of devices from PCF85XX family (clips) to be connected via the two-line serial bus. The ability of any devices to communicate, without interrupting the operation of any other devices on the bus, is an outstanding attribute of the system. This is achieved by allocating a specific 7-bit address to each device and providing a system whereby a device reacts only to a message prefixed with its own address or the 'general CALL' address. Address recognition is performed by the interface hardware so that operation of the microcontroller need only be interrupted when a valid address has been received. This saves significant processing time and memory space compared with a conventional microcontroller employing a software serial interface. When the addressing facility is not required, for instance in a system with only two microcontrollers, direct data transfer without addressing can be performed. In multi-master systems, an automatically invoked arbitration procedure prevents two or more devices from continuing simultaneous transmission.

In NORMAL (running) and IDLE mode, the serial I/O logic remains active; its internal system clock will be switched off when there is no activity on the serial bus.

After execution of the STOP instruction, the oscillator of the PCF84C85 is switched off. This means that the serial I/O logic will remain in the state it was at the occurance of the STOP instruction. To avoid "bus block" problems and to assure correct start-up of the bus after exit from the STOP mode, the user should disable the serial logic (ESO = 0) prior to the execution of the STOP instruction. This must be carried out only when the PCF84C85 has finished a serial data transfer.

Serial I/O interface

Figure 13 shows the serial I/O interface. The clock line of the serial bus has exclusive use of pin 38 (SCLK) while the data line shares pin 37 (serial data) with the I/O line P23 of port 2. When the serial I/O is enabled, P23 is disabled as a parallel port line; (P23 and SCLK only open drain).

The microcontroller and interface communicate via the internal microcontroller bus and the Serial Interrupt Request line. Data and information controlling the operation of the interface are stored in four registers:

- Data shift register (S0)
- Serial I/O interface status word (S1)
- Serial clock control word (S2)
- Address register

FUNCTIONAL DESCRIPTION (continued)

Data shift register (S0)

Register S0 converts serial data to parallel format and vice versa. A pending interrupt is generated only after a complete byte has been transmitted, or after a complete data byte, specific address or 'general CALL' address has been received. The most significant bit is transmitted first.

Serial I/O interface status word (S1)

Register S1 provides information concerning the state of the interface and stores information from the microcontroller. Bits 0 to 3 are duplicated: control bits in these positions can only be written by the microcontroller, while interface bits can only be read.

MST and TRX (see Table 1)

These bits determine the operating mode of the serial I/O interface.

Table 1 Operating modes of the serial I/O interface

MST	TRX	operating mode
0	0	slave receiver
1	0	master receiver
0	1	slave transmitter
1	1	master transmitter

BB: Bus Busy.

This is the flag which indicates the status of the bus.

PIN: Pending Interrupt Not

PIN = '0' indicates the presence of a pending interrupt, which will cause a Serial Interrupt Request when the serial interrupt mechanism is enabled.

ESO: Enable Serial output

The ESO flag enables/disables the serial I/O interface: ESO = '1' enables, ESO = '0' disables. ESO can only be written by software.

BC0, BC1 and BC2

Bits BC0, BC1 and BC2 indicate the number of bits received or transmitted in a data stream. These bits can only be written by software.

AL: Arbitration Lost

The arbitration lost flag is set by hardware when the serial I/O interface, as master transmitter, loses a bus arbitration procedure.

AAS: Addressed As Slave

This flag is set by hardware when the interface detects either its own specific address or the 'general CALL' address as the first byte of a transfer and the interface has been programmed to operate in the address recognition mode.

AD0: Address Zero

This flag is set by hardware after detection of the 'general CALL' address when the interface is operating in the address recognition mode.

LRB: Last Received Bit

This contains either the last data bit received or, for a transmitting device in the acknowledgement mode, the acknowledgement signal from the receiving device.

Bits AL, AAS, AD0 and LRB can only be read by software.

Serial clock control word (S2)

Bits 0 to 4 of the clock control register S2 are used to set the frequency of the serial clock signal. When a 6 MHz crystal is used, the frequency of the serial clock can be varied between 154 kHz and 1 kHz (see Table 2). An asymmetrical clock with a HIGH-to-LOW ratio of 3 : 1 can be generated using bit 5. The asymmetrical clock allows a microcontroller more time per clock period for sampling the data line, making the timing of this action less critical. Bit 6 can be used to activate the acknowledge mode of the serial I/O. S2 is a write only register.

Address register

The address register contains the 7-bit address back-up latches and the bit (ALS) used to enable/disable the address recognition mode. The address register can be written using the MOV S0, A and MOV S0, # data instructions, but only when ES0 = '0'.

Serial I/O interrupt logic

An EN SI instruction enables and a DIS SI instruction disables the interrupt logic. When the logic is enabled, a pending interrupt results in a serial I/O interrupt to the processor, causing a CALL to location 5 in the ROM. When disabled, the presence of an interrupt is still indicated by PIN in S1, allowing the interrupt to be serviced. However, vectored interrupt will not occur.

FUNCTIONAL DESCRIPTION (continued)

Table 2 SIO clock pulse frequency control when using 6 MHz crystal

hexadecimal S20-S24 code	divisor	f_{XTAL} (6 MHz) f_{SCLK} (kHz)	f_{XTAL} (10 MHz) f_{SCLK} (kHz)
0	not allowed		
1	39	*154	*256
2	45	*133	*222
3	51	*118	*196
4	63	95	*159
5	75	80	*133
6	87	69	*115
7	99	61	*101
8	123	49	81
9	147	41	68
A	171	35	58
B	195	31	51
C	243	25	41
D	291	21	34
E	339	18	29
F	387	16	26
10	483	12	21
11	579	10	17
12	675	8,9	15
13	771	7,8	13,4
14	963	6,2	10,4
15	1155	5,2	8,7
16	1347	4,5	7,4
17	1539	3,9	6,5
18	1923	3,1	5,2
19	2307	2,6	4,3
1A	2691	2,2	3,7
1B	3075	2,0	3,3
1C	3843	1,6	2,6
1D	4611	1,3	2,2
1E	5379	1,1	1,9
1F	6147	1,0	1,6

* Not permitted for I^2C operation; the maximum clock frequency in the I^2C systems is 100 kHz.

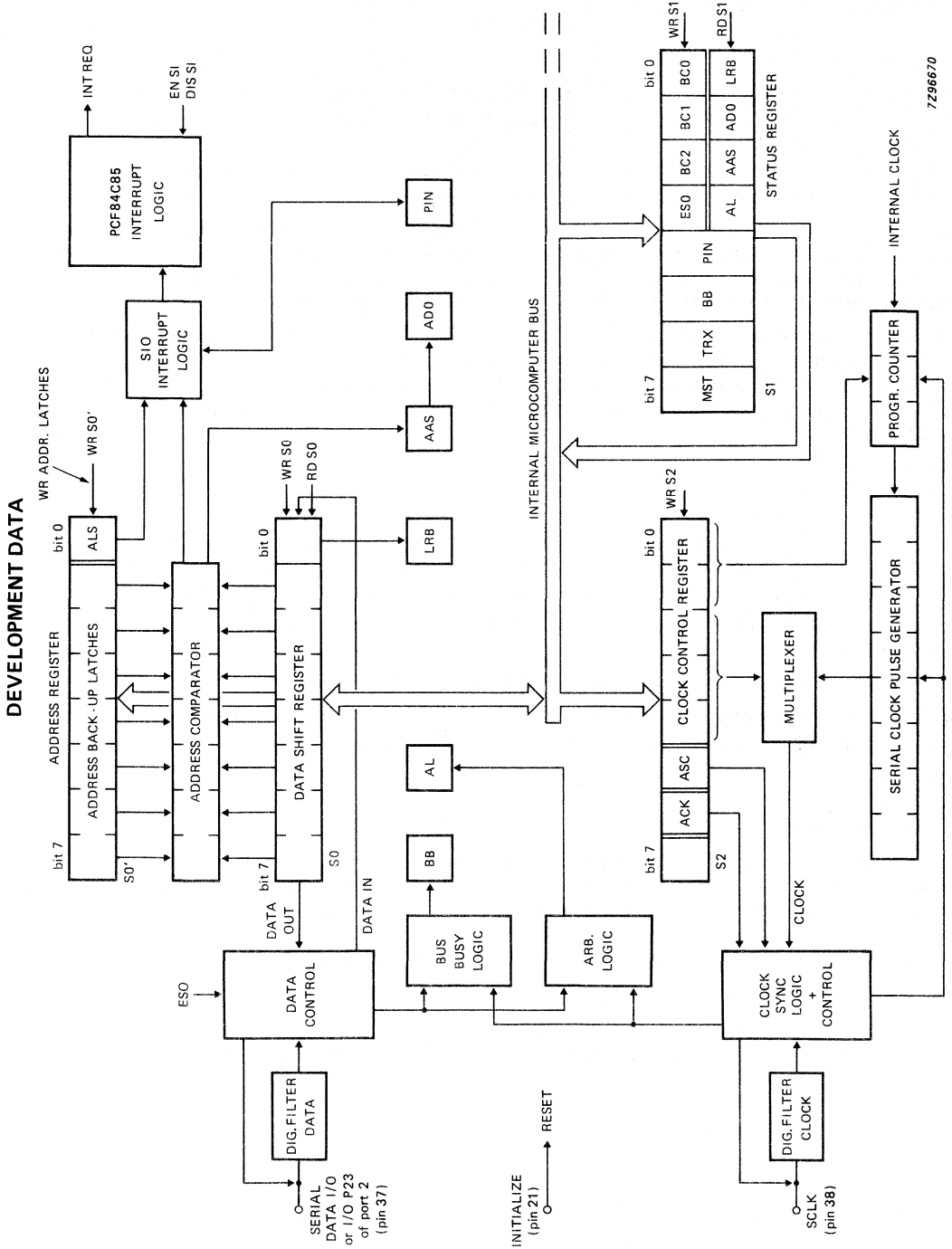


Fig. 13 Serial I/O interface.

FUNCTIONAL DESCRIPTION (continued)

Interrupts (see Fig. 14)

When the external interrupt is enabled, a HIGH-to-LOW transition on the $\overline{INT}$/T0 input initiates an external interrupt subroutine which causes a CALL to program memory location 3 following completion of the current instruction.

Serial I/O interrupt, when enabled, causes a CALL to location 5, and a timer/event counter overflow forces a CALL to location 7 when the timer interrupt is enabled.

When an interrupt subroutine starts, the contents of the program counter and bits 4, 6 and 7 of the PSW have been saved in the program counter stack. Accumulator contents have to be saved by software. Interrupt acknowledgement can be carried out by software via port pins. All interrupt subroutines must reside in memory bank 0.

Since the interrupt system is single level, once an interrupt is detected, all further interrupt requests are latched, but ignored, pending a RETR instruction to re-enable the interrupt input logic. After executing RETR, the program continues in the main part; this is independent of the occurance of a second interrupt during the running of the first routine. If 2 or 3 interrupts occur simultaneously, their priority is:

- (1) external
- (2) serial I/O
- (3) timer/event counter

An external interrupt can be generated by using the timer/counter in the event counter mode. The counter is first preset to (FFH), then EN TCNTI instruction is executed. A LOW-to-HIGH transition of the T1 input will then initiate an interrupt subroutine and cause a CALL to location 7.

On execution of a DIS I instruction, the PCF84C85 always clears the digital filter/latch and the External Interrupt Flag.

The Timer Flag (TF) is reset only when the JTF or JNTF instruction is executed or after RESET.

The Timer Interrupt Flag is set when timer overflow occurs, only if the timer interrupt is enabled.

The microcontroller will exit the IDLE mode when any one of the following three interrupts is enabled:

- External
- Serial I/O
- Timer/event counter

There is no internal pull-up or pull-down device connected to the external interrupt input (pin 1). If required pin 1 must be externally connected to a resistor ($R \leqslant 100\ k\Omega$). When the external interrupt is not used pin 1 must be connected to V_{DD}.

Improvements to interrupt and timer logic with respect to the MAB8400 family

For detailed information see the user manual "Single-chip 8-bit Microcontrollers".

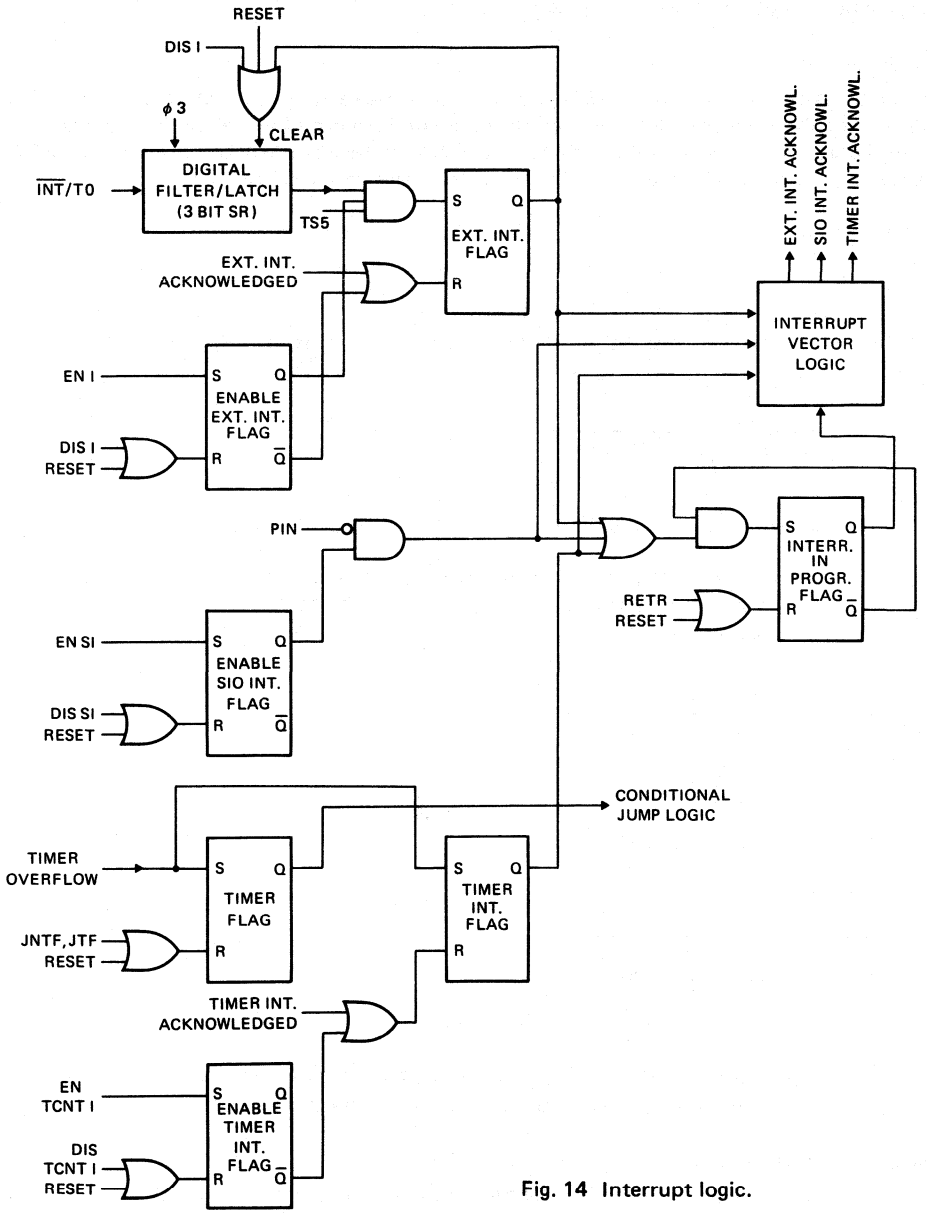


Fig. 14 Interrupt logic.

Notes to figure 14

1. $\overline{INT}$/T0 negative edge is always latched in the digital filter/latch.
2. Correct interrupt timing is ensured when INT/T0 is HIGH for > 4 CP followed by a LOW for > 7 CP.
3. When the interrupt in progress flag is set, further interrupts are latched but ignored, until RETR is executed.
4. A DIS I instruction always clears a pending external interrupt.
5. For all flip-flops, RESET overrules SET.

FUNCTIONAL DESCRIPTION (continued)

Oscillator (see Fig. 15)

The oscillator can be inhibited by the STOP instruction under software control. It is also inhibited when a low-supply voltage condition is present to prevent discharge of a weak back-up battery. Provided the supply voltage is within the operating range the oscillator will be restarted after a STOP instruction by a LOW level at the $\overline{INT}$/T0 pin or a HIGH level at the RESET pin.

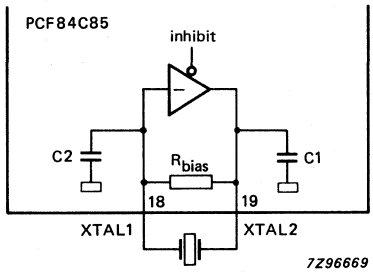


Fig. 15 Oscillator with integrated elements.

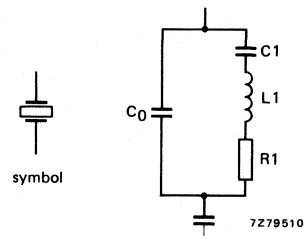


Fig. 16 Crystal unit equivalent circuit.

The values of crystal series resistance R1 and the crystal's total load capacitance C_L (C_0 + wiring + external capacitors) must not be above the curve (Fig. 17) for the corresponding frequency.
Note; if external capacitors are connected to XTAL 1 and XTAL2 they must be of equal value.

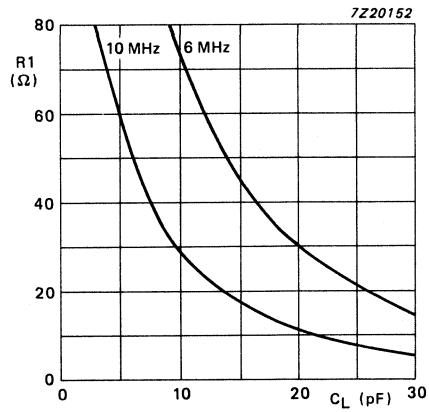


Fig. 17 Crystal circuit criteria.

The oscillator has the output drive capability via pin 19 (XTAL2). An external clock can be applied to pin 18 (XTAL1). A machine cycle consists of 10 time slots, each time slot being 3 oscillator periods.

Timer/event counter (see Fig. 18)

An internal 8-bit up-counter is provided. This can count external events, modulo-32 machine cycles, or machine cycles directly. Table 3 gives the instructions that control the counter and the prescaler, and the functions performed.

When used as a timer, the input to the counter is either the overflow or input of a 5-bit prescaler. When used as an event counter, LOW-to-HIGH transitions on pin 39 (T1) are counted. The maximum rate at which the counter may be incremented is once every machine cycle. When the counter overflows, the timer flag is set. The flag can be tested and reset using the JTF (jump is timer flag = 1) or JNTF instruction. Overflow also generates an interrupt to the processor via setting of the Timer Interrupt Flag when the timer/event counter interrupt is enabled.

Table 3 Timer/event counter control

function	timer mode modulo-1, modulo-32*	counter mode
CLEAR	MOV T,A (A) = 0 or RESET	MOV T,A (A) = 0 or RESET
PRESET	MOV T,A	MOV T,A
START	STRT T	STRT CNT
STOP	STOP TCNT or RESET	STOP TCNT or RESET
TEST	JTF/JNTF	JTF/JNTF
READ**	MOV A,T	MOV A,T

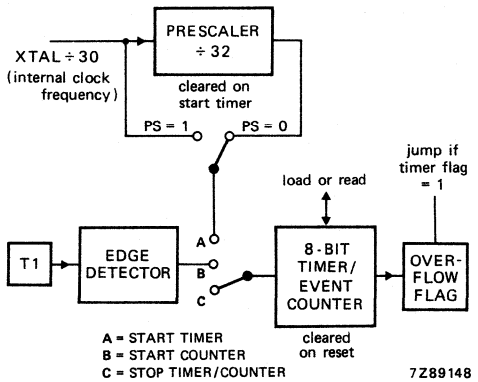


Fig. 18 Timer/event counter.

DEVELOPMENT DATA

Program status word (see Fig. 19)

The program status word (PSW) is an 8-bit word (1 byte) in the CPU which stores information about the current status of the microcontroller.
The PSW bits are:

- Bits 0 to 2 — stack pointer bits (SP_0, SP_1, SP_2)
- Bit 3 — prescaler select (PS);
  0 = modulo-32; 1 = modulo-1 (no prescaling)
- Bit 4 — working register bank select (RBS);
  0 = register bank 0; 1 = register bank 1
- Bit 5 — not used (1)
- Bit 6 — auxiliary carry (AC); half-carry bit generated by an ADD instruction and used by the decimal adjust instruction DA A
- Bit 7 — carry (CY); the carry flag indicates that previous operation has resulted in an overflow of the accumulator.

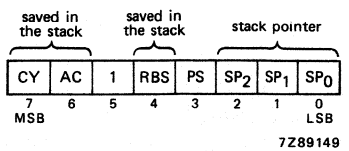


Fig. 19 Program status word.

* With prescaler select, PS = 0, the timer counts modulo-32 machine cycles, with PS = 1 it counts modulo-1 cycles (prescaler not used); prescaler cleared with STRT T, prescaler not readable.
** READ does not disturb the counting process.

FUNCTIONAL DESCRIPTION (continued)

Program status word (continued)

All bits can be read using the MOV A, PSW instruction. Bits 7 and 6 are set and cleared by CPU operation. Bit 4 can be changed by a SEL RB instruction, bit 3 by the MOV PSW, A instruction, and bits 0, 1 and 2 by the CALL, RET or RETR instructions in the event of an interrupt. Bits 7, 6 and 4 are stored in the program counter stack during subroutine and interrupt calls. These bits are restored in the PSW with a RETR (return and restore) instruction which must be used at the end of an interrupt and can be used at the end of a normal subroutine. The RET instruction has no restore feature and cannot be used at the end of an interrupt.

Program counter (see Fig. 20)

The 13-bit program counter is able to address 8 K bytes of ROM. The arrangement of the bits is shown in figure 20. During an interrupt subroutine PC_{11} and PC_{12} are forced to logic 0. All 13 bits are saved in the stack during CALL and interrupt routines.

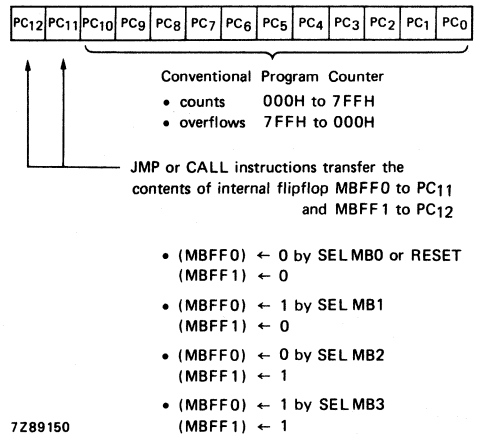


Fig. 20 Program counter.

Central processing unit

The PCF84C85 has arithmetic, logical and branching capabilities. The DA A, SWAP A and XCHD instructions simplify BCD arithmetic and the handling of nibbles. The MOVP A,@A instruction permits efficient table look-up from the current ROM page.

Conditional branch logic

The conditional branch logic within the processor enables several conditions, internal and external to the processor, to be tested by the user's program. Table 4 lists the conditional jump instructions used to change the program sequence. The DJNZ instruction decrements a designated register or data memory location and branches if the contents are not zero. This instruction is useful for looping control. The JMPP@A instruction allows multiway branches to destinations determined by the contents of the accumulator.

Table 4 Conditional branches

test	jump condition	jump instruction
accumulator	all bits zero	JZ
	any bit non-zero	JNZ
accumulator bit test	1	JB0 to JB7
carry flag	1	JC
	0	JNC
timer overflow flag	1	JTF
	0	JNTF
test input T0	1	JT0
	0	JNT0
test input T1	1	JT1
	0	JNT1
register	non-zero	DJNZ

Test input T1 (pin 39)

The T1 input line can be used as:

- A test input for branch instructions JT1 and JNT1
- An external input to the event counter

When used as a test input:

- JT1 instruction tests for logic 1 level
- JNT1 instruction tests for logic 0 level

When used as an input to the event counter, T1 must be LOW for > 4 CP, followed by a HIGH for > 4 CP. The transition can be recognized with a repetition rate of once per 30 oscillator clock periods (1 machine cycle).

There is no internal pull-up or pull-down resistor connected to the T1 input. If required it must be externally connected to a resistor ($R = \leqslant 100\ k\Omega$). When T1 is not used pin 39 must be connected to V_{DD} or V_{SS}.

Reset (pin 21)

A positive-going signal on the RESET input

- Sets the program counter to zero
- Selects location 0 of memory bank 0 and register bank 0
- Sets the stack pointer to zero (000); pointing to RAM address 8
- Disables the interrupts (external, timer and serial I/O)
- Stops the timer/event counter, then sets it to zero
- Sets the timer prescaler to modulo-32
- Resets the timer flag
- Sets all ports to input mode
- Sets the serial I/O to slave receiver mode and disables the serial I/O
- Cancels IDLE and STOP mode

FUNCTIONAL DESCRIPTION (continued)

Power-on-reset

The internal power-on reset circuit monitors the PCF84C85 supply voltage V_{DD}. For as long as the supply voltage remains below the internal reference level V_{ref} (typically 1,5 V) the oscillator is inhibited and RESET (pin 21) has an undefined level. When V_{DD} rises above the internal reference level, the oscillator is released and RESET is pulled high to V_{DD} by TR1 for a period t_D (typically 50 μs).

N.B. Because of the narrow bandwidth of the crystal, the start-up time of the oscillator is typically 10 ms.

Three modes of power-on reset are possible:

1. If V_{DD} can be switched on with fast rise time i.e. V_{DD} reaches its minimum operating value (corresponding to the selected oscillator frequency) before the RESET signal (t_D) has finished, then no extra components are required (see Fig. 21 and 22). Note that the first instruction is executed after the oscillator start-up time plus 1866 clock periods have elapsed.

2. If V_{DD} has a slow rise time then the RESET signal should be stretched by an external RC circuit (see Fig. 23 and 24). In the event of a short drop in the supply voltage, the diode path rapidly discharges the capacitor to ensure a reliable power-on reset. To ensure a correct reset, the RESET signal should reach at least 70% of the final value of V_{DD}. Given that the RESET voltage and V_{DD} rise exponentially, the above requirement is satisfied when the time constant τ of the RESET pulse is >8 times the time constant of V_{DD}. If V_{DD} rises linearly, then a RESET time constant >2 times the rise time of V_{DD} is required.

   When a reset is completed (RESET goes LOW) before the oscillator has started up, program execution begins after the oscillator start-up time plus 1866 clock periods have elapsed (see Fig. 24). If the oscillator is started-up prior to the completion of RESET, then program execution begins 1866 clock periods after RESET goes LOW.

3. Figure 25 shows an external reset to the PCF84C85 during power-on. The external reset signal must remain HIGH until V_{DD} has reached its minimum operating value corresponding to the selected oscillator frequency. When a reset is completed (RESET goes LOW) before the oscillator has started up, program execution begins after the oscillator start-up time plus 1866 clock periods have elapsed (see Fig. 26). If the oscillator is started-up prior to the completion of RESET, then program execution begins 1866 clock periods after RESET goes LOW.

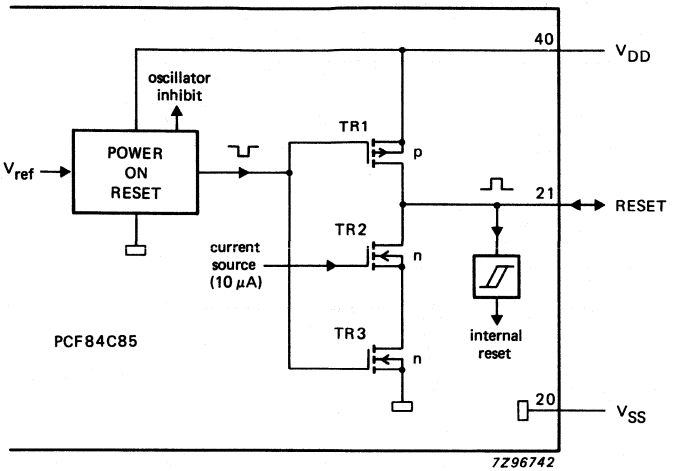


Fig. 21 Power-on-reset configuration.

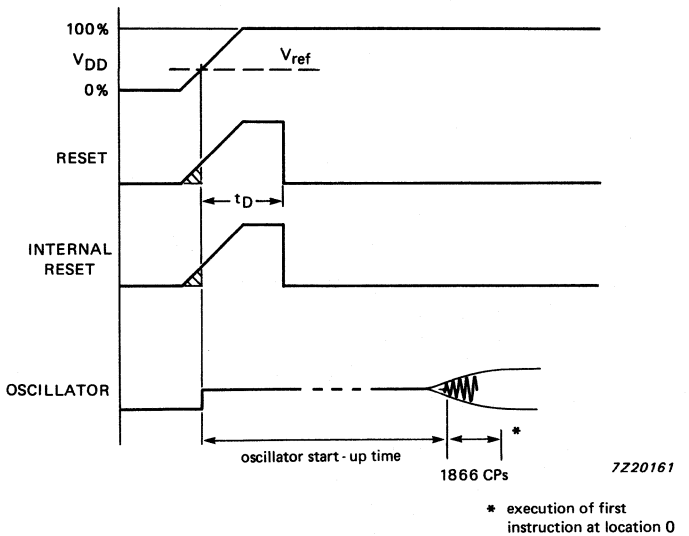


Fig. 22 Timing of power-on-reset with fast rise time.

FUNCTIONAL DESCRIPTION (continued)

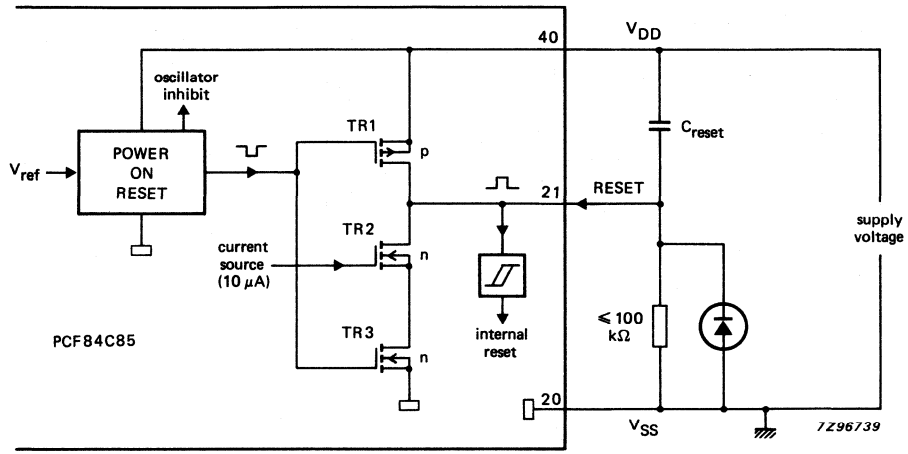


Fig. 23 Stretched power-on-reset with external components.

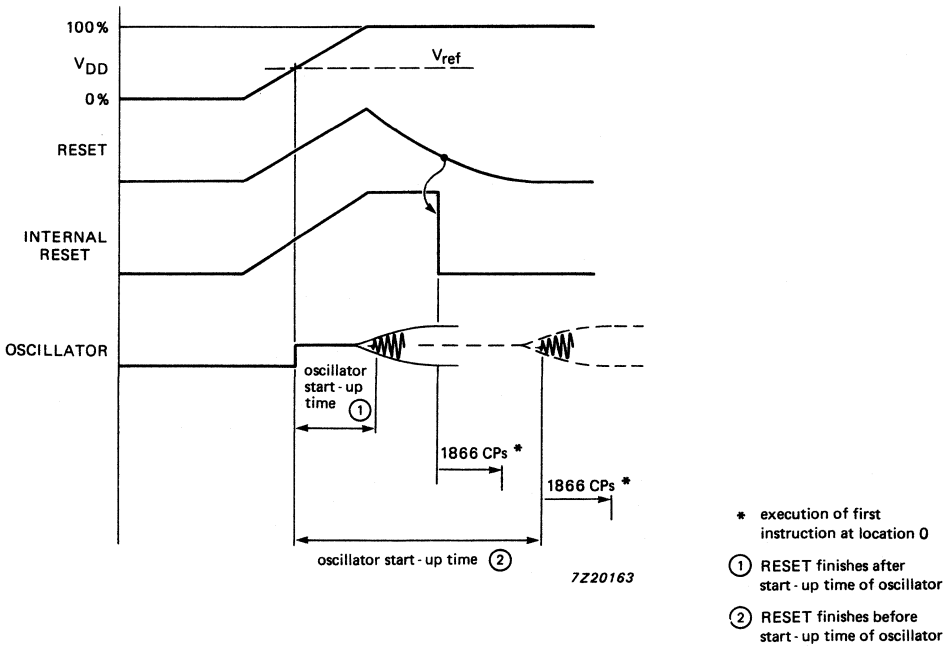


Fig. 24 Timing of power-on-reset with a slowly rising V_{DD} and a stretched RESET pulse.

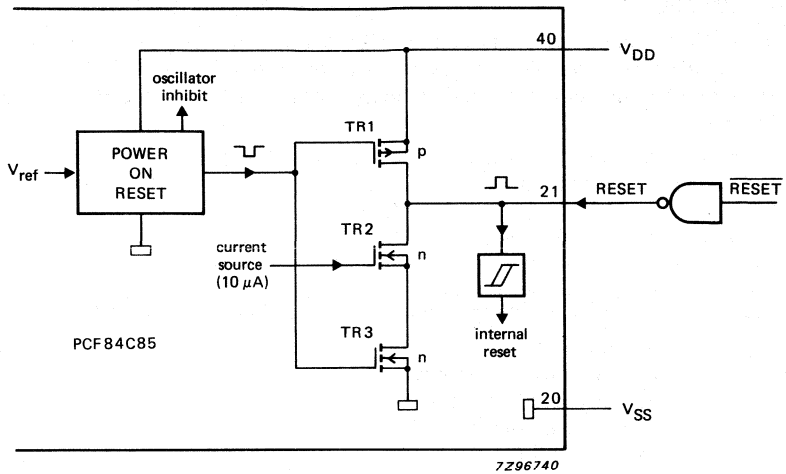


Fig. 25 External power-on-reset configuration.

DEVELOPMENT DATA

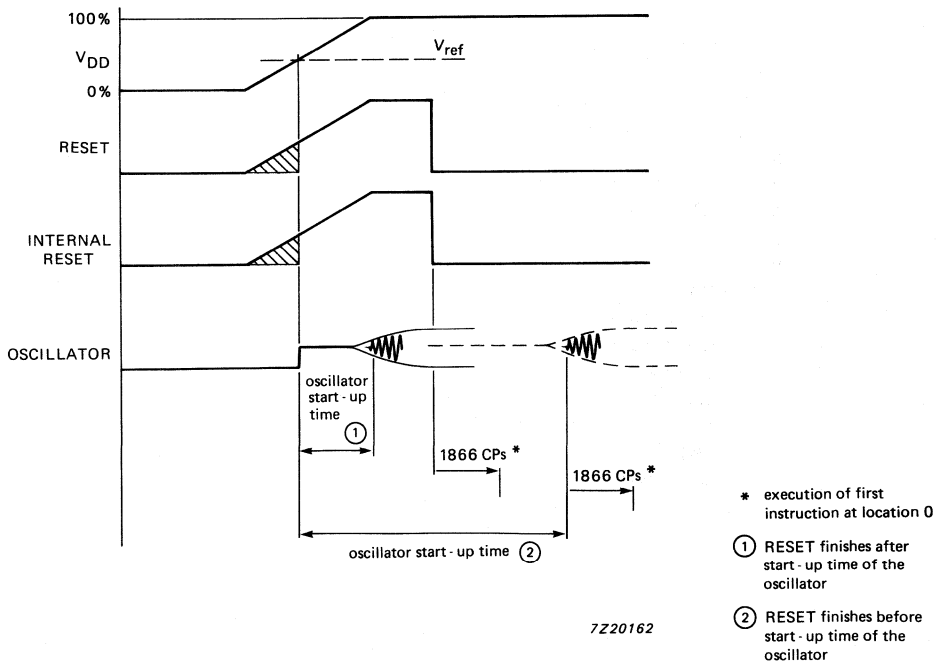


Fig. 26 Timing of external power-on-reset.

INSTRUCTION SET

The PCF84C85 instruction set consists of over 80 one and two byte instructions and is identical to the MAB8400 instruction set. New instructions are added for STOP and IDLE mode. Program code efficiency is high because all RAM locations and all ROM locations on a 256 byte page require only a single byte address.

Table 7 gives the instruction set of the PCF84C85. Table 6 shows the instruction map and Table 5 details the symbols and definition descriptions that are used.

Table 5 Symbols and definitions used in Table 7

symbol	definition description
A	accumulator
addr	program memory address
Bb	bit designation (b = 0-7)
RBS	register bank select
C	carry bit (bit CY)
CNT	event counter
Dx	Derivative register designation (x = 0,1,2 or 3)
data	8-bit number or expression
I	interrupt
MB	memory bank
MBFF	memory bank flip-flop
P	mnemonic for 'in-page' operation
PC	program counter
Pp	port designation (p = 0, 1 or 2)
PSW	program status word
RB	register bank
Rr	register designation (r = 0-7)
Sn	serial I/O register
SP	stack pointer
T	timer
TF	timer flag
T0, T1	test 0 and 1 inputs
#	immediate data prefix
@	indirect address prefix
(X)	contents of X
((X))	contents of location addressed by X
←	is replaced by
↔	is exchanged with

Table 6 PCF84C85 instruction map

first hexadecimal character of opcode ↓	second hexadecimal character of opcode																			
	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	NOP	IDLE		ADD A,#data	JMP page 0	EN I	JNTF addr	DEC A	IN A,Pp 0	1	2		MOV A,Sn 0	1		
1	INC @Rr 0	1	JB0 addr	ADDC A,#data	CALL page 0	DIS I	JTF addr	INC A	INC Rr 0	1	2	3	4	5	6	7
2	XCH A,@Rr 0	1	STOP	MOV A,#data	JMP page 1	EN TCNTI	JNTO addr	CLR A	XCH A,Rr 0	1	2	3	4	5	6	7
3	XCHD A,@Rr 0	1	JB1 addr		CALL page 1	DIS TCNTI	JTO addr	CPL A	OUTL Pp,A 0	1	2		MOV Sn,A 0	1	2	
4	ORL A,@Rr 0	1	MOV A,T	ORL A,#data	JMP page 2	STRT CNT	JNT1 addr	SWAP A	ORL A,Rr 0	1	2	3	4	5	6	7
5	ANL A,@Rr 0	1	JB2 addr	ANL A,#data	CALL page 2	STRT T	JT1 addr	DA A	ANL A,Rr 0	1	2	3	4	5	6	7
6	ADD A,@Rr 0	1	MOV T,A		JMP page 3	STOP TCNT		RRC A	ADD A,Rr 0	1	2	3	4	5	6	7
7	ADDC A,@Rr 0	1	JB3 addr		CALL page 3			RR A	ADDC A,Rr 0	1	2	3	4	5	6	7
8				RET	JMP page 4	EN SI			ORL Pp,#data 0	1	2		MOV A,Dx	MOV Dx,A	ANL Dx,A	ORL Dx,A
9			JB4 addr	RETR	CALL page 4	DIS SI	JNZ addr	CLR C	ANL Pp,#data 0	1	2		MOV Sn,#data 0	1	2	
A	MOV @Rr,A 0	1		MOVP A,@A	JMP page 5	SEL MB2		CPL C	MOV Rr,A 0	1	2	3	4	5	6	7
B	MOV @Rr,#data 0	1	JB5 addr	JMPP @A	CALL page 5	SEL MB3			MOV Rr,#data 0	1	2	3	4	5	6	7
C	DEC @Rr 0	1			JMP page 6	SEL RB0	JZ addr	MOV A,PSW	DEC Rr 0	1	2	3	4	5	6	7
D	XRL A,@Rr 0	1	JB6 addr	XRL A,#data	CALL page 6	SEL RB1		MOV PSW,A	XRL A,Rr 0	1	2	3	4	5	6	7
E	DJNZ @Rr,addr 0	1			JMP page 7	SEL MB0	JNC addr	RL A	DJNZ Rr, addr 0	1	2	3	4	5	6	7
F	MOV A,@Rr 0	1	JB7 addr		CALL page 7	SEL MB1	JC addr	RLC A	MOV A,Rr 0	1	2	3	4	5	6	7

INSTRUCTION SET (continued)

Table 7 Instruction set

	mnemonic	opcode (hex.)	bytes/ cycles	description	function		notes
ACCUMULATOR	ADD A, Rr	6*	1/1	Add register contents to A	(A)←(A) + (Rr)	r = 0–7	1
	ADD A, @Rr	60 61	1/1	Add RAM data, addressed by Rr, to A	(A)←(A) + ((R0)) (A)←(A) + ((R1))		1
	ADD A, #data	03 data	2/2	Add immediate data to A	(A)←(A) + data		1
	ADDC A, Rr	7*	1/1	Add carry and register contents to A	(A)←(A) + (Rr) + (C)	r = 0–7	1
	ADDC A, @Rr	70 71	1/1	Add carry and RAM data, addressed by Rr, to A	(A)←(A) + ((R0)) + (C) (A)←(A) + ((R1)) + (C)		1
	ADDC A, #data	13 data	2/2	Add carry and immediate data to A	(A)←(A) + data + (C)		1
	ANL A, Rr	5*	1/1	'AND' Rr with A	(A)←(A) AND (Rr)	r = 0–7	
	ANL A, @Rr	50 51	1/1	'AND' RAM data, addressed by Rr, with A	(A)←(A) AND ((R0)) (A)←(A) AND ((R1))		
	ANL A, #data	53 data	2/2	'AND' immediate data with A	(A)←(A) AND data		
	ORL A, Rr	4*	1/1	'OR' Rr with A	(A)←(A) OR (Rr)	r = 0–7	
	ORL A, @Rr	40 41	1/1	'OR' RAM data, addressed by Rr, with A	(A)←(A) OR ((R0)) (A)←(A) OR ((R1))		
	ORL A, #data	43 data	2/2	'OR' immediate data with A	(A)←(A) OR data		
	XRL A, Rr	D*	1/1	'XOR' Rr with A	(A)←(A) XOR (Rr)	r = 0–7	
	XRL A, @Rr	D0 D1	1/1	'XOR' RAM, addressed by Rr, with A	(A)←(A) XOR ((R0)) (A)←(A) XOR ((R1))		
	XRL A, #data	D3 data	2/2	'XOR' immediate data with A	(A)←(A) XOR data		
	INC A	17	1/1	increment A by 1	(A)←(A) + 1		
	DEC A	07	1/1	decrement A by 1	(A)←(A) −1		
	CLR A	27	1/1	clear A to zero	(A)←0		
	CPL A	37	1/1	one's complement A	(A)←NOT(A)		
	RL A	E7	1/1	rotate A left	$(A_{n+1})\leftarrow(A_n)$ $(A_0)\leftarrow(A_7)$	n = 0–6	

ACCUMULATOR (cont.)	RLC A	F7	1/1	rotate A left through carry	$(A_{n+1})\leftarrow A_n$ $(A_0)\leftarrow(C)$, $(C)\leftarrow(A_7)$	n = 0–6	2
	RR A	77	1/1	rotate A right	$(A_n)\leftarrow(A_{n+1})$ $(A_7)\leftarrow(A_0)$	n = 0–6	
	RRC A	67	1/1	rotate A right through carry	$(A_n)\leftarrow(A_{n+1})$ $(A_7)\leftarrow(C)$, $(C)\leftarrow(A_0)$	n = 0–6	2
	DA A	57	1/1	decimal adjust A			2
	SWAP A	47	1/1	swap nibbles of A	$(A_{4-7})\leftrightarrow(A_{0-3})$		
DATA MOVES	MOV A, Rr	F*	1/1	move register contents to A	(A)←(Rr)	r = 0–7	
	MOV A, @Rr	F0 F1	1/1	move RAM data, addressed by Rr, to A	(A)←((R0)) (A)←((R1))		
	MOV A, #data	23 data	2/2	move immediate data to A	(A)←data		
	MOV Rr, A	A*	1/1	move accumulator contents to register	(Rr)←(A)	r = 0–7	
	MOV @Rr, A	A0 A1	1/1	move accumulator contents to RAM location addressed by Rr	((R0))←(A) ((R1))←(A)		
	MOV Rr, #data	B* data	2/2	move immediate data to Rr	(Rr)←data		
	MOV @Rr, #data	B0 data B1 data	2/2	move immediate data to RAM location addressed by Rr	((R0))←data ((R1))←data		
	XCH A, Rr	2*	1/1	exchange accumulator contents with Rr	(A)↔(Rr)	r = 0–7	
	XCH A, @Rr	20 21	1/1	exchange accumulator contents with RAM data addressed by Rr	(A)↔((R0)) (A)↔((R1))		
	XCHD A, @Rr	30 31	1/1	exchange lower nibbles of A and RAM data addressed by Rr	$(A_{0-3})\leftrightarrow((R0_{0-3}))$ $(A_{0-3})\leftrightarrow((R1_{0-3}))$		
	MOV A, PSW	C7	1/1	move PSW contents to accumulator	(A)←(PSW)		
	MOV PSW, A	D7	1/1	move accumulator bit 3 to PSW_3	$(PSW_3)\leftarrow(A_3)$		3
	MOVP A, @A	A3	1/2	move indirectly addressed data in current page to A	$(PC_{0-7})\leftarrow(A)$, $(A)\leftarrow((PC))$		
FLAGS	CLR C	97	1/1	clear carry bit	(C)←0		2
	CPL C	A7	1/1	complement carry bit	(C)←NOT(C)		2

INSTRUCTION SET (continued)

	mnemonic	opcode (hex.)	bytes/ cycles	description	function	notes
REGISTER	INC Rr	1*	1/1	increment register by 1	(Rr)←(Rr) + 1 r = 0–7	
	INC @Rr	10 11	1/1	increment RAM data, addressed by Rr, by 1	((R0))←((R0)) + 1 ((R1))←((R1)) + 1	
	DEC Rr	C*	1/1	decrement register by 1	(Rr)←(Rr) – 1 r = 0–7	
	DEC @Rr	C0 C1	1/1	decrement RAM data, addressed by Rr, by 1	((R0))←((R0)) – 1 ((R1))←((R1)) – 1	
BRANCH	JMP addr	● 4 address	2/2	unconditional jump within a 2 K bank	(PC_{8-10})←$addr_{8-10}$ (PC_{0-7})←$addr_{0-7}$ (PC_{11-12})←MBFF 0–1	
	JMPP @A	B3	1/2	indirect jump within a page	(PC_{0-7})←((A))	
	DJNZ Rr, addr	E* address	2/2	decrement Rr by 1 and jump if not zero to addr	(Rr)←(Rr) – 1 r = 0–7 if (Rr) not zero (PC_{0-7})←addr	
	DJNZ @Rr, addr	E0 address	2/2	decrement RAM data, addressed by Rr by 1 and jump if not zero to addr	((R0))←((R0)) – 1 if ((R0)) not zero (PC_{0-7})←addr	
		E1 address			((R1))←((R1)) – 1 if ((R1)) not zero (PC_{0-7})←addr	
	JBb addr	▲ 2 address	2/2	jump to addr if Acc. bit b = 1	if b = 1 : (PC_{0-7})←addr b = 0–7	
	JC addr	F6 address	2/2	jump to addr if C = 1	if C = 1 : (PC_{0-7})←addr	
	JNC addr	E6 address	2/2	jump to addr if C = 0	if C = 0 : (PC_{0-7})←addr	
	JZ addr	C6 address	2/2	jump to addr if A = 0	if A = 0 : (PC_{0-7})←addr	
	JNZ addr	96 address	2/2	jump to addr if A is NOT zero	if A≠0 : (PC_{0-7})←addr	
	JT0 addr	36 address	2/2	jump to addr if T0 = 1	if T0 = 1: (PC_{0-7})←addr	
	JNT0 addr	26 address	2/2	jump to addr if T0 = 0	if T0 = 0: (PC_{0-7})←addr	
	JT1 addr	56 address	2/2	jump to addr if T1 = 1	if T1 = 1: (PC_{0-7})←addr	
	JNT1 addr	46 address	2/2	jump to addr if T1 = 0	if T1 = 0: (PC_{0-7})←addr	
	JTF addr	16 address	2/2	jump to addr if Timer Flag = 1	if TF = 1: (PC_{0-7})←addr	4
	JNTF addr	06 address	2/2	jump to addr if Timer Flag = 0	if TF = 0: (PC_{0-7})←addr	

TIMER/EVENT COUNTER	MOV A, T	42	1/1	move timer/event counter contents to accumulator	(A)←(T)	
	MOV T, A	62	1/1	move accumulator contents to timer/event counter	(T)←(A)	
	STRT CNT	45	1/1	start event counter		
	STRT T	55	1/1	start timer		
	STOP TCNT	65	1/1	stop timer/event counter		
	EN TCNTI	25	1/1	enable timer/event counter interrupt		
	DIS TCNTI	35	1/1	disable timer/event counter interrupt		
CONTROL	EN I	05	1/1	enable external interrupt		
	DIS I	15	1/1	disable external interrupt		
	SEL RB0	C5	1/1	select register bank 0	(RBS)←0	5
	SEL RB1	D5	1/1	select register bank 1	(RBS)←1	5
	SEL MB0	E5	1/1	select program memory bank 0	(MBFF0)←0, (MBFF1)←0	
	SEL MB1	F5	1/1	select program memory bank 1	(MBFF0)←1, (MBFF1)←0	
	SEL MB2	A5	1/1	select program memory bank 2	(MBFF0)←0, (MBFF1)←1	
	SEL MB3	B5	1/1	select program memory bank 3	(MBFF0)←1, (MBFF1)←1	
	STOP	22	1/1	enter STOP mode		
	IDLE	01	1/1	enter IDLE mode		
SUBROUTINE	CALL addr	▲ 4 address	2/2	jump to subroutine	((SP))←(PC), ($PSW_{4,6,7}$) 6 (SP)←(SP) + 1 (PC_{8-10})←$addr_{8-10}$ (PC_{0-7})←$addr_{0-7}$ (PC_{11-12})←MBFF 0–1	6
	RET	83	1/2	return from subroutine	(SP)←(SP) – 1 (PC)←((SP))	6
	RETR	93	1/2	return from interrupt and restore bits 4, 6, 7 of PSW	(SP)←(SP) – 1 ($PSW_{4,6,7}$) + (PC)←((SP))	6

INSTRUCTION SET (continued)

	mnemonic	opcode (hex.)	bytes/ cycles	description	function	notes
PARALLEL INPUT/OUTPUT	IN A, Pp	08 09 0A	1/2	input port p data to accumulator	(A)←(P0) (A)←(P1) (A)←(P2)	7
	OUTL Pp, A	38 39 3A	1/2	output accumulator data to port p	(P0)←(A) (P1)←(A) (P2)←(A)	
	ANL Pp, #data	98 data 99 data 9A data	2/2	AND port p data with immediate data	(P0)←(P0) AND data (P1)←(P1) AND data (P2)←(P2) AND data	
	ORL Pp, #data	88 data 89 data 8A data	2/2	OR port p data with immediate data	(P0)←(P0) OR data (P1)←(P1) OR data (P2)←(P2) OR data	
DERIVATIVE INPUT/OUTPUT	MOV A, Dx	8C 00 8C 01	2/2 2/2	input pin data of port DP0, DP1 to accumulator	(A)←(D0) (A)←(D1)	8
	MOV Dx, A	8D 02 8D 03	2/2 2/2	move contents of accumulator to latch of port DP0, DP1	(D2)←(A) (D3)←(A)	
	ANL Dx, A	8E 02 8E 03	2/2 2/2	AND contents of DP0, DP1 latch with accumulator	(D2)←(D2) AND (A) (D3)←(D3) AND (A)	
	ORL Dx, A	8F 02 8F 03	2/2 2/2	OR contents of DP0, DP1 latch with accumulator	(D2)←(D2) OR (A) (D3)←(D3) OR (A)	
	MOV A, Dx	8C 02 8C 03	2/2 2/2	move contents of DP0, DP1 latch to accumulator	(A)←(D2) (A)←(D3)	8

	mnemonic	opcode (hex.)	bytes/ cycles	description	function	notes
SERIAL INPUT/OUTPUT	MOV A, S_n	0C 0D	1/2	move serial I/O register contents to accumulator	(A)←(S0) (A)←(S1)	9
	MOV S_n, A	3C 3D 3E	1/2	move accumulator contents to serial I/O register	(S0)←(A) (S1)←(A) (S2)←(A)	
	MOV S_n, #data	9C data 9D data 9E data	2/2	move immediate data to serial I/O register	(S0)←data (S1)←data (S2)←data	
	EN SI	85	1/1	enable serial I/O interrupt		
	DIS SI	95	1/1	disable serial I/O interrupt		
	NOP	00	1/1	no operation		

Notes to Table 8

1. PSW CY, AC affected
2. PSW CY affected
3. PSW PS affected
4. Execution of JTF and JNTF instructions resets the Timer Flag (TF).

* : 8,9,A,B,C,D,E,F
● : 0,2,4,6,8,A,C,E
▲ : 0,3,5,7,9,B,D,F

5. PSW RBS affected
6. PSW SP_0, SP_1, SP_2 affected
7. (A) = 0000 (P23) 111
8. The MSB of A becomes a logic 0
9. (S1) has a different function in read and write operations, see serial I/O interface.

RATINGS

Limiting values in accordance with the Absolute Maximum System (IEC 134)

Supply voltage (pin 40)	V_{DD}		−0,8 to +8	V
All input voltages	V_I		−0,8 to V_{DD} +0,8	V
D.C. current into any input or output	$\pm I_I, \pm I_O$	max.	10	mA
Total power dissipation				
Power dissipation per output				
except P23, SCLK	P_O	max.	50	mW
P23, SCLK	P_O	max.	180	mW
Storage temperature range	T_{stg}		−65 to +150	°C
Operating ambient temperature range	T_{amb}		−40 to +85	°C
Operating junction temperature	T_j	max.	125	°C

HANDLING

Inputs and outputs are protected against electrostatic discharge in normal handling. However, to be totally safe, it is desirable to take normal precautions appropriate to handling MOS devices (see 'Handling MOS devices').

DEVELOPMENT DATA

D.C. CHARACTERISTICS

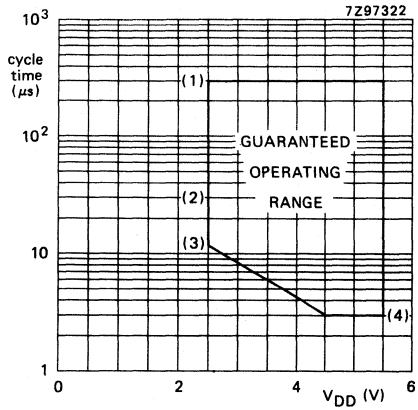
V_{DD} = 2,5 to 5,5 V; V_{SS} = 0 V; T_{amb} = −40 to + 85 °C; all voltages with respect to V_{SS}; unless otherwise specified.

parameter	symbol	min.	typ.	max.	unit
Supply voltage					
operating (see Fig. 27)	V_{DD}	2,5	–	5,5	V
Supply current					
operating (see Fig. 28)					
at V_{DD} = 5 V; f_{XTAL} = 10 MHz	I_{DD}	–	1,6	3,2	mA
at V_{DD} = 5 V; f_{XTAL} = 6 MHz	I_{DD}	–	1	2	mA
at V_{DD} = 3 V; f_{XTAL} = 3,58 MHz	I_{DD}	–	0,5	0,6	mA
IDLE mode (see Fig. 29)					
at V_{DD} = 5 V; f_{XTAL} = 10 MHz	I_{DD}	–	0,8	1,6	mA
at V_{DD} = 5 V; f_{XTAL} = 6 MHz	I_{DD}	–	1	1	mA
at V_{DD} = 3 V; f_{XTAL} = 3,58 MHz	I_{DD}	–	0,25	0,4	mA
STOP mode (see Fig. 35 and note 1)					
at V_{DD} = 2,5 V; T_{amb} = 25 °C	I_{DD}	–	1,2	2,5	μA
at V_{DD} = 2,5 V; T_{amb} = 70 °C	I_{DD}	–	–	10	μA
Inputs					
Input voltage LOW	V_{IL}	0	–	$0{,}3V_{DD}$	V
Input voltage HIGH	V_{IH}	$0{,}7V_{DD}$	–	V_{DD}	V
Input leakage current					
as $V_{SS} < V_I < V_{DD}$	$\pm I_{IL}$	–	–	1	μA
Outputs					
Output voltage LOW					
at $V_I = V_{SS}$ or V_{DD}; $\lvert I_O \rvert < 1$ μA	V_{OL}	–	–	0,05	V
Output sink current LOW					
at V_{DD} = 5 V ± 10%; V_O = 0,4 V					
except P23/SDA, SCLK (see Fig. 31)	I_{OL}	1,6	3	–	mA
P23/SDA, SCLK (see Fig. 32)	I_{OL}	3	–	–	mA
Pull-up output source current HIGH (see Fig. 33)					
at V_{DD} = 5 V ± 10%; $V_O = 0{,}7V_{DD}$	$-I_{OH}$	40	–	–	μA
at V_{DD} = 5 V ± 10%; $V_O = V_{SS}$	$-I_{OH}$	–	–	400	μA
Push-pull output source current HIGH					
at V_{DD} = 5 V ± 10%; $V_O = V_{DD}$−0,4 V	$-I_{OH}$	1,6	3	–	mA

Note 1

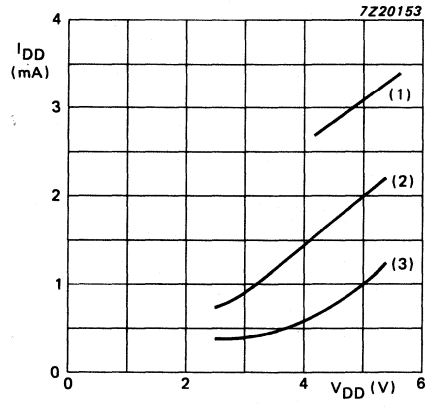
Crystal connected between XTAL1 and XTAL2; SCL and SDA pulled to V_{DD} via 5,6 kΩ resistor; T1 at V_{SS}, $\overline{INT}$ at V_{DD}.

A.C. CHARACTERISTICS (continued)



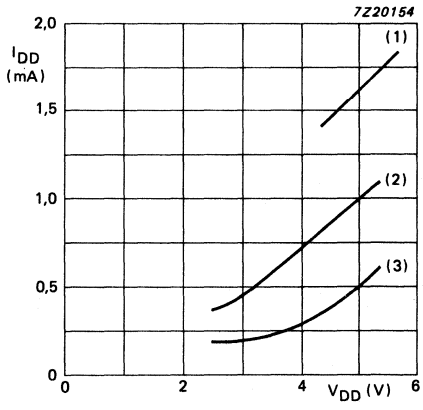
(1) clock frequency = 100 kHz
(2) clock frequency = 1 MHz
(3) clock frequency = 3 MHz
(4) clock frequency = 10 MHz

Fig. 27 Maximum clock frequency (f_{XTAL}) as a function of the supply voltage (V_{DD}).



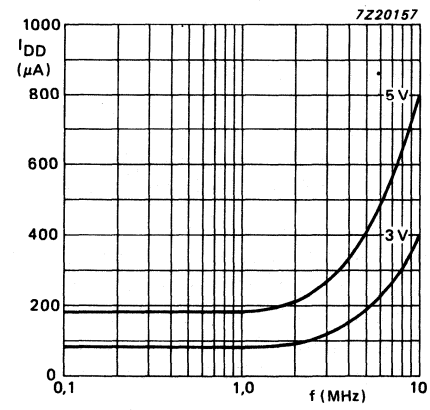
(1) clock frequency = 10 MHz
(2) clock frequency = 6 MHz
(3) clock frequency = 3,58 MHz

Fig. 28 Maximum supply current (I_{DD}) in operating mode as a function of the supply voltage.



(1) clock frequency = 10 MHz
(2) clock frequency = 6 MHz
(3) clock frequency = 3,58 MHz

Fig. 29 Maximum supply current (I_{DD}) in IDLE mode as a function of the supply voltage (V_{DD}).



(1) V_{DD} = 3 V
(2) V_{DD} = 5 V

Fig. 30 Typical supply current during IDLE mode as a function of frequency.

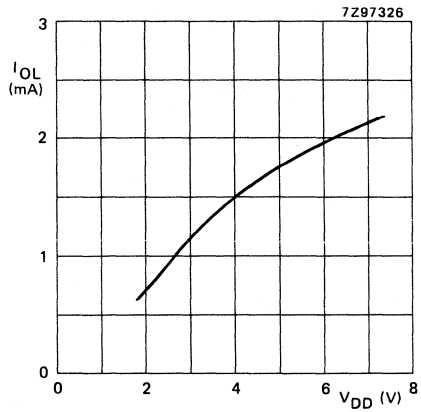


Fig. 31 Output sink current LOW (I_{OL}), except outputs P23/SDA and SCLK, as a function of supply voltage (V_{DD}); V_O = 0,4 V.

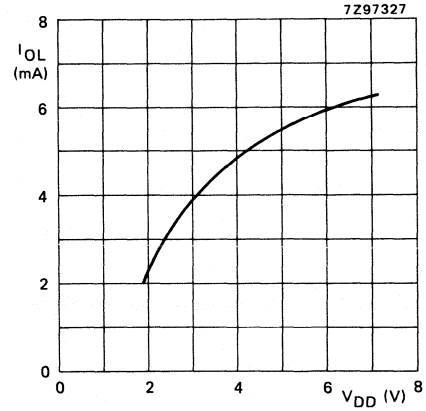
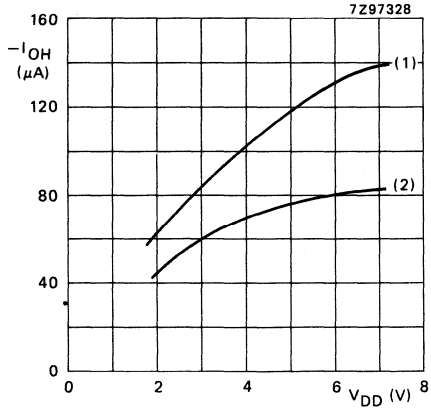


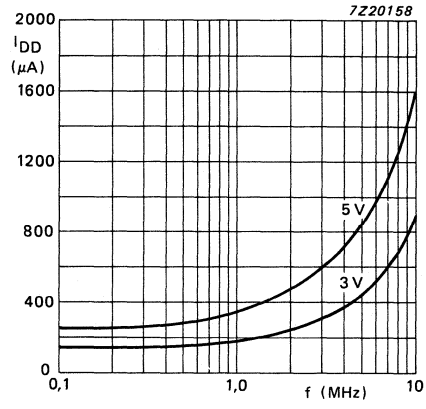
Fig. 32 Output current LOW (I_{OL}), outputs P23/SDA and SCLK, as a function of supply voltage (V_{DD}); V_O = 0,4 V.

DEVELOPMENT DATA



(1) $V_O = V_{SS}$
(2) $V_O = 0{,}7\ V_{DD}$

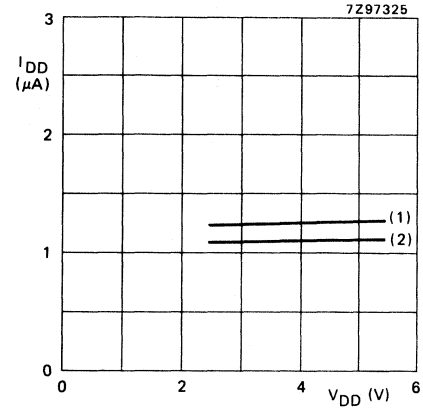
Fig. 33 Output source current HIGH ($-I_{OH}$) as a function of supply voltage (V_{DD}).



(1) V_{DD} = 3 V
(2) V_{DD} = 5 V

Fig. 34 Typical supply current during operating mode as a function of frequency.

A.C. CHARACTERISTICS (continued)



(1) T_{amb} = 85 °C
(2) T_{amb} = 25 °C

Fig. 35 Typical supply current (I_{DD}) in STOP mode as a function of the supply voltage (V_{DD}).

Table 8 Input timing shown in figure 36.

symbol	timing
t_{BUF}	$\geqslant 14t_{XTAL}$
$t_{HD;STA}$	$\geqslant 14t_{XTAL}$
t_{HIGH}	$\geqslant 17t_{XTAL}$
t_{LOW}	$\geqslant 17t_{XTAL}$
$t_{SU;STO}$	$\geqslant 14t_{XTAL}$
$t_{HD;DAT}$	> 0
$t_{SU;DAT}$	$\geqslant 250$ ns
t_{RD}	$\leqslant 1$ μs
t_{RC}	$\leqslant 1$ μs
t_{FD}	$\leqslant 1$ μs
t_{FC}	$\leqslant 0{,}3$ μs

Notes to Table 8

t_{XTAL} = one period of the XTAL input frequency (f_{XTAL})
= 167 ns for f_{XTAL} = 6 MHz.
These figures apply to all modes.

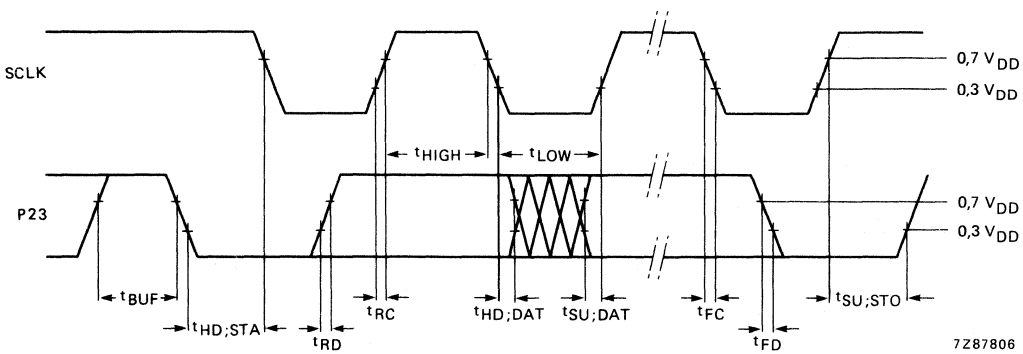


Fig. 36 PCF84C85 timing requirements for the P23 and SCLK *input* signals.

Purchase of Philips' I²C components conveys a license under the Philips' I²C patent to use the components in the I²C-system provided the system conforms to the I²C specifications defined by Philips.

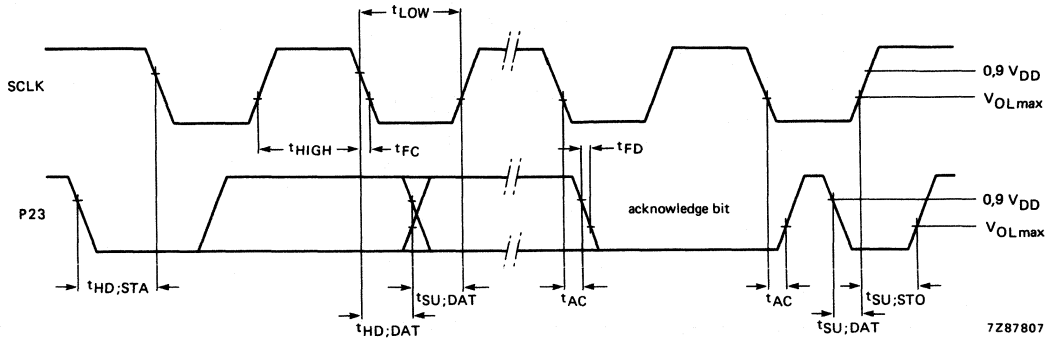


Fig. 37 PCF84C85 timing requirements for the P23 and SCLK *output* signals.

Table 9 Output timing shown in Figure 37

symbol	timing	
	normal mode (ASC in S2 = 0)	low-speed mode (ASC in S2 = 1)
$t_{HD;STA}$	½ (DF + 9) t_{XTAL}	¾ (DF + 9) t_{XTAL}
t_{HIGH}	½ (DF) t_{XTAL}	¾ (DF) t_{XTAL}
t_{LOW}	½ (DF) t_{XTAL}	¼ (DF) t_{XTAL}
$t_{SU;STO}$	½ (DF − 3) t_{XTAL}	¼ (DF − 3) t_{XTAL}
$t_{HD;DAT}$ (slave transmitter)		
any DF	$\geqslant 9t_{XTAL}$ $\leqslant 12t_{XTAL}$	$\geqslant 9t_{XTAL}$ $\leqslant 12t_{XTAL}$
$t_{HD;DAT}$ (master transmitter)		
for DF $\leqslant$ 51	$\geqslant 9t_{XTAL}$ $\leqslant 12t_{XTAL}$	– –
for DF $\leqslant$ 99	– –	$\geqslant 9t_{XTAL}$ $\leqslant 12t_{XTAL}$
$t_{SU;DAT}$ (master transmitter)		
for DF > 51	$\geqslant 15t_{XTAL}$ $\leqslant 24t_{XTAL}$	– –
for DF > 99	– –	$\geqslant 15t_{XTAL}$ $\leqslant 24t_{XTAL}$
t_{AC}	$\geqslant 9t_{XTAL}$ $\leqslant 12t_{XTAL}$	$\geqslant 9t_{XTAL}$ $\leqslant 12t_{XTAL}$
t_{FD},t_{FC}	$\leqslant$ 100 ns at C_b = 400 pF	$\leqslant$ 100 ns at C_b = 400 pF

Notes to Table 9

t_{XTAL} = one period of the XTAL input frequency (f_{XTAL})
= 167 ns for f_{XTAL} = 6 MHz.
DF = divisor (see Table 2 Serial I/O section).
C_b = the maximum bus capacitance for each line.

DEVELOPMENT DATA

SINGLE-CHIP 8-BIT MICROCONTROLLER WITH LCD DRIVER

DESCRIPTION

The PCF84C430 microcontroller is a derivative of the PCF84CXX family of microcontrollers and is manufactured in CMOS technology. For detailed information see the "Single-chip 8-bit Microcontrollers user manual".

The PCF84C430 contains a PCF84CXX core CPU and is completely software compatible. In addition, the PCF84C430 contains an LCD driver supporting four back planes and a maximum driving capacity of up to 96 segments.

The PCF84C430 has 16 quasi-bidirectional I/O port lines, plus a derivative 8-bit port, a serial I/O interface, a single-level vectored interrupt circuit, an 8-bit timer/event counter and on-board clock oscillator and clock circuits.

Features

- 8-bit CPU, ROM, RAM, I/O in a single 64-lead QFP package
- 4 K ROM bytes
- 128 RAM bytes
- On-chip LCD driver with 24 outputs (max. 96 segments)
- LCD multiplexing rates at 1:1 (static), 1:2, 1:3 and 1:4
- Low-power oscillator for LCD driver during STOP mode
- 25 quasi-bidirectional I/O port lines are configured as two 8-bit ports, a 1-bit port (shared with SDA) and an 8-bit derivative port
- Two test inputs: one of which is also the external interrupt input
- Single-level vectored interrupts: external, timer/event counter, serial I/O
- I^2C-bus hardware interface for serial data transfer on two separate lines
- 8-bit programmable timer/event counter
- Clock frequency 100 kHz to 10 MHz
- Over 80 instructions (based on MAB8048) all of 1 or 2 cycles
- Single supply voltage from 2,5 V to 5,5 V ($V_{SS} \leqslant V_{LCD} < V_{DD}$) ←
- STOP and IDLE mode
- Power-on-reset circuit
- Operating temperature range: −40 to + 85 °C

PACKAGE OUTLINE

PCF84C430H: 64-lead quad flat-pack; plastic (SOT208).

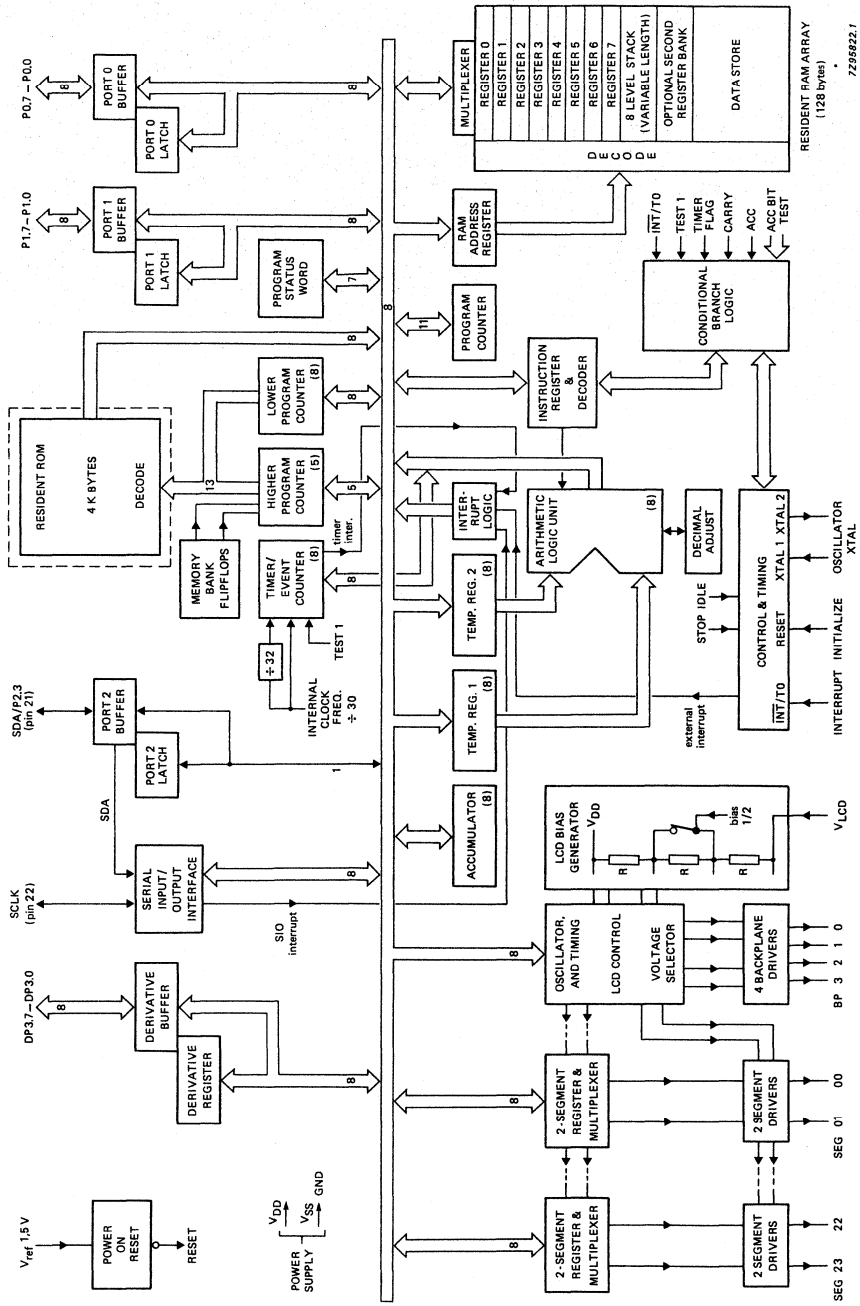


Fig. 1 Block diagram; PCF84C430.

DEVELOPMENT DATA

PINNING

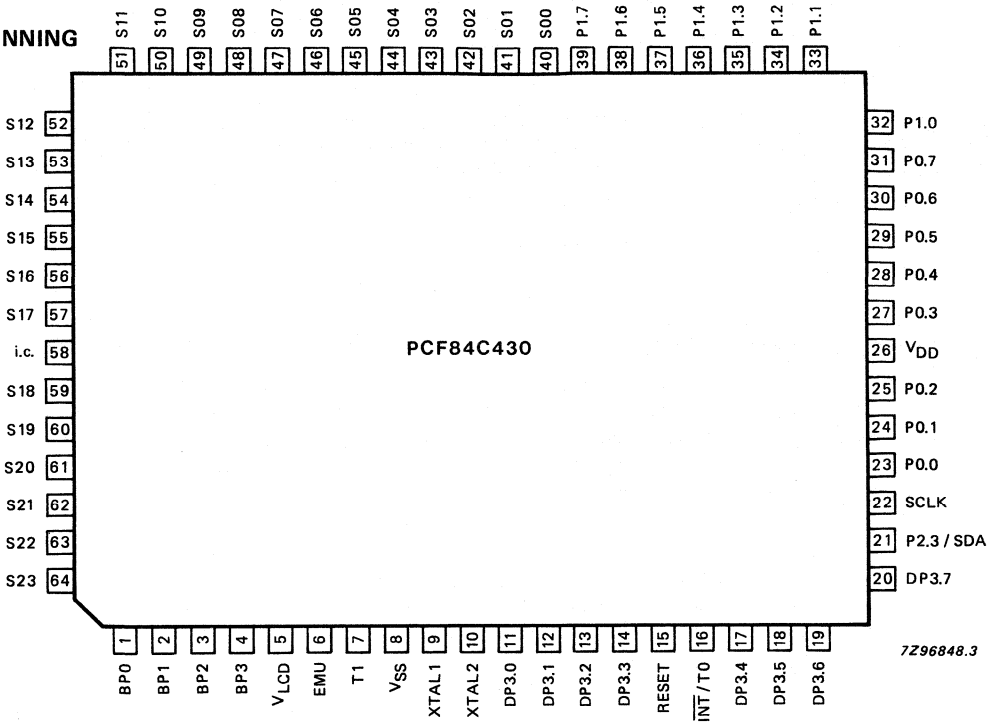


Fig. 2 Pin designation of the PCF84C430.

PIN DESIGNATION

pin	symbol	type	function
1-4	BP0-BP3	O	**LCD:** backplane outputs.
5	V_{LCD}		**LCD:** supply ground level.
6	EMU	I	**EMU pin:** to be grounded for normal operation.
7	T1	I	**Test 1:** test input pin, directly tested by conditional branch instructions JT1 and JNT1. T1 also functions as an input to the 8-bit timer/event counter, using the STRT CNT instruction.
8	V_{SS}		**Ground:** circuit earth potential.
9	XTAL1	I	**Oscillator input:** crystal which determines the internal oscillator frequency or the external clock generator.
10	XTAL2	I/O	Connection to other side of oscillator.
11-14	DP3.0-DP3.3	I/O	**Derivative port 3:** 8-bit parallel port LSBs.
15	RESET	I/O	**Reset input:** used to initialize the processor (active HIGH), or output of power-on-reset circuit.
16	$\overline{INT}$/T0	I	**Interrupt/Test 0:** external interrupt input (sensitive to negative going edge)/test input; when used as a test input directly tested by conditional branch instructions JT0 and JNT0.
17-20	DP3.4-DP3.7	I/O	**Derivative port 3:** 8-bit parallel port MSBs.
21	SDA/P23	I/O	**Serial data:** input/output in serial I/O mode. If not selected as serial data pin, P23 functions as a quasi-bidirectional I/O line.
22	SCLK	I/O	**Serial clock:** bidirectional clock for serial I/O.
23-25 27-31	P0.0-P0.7	I/O	**Port 0:** 8-bit quasi-bidirectional I/O port.
26	V_{DD}		**Power supply:** 2,5 V to 5,5 V.
32-39	P1.0-P1,7	I/O	**Port 1:** 8-bit quasi-bidirectional I/O port.
40-57 59-64	S00-S23	O	**LCD** segment driver outputs.

FUNCTIONAL DESCRIPTION

Program memory

The program memory consists of 4 K bytes, in a read-only memory (ROM).
Each location is directly addressable by the program counter. The memory is mask-programmed at the factory. Figure 3 shows the program memory map.

Four program memory locations are of special importance:

- Location 0; contains the first instruction to be executed after the processor is initialized (RESET)
- Location 3; contains the first byte of an external interrupt service subroutine
- Location 5; contains the first byte of a serial I/O and interrupt service subroutine
- Location 7; contains the first byte of a timer/event counter interrupt service subroutine

Program memory is arranged in banks of 2 K bytes, which are selected by SEL MB instructions. The program memory is further divided into location 'pages', each of 256 bytes. This latter division applies only for conditional branches. Memory bank boundaries can be crossed only by using the unconditional branch instructions after the appropriate memory bank has been selected. A CALL instruction can transfer control to a subroutine on any 'page'; RET and RETR instructions can transfer control from a subroutine back to the main program.

Data memory

Data memory consists of 128 bytes, random-access data memory (RAM).
All locations are indirectly addressable using RAM pointer registers; up to 16 designated locations are directly addressable. Memory also includes an 8-level program counter stack addressed by a 3-bit stack pointer. Figure 4 shows the data memory map.

Working registers

Locations 0 to 7 are designated as working registers, directly addressable by the direct register instructions. Ease of addressing, and a minimum requirement of instruction bytes to manipulate their contents, makes these locations suitable for storing frequently addressed intermediate results. This bank of registers can be selected by the SEL RB0 instruction.

Executing the select register bank instruction SEL RB1, designates locations 24 to 31 as working registers, instead of locations 0 to 7, and these are then directly addressable. This second bank of working registers may be used as an extension of the first or reserved for use during interrupt service subroutines saving the first bank for use in the main program. If the second bank is not used, locations 24 to 31 may serve as general purpose RAM.

The first locations of each bank contain the RAM pointer registers R0, R1, R0' and R1', which indirectly address all RAM locations.

All RAM locations make efficient program loop counters when used with the decrement register and test instruction DJNZ.

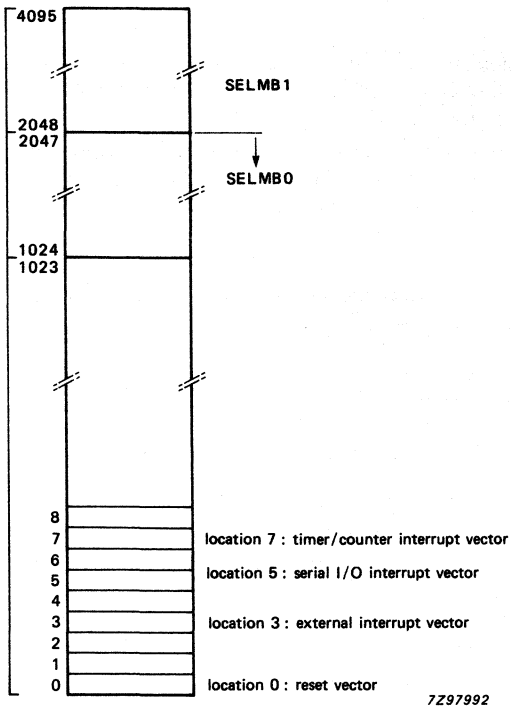


Fig. 3 Program memory map.

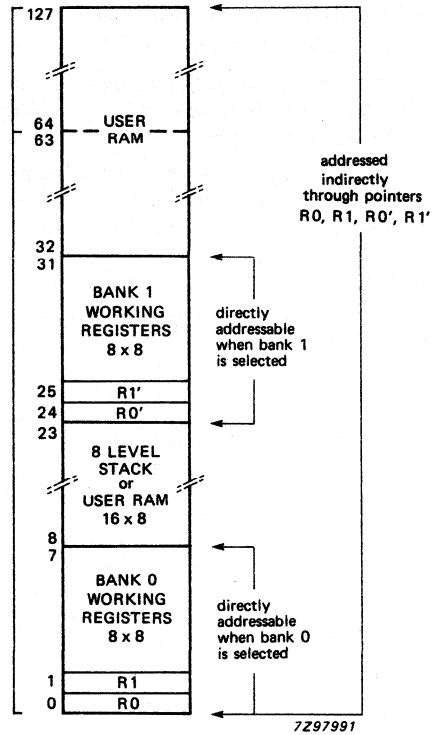


Fig. 4 Data memory map.

DEVELOPMENT DATA

Program counter stack

Locations 8 to 23 may be designated as an 8-level program counter stack (2 locations per level), or as general purpose RAM. The program counter stack (Fig. 5) enables the processor to keep track of the return addresses and status generated by interrupts or CALL instructions by storing the contents of the program counter prior to servicing the subroutine. A 3-bit stack pointer determines which of the eight register pairs of the program counter stack will be loaded with the next generated return address.

The stack pointer, when initialized to 000 by RESET, points to RAM locations 8 and 9. On the first subroutine CALL or interrupt, the contents of the program counter and bits 4, 6 and 7 of the program status word (PSW) are transferred to locations 8 and 9. The stack pointer increments by one and points to locations 10 and 11 ready for another CALL. Because an address may be up to 13 bits long, two bytes must be used to store each address.

At the end of a subroutine, which is signalled by a return instruction (RET or RETR), the stack pointer decrements by one and the contents of the register pair on top of the stack are transferred to the program counter. The saved PSW bits are transferred to the PSW only by the RETR instruction.

If not all 8 levels of subroutine and interrupt nesting are used, the unused portion of the stack may be used as any other indirectly addressable RAM locations.

FUNCTIONAL DESCRIPTION (continued)

Program counter stack (continued)

Nesting of subroutines within subroutines can continue up to 8 times without overflowing the stack. If overflow does occur the deepest address stored (locations 8 and 9) will be overwritten and lost since the stack pointer overflows from 111 to 000. It also underflows from 000 to 111.

The value of the saved contents of the program counter is different for an interrupt CALL compared to a normal CALL to subroutine. With an interrupt CALL, the program counter return address is saved; with a subroutine CALL, the saved program counter value is one less than the program counter return address.

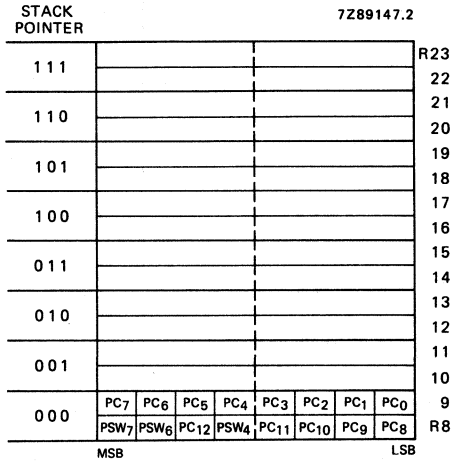


Fig. 5 Program counter stack.

IDLE and STOP modes

Idle mode

When the microcontroller enters the IDLE mode via the IDLE instruction (01 H) the oscillator, timer/counter and serial I/O are kept running. The microcontroller exits from the IDLE mode by one of three interrupts if they are enabled,or by activating a RESET. If the interrupt is not enabled the processor will remain in the IDLE mode. An active signal on the RESET pin restarts the microcontroller and a normal RESET sequence is executed (see Fig. 6).

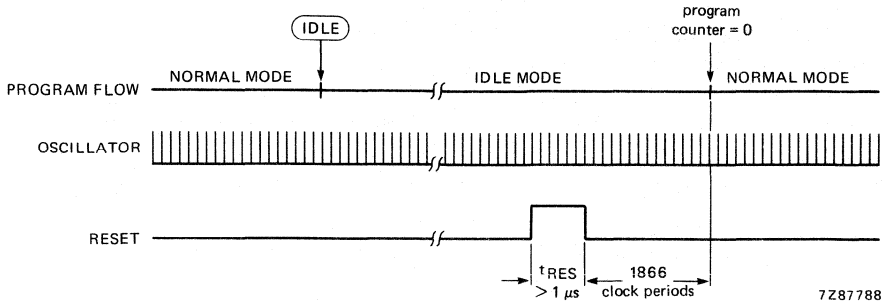


Fig. 6 Exit from IDLE mode via a RESET.

An active signal coming from an enabled interrupt causes the execution of the normal interrupt routine since normal interrupt scanning is still being carried out. A HIGH-to-LOW transition on the external interrupt pin ($\overline{INT}$/T0) reactivates the microcontroller. A LOW level applied to $\overline{INT}$/T0 will reactivate the microcontroller only in the STOP mode. Thus, if $\overline{INT}$/T0 was LOW before the microcontroller entered the IDLE mode, it must go HIGH before the microcontroller can be reactivated (see Fig. 7.).

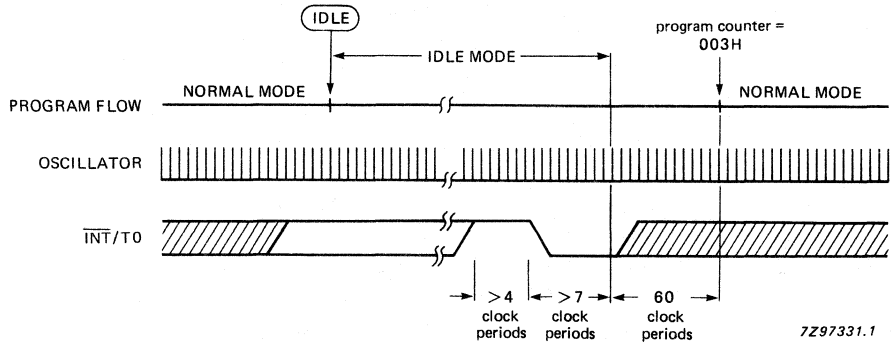


Fig. 7 Exit from IDLE mode via an interrupt.

Wake-up from the IDLE mode is ensured when $\overline{INT}$/T0 is HIGH for at least 4 CP (clock periods) followed by a LOW for 7 CP. After the initial forced CALL 003 H operation (60 CP) the program continues with the external interrupt service routine. During IDLE mode operation, the address of the instruction immediately following that which caused the processor to enter the IDLE mode is present on the address bus.

STOP mode

The microcontroller enters the STOP mode via the STOP instruction (22H). The oscillator is switched off but internal status of the CPU, RAM contents and the state of I/O ports are unaffected. The microcontroller can be brought-out of the STOP mode by an active signal at the external interrupt input or by an external RESET signal. When one of these two signals is applied an internal delay of 1866 CP is provided to ensure that all internal clocks are operating correctly before restart (see Fig. 8). Note: the start-up time of a crystal oscillator is measured in milliseconds, and the 1866 CP count begins after this start-up time.

If the microcontroller exits from the STOP mode by activating RESET, a normal RESET sequence is executed.

FUNCTIONAL DESCRIPTION (continued)

STOP mode (continued)

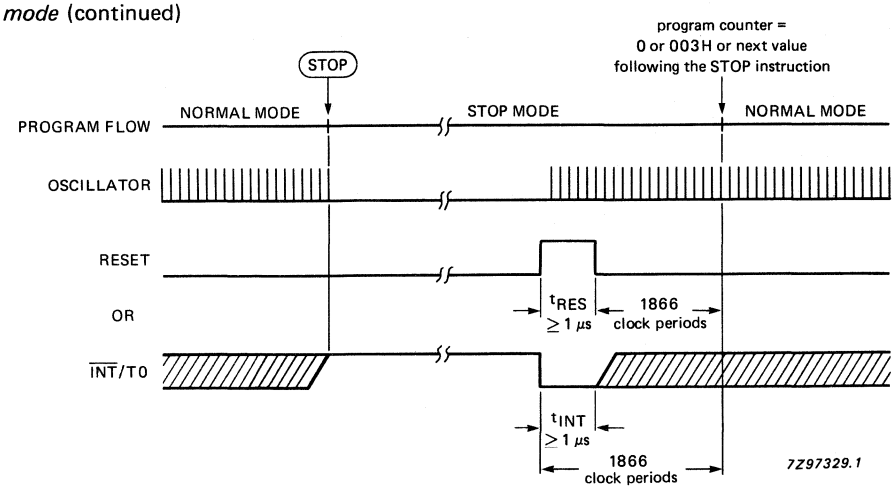


Fig. 8 Entering and exiting the STOP mode.

If the microcontroller exits the STOP mode by pulling the external interrupt input pin LOW, an interrupt sequence is executed only if the external interrupt is enabled. In this event the microcontroller resumes the normal program sequence after returning from the interrupt routine, as in the normal mode. If the interrupt is not enabled, it continues the normal program sequence, executing the instruction following the STOP instruction.

The microcontroller is restarted by a LOW level applied at the $\overline{INT}$/T0 pin.
Note: when leaving the STOP mode via an interrupt, a further instruction in the main program series is executed prior to entering the interrupt routine.

When the $\overline{INT}$/T0 level is active during the STOP instruction then no STOP is executed.

A LOW level on the external interrupt input of at least 1 μs will cause the microcontroller to exit the STOP mode. During the STOP mode, the address of the instruction immediately following the last STOP instruction is present on the address bus.

I/O facilities

The PCF84C430 has 25 quasi-bidirectional I/O lines arranged as:

- Port 0 parallel port of 8 lines (P0.0 to P0.7)
- Port 1 parallel port of 8 lines (P1.0 to P1.7)
- Port 2 I/O port of 1 line (SDA/P2.3) active when not selected as serial I/O
- Port 3 a derivative 8-bit parallel port (DP3.0 to DP3.7)

In addition 4 specialized I/O lines are comprised:

- SCLK serial I/O clock line
- SDA/P2.3 serial I/O data line
- $\overline{INT}$/T0 external interrupt and test input. When used as a test input it can be directly tested by conditional branch instructions JT0 and JNT0
- T1 test input which can alter program sequences when tested by conditional jump instructions JT1 and JNT1. T1 also functions as an input to the 8-bit timer/event counter.

Parallel ports

All parallel ports can be used as outputs or inputs, their structure is quasi-bidirectional.
Output data written to a port is latched and remains unchanged until rewritten.
Input data is not latched and so must be present until read by an input instruction.

Input lines are fully CMOS compatible, output lines can drive one TTL or CMOS load.

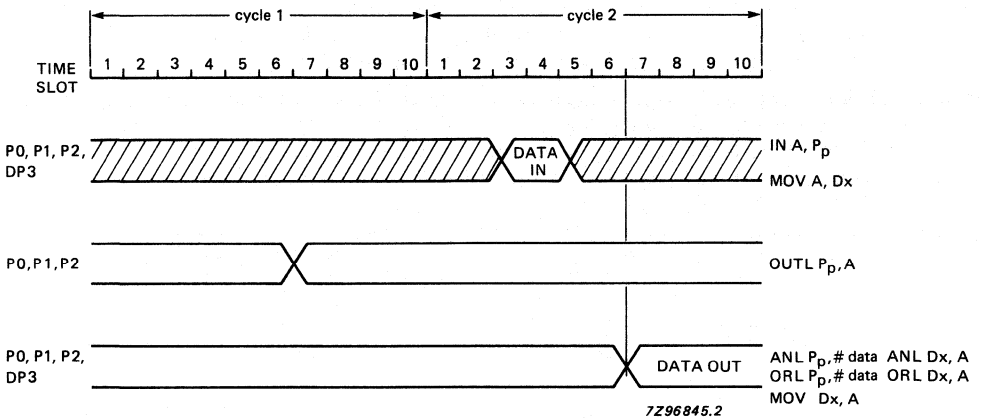


Fig. 9 Shows the timing diagram for all ports using IN, OUTL, ANL and ORL instructions. For the OUTL instruction data changes on time slot 7 of cycle 1. For the MOV, ANL and ORL instructions, the ports change on time slot 7 of cycle 2.

Fig. 10 shows the quasi-bidirectional I/O interface with push-pull output and switched pull-up current source.
Each line is pulled up to V_{DD} via a constant current source (TR4), which is enabled via TR3 whenever one of the two output latches contains a logic 1. This current source is sufficient for a TTL HIGH level, yet can be pulled LOW by an external CMOS device, thus allowing the same pin to be used for both input and output.

When a logic 1 is written to the line for the first time (MQ = 1, SQ = 0), TR2 is switched on for the duration of the internal write pulse (one oscillator period), to provide a fast transition from logic 0 to logic 1. Subsequent writing of a logic 1 to the port lines will not switch TR2 on. This prevents unnecessary current through external components connected to the port lines of the same port which might be in the input mode and also connected to ground.

When a logic 0 is written to the line, TR3 switches off the current source. Current sinking capability is provided by TR1, which is now switched on. When used as an input, a logic 1 must first be written to the line, otherwise TR1 will remain low impedance.

FUNCTIONAL DESCRIPTION (continued)

Parallel ports (continued)

The PCF84C430 family offers the possibility to select individually 24 of the 25 parallel port pins (not P2.3)* from the following mask options:

Option 1 – STANDARD PORT; quasi-bidirectional I/O with switched pull-up current source of 100 μA (typ.) and P-channel booster transistor TR2.
TR2 is only active during 1 clock cycle (Fig. 10).

Option 2 – OPEN DRAIN; quasi-bidirectional I/O with only an N-channel open drain output. Application as an output requires connection of an external pull-up resistor (Fig. 11). When an open drain port is unused it must be connected to V_{SS}.

Option 3 – PUSH-PULL OUTPUT; drive capability of the output is 1,6 mA (min.) at V_{DD} = 5 V in both polarities. To avoid a large current flowing through the output transistors during the input mode, these push-pull pins must only be used as outputs (Fig. 12).
Note: the port latch may be read back using the IN A,pp instruction.

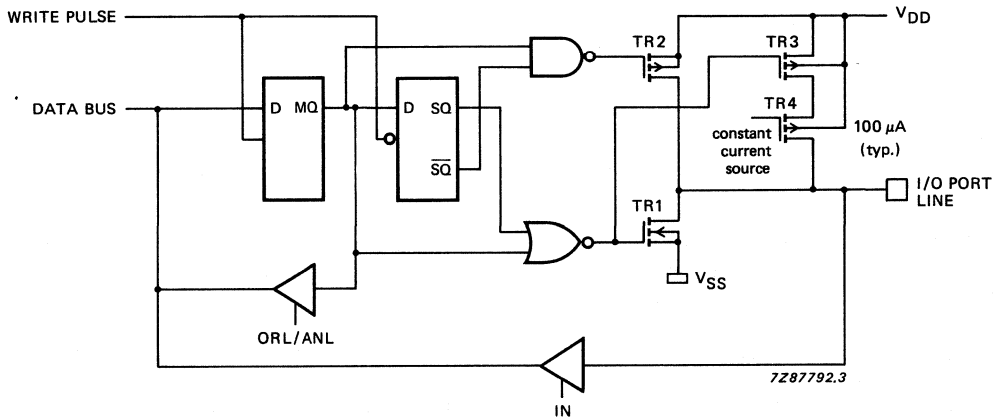


Fig. 10 Standard output with switch pull-up current source.

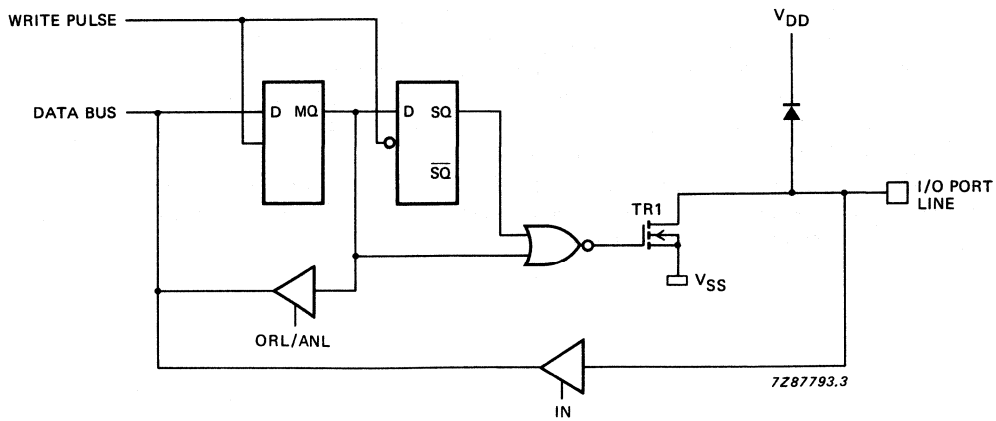


Fig. 11 Open drain output.

* P2.3 is always open drain.

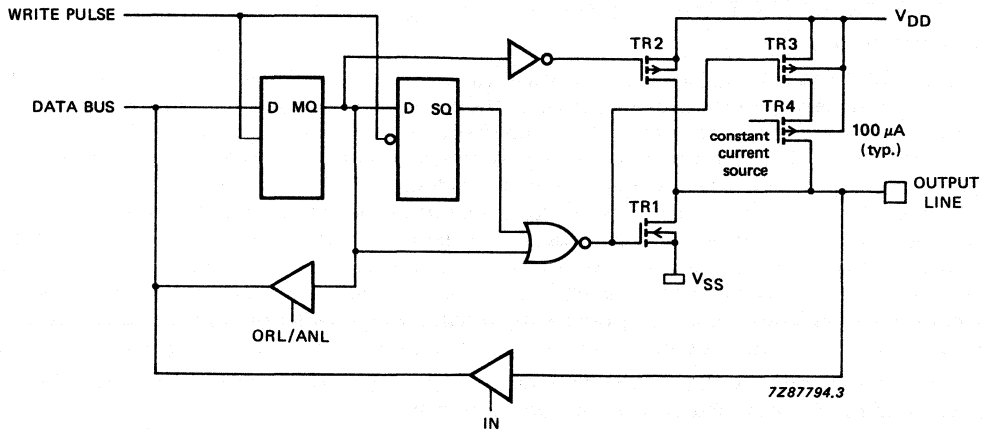


Fig. 12 Push-pull output.

Derivative I/O addresses

For extension of the I/O facilities, instructions MOV A,Dx, MOV Dx,A, ANL Dx,A and ORL Dx,A have been added to the instruction set. The derivative register address is the second byte of the instruction. The address map for the derivative registers is given in Table 1.

Table 1 address map of PCF84C430 derivative registers

Dx	R/W	bits	description
4	R/W	8	PR3; derivative I/O port 3 latch
5	R	8	PR3; derivative I/O port 3 input pins
16	R/W	8	LCDC; LCD control register
17-28	R/W	8	LCDB1-LCDB12; 12 consecutive segment data bytes

Serial I/O (SIO)

The PCF84C430 has an on-chip serial I/O interface that supports I^2C-bus communication. Whereas a normal microcontroller must regularly monitor the serial data bus for the presence of data, the serial I/O interface detects, receives and converts the serial data stream into parallel format without interrupting the execution of the current program. An interrupt is sent to the PCF84C430 only when a complete byte is received. It then reads the data byte in one instruction.

During transmission, the serial I/O interface performs parallel to serial conversion and subsequent serial output of the data. The microcontroller is only interrupted in the execution of its programmed tasks when a complete byte has been transmitted.

FUNCTIONAL DESCRIPTION (continued)

Serial I/O (SIO) (continued)

The design of the PCF84C430 serial I/O system allows any number of devices from PCF85XX family (clips) to be connected via the two-line serial bus. The ability of any devices to communicate, without interrupting the operation of any other devices on the bus, is an outstanding attribute of the system. This is achieved by allocating a specific 7-bit address to each device and providing a system whereby a device reacts only to a message prefixed with its own address or the 'general CALL' address. Address recognition is performed by the interface hardware so that operation of the microcontroller need only be interrupted when a valid address has been received. This saves significant processing time and memory space compared with a conventional microcontroller employing a software serial interface. When the addressing facility is not required, for instance in a system with only two microcontrollers, direct data transfer without addressing can be performed. In multi-master systems, an automatically, invoked arbitration procedure prevents two or more devices from continuing simultaneous transmission.

In NORMAL (running) and IDLE mode, the serial I/O logic remains active; its internal system clock will be switched off when there is no activity on the serial bus.

After execution of the STOP instruction, the oscillator of the PCF84C430 is switched off. This means that the serial I/O logic will remain in the state it was at the occurance of the STOP instruction. To avoid "bus block" problems and to assure correct start-up of the bus after exit from the STOP mode, the user should disable the serial logic (ESO = 0) prior to the execution of the STOP instruction. This must be carried out only when the PCF84C430 has finished a serial data transfer.

After a negative-going RESET signal, the first 30 clock pulses of the 1866-pulse initialization phase set P2.3/SDA and SCLK to HIGH. When P2.3/SDA or SCLK are not used they must be connected to V_{SS}.

Serial I/O interface

Figure 13 shows the serial I/O interface. The clock line of the serial bus has exclusive use of pin 22 (SCLK) while the data line shares pin 21 (SERIAL DATA) with the I/O line P2.3 of port 2. When the serial I/O is enabled, P2.3 is disabled as a parallel port line; (P2.3 and SCLK only open drain).

The microcontroller and interface communicate via the internal microcontroller bus and the Serial Interrupt Request line. Data and information controlling the operation of the interface are stored in four registers:

- Data shift register (S0)
- Serial I/O interface status word (S1)
- Serial clock control word (S2)
- Address register (SO')

Data shift register (S0)

Register S0 converts serial data to parallel format and vice versa. A pending interrupt is generated only after a complete byte has been transmitted, or after a complete data byte, specific address or 'general CALL' address has been received. The most significant bit is transmitted first.

Serial I/O interface status word (S1)

Register S1 provides information concerning the state of the interface and stores information from the microcontroller. Bits 0 to 3 are duplicated: control bits in these positions can only be written by the microcontroller, while status bits can only be read.

MST and TRX (see Table 2)

These bits determine the operating mode of the serial I/O interface.

Table 2 Operating modes of the serial I/O interface

MST	TRX	operating mode
0	0	slave receiver
1	0	master receiver
0	1	slave transmitter
1	1	master transmitter

BB: Bus Busy.

This is the flag which indicates the status of the bus.

PIN: Pending Interrupt Not

PIN = '0' indicates the presence of a pending interrupt, which will cause a Serial Interrupt Request when the serial interrupt mechanism is enabled.

ESO: Enable Serial Output

The ESO flag enables/disables the serial I/O interface: ESO = 1 enables, ESO = 0 disables. ESO can only be written by software.

BC0, BC1 and BC2

Bits BC0, BC1 and BC2 indicate the number of bits received or transmitted in a data stream. These bits can only be written by software.

AL: Arbitration Lost

The arbitration lost flag is set by hardware when the serial I/O interface, as master transmitter, loses a bus arbitration procedure.

AAS: Addressed As Slave

This flag is set by hardware when the interface detects either its own specific address or the 'general CALL' address as the first byte of a transfer and the interface has been programmed to operate in the address recognition mode.

AD0: Address Zero

This flag is set by hardware after detection of the 'general CALL' address when the interface is operating in the address recognition mode.

LRB: Last Received Bit

This contains either the last data bit received or, for a transmitting device in the acknowledgement mode, the acknowledgement signal from the receiving device.

Bits AL, AAS, AD0 and LRB can only be read by software.

FUNCTIONAL DESCRIPTION (continued)

Serial I/O interface (continued)

Serial clock control word (S2)

Bits 0 to 4 of the clock control register S2 are used to set the frequency of the serial clock signal. When a 6 MHz crystal is used, the frequency of the serial clock can be varied between 154 kHz and 1 kHz (see Table 3).
An asymmetrical clock with a HIGH-to-LOW ratio of 3:1 can be generated using bit 5. The asymmetrical clock allows a microcontroller more time per clock period for sampling the data line, making the timing of this action less critical. Bit 6 can be used to activate the acknowledge mode of the serial I/O.
S2 is a write only register.

Address register

The address register contains the 7-bit address back-up latches and the bit (ALS) used to enable/disable the address recognition mode. The address register can be written using the MOV S0, A and MOV S0, # data instructions, but only when ESO = 0 .

Serial I/O interrupt logic

An EN SI instruction enables and a DIS SI instruction disables the interrupt logic. When the logic is enabled, a pending interrupt results in a serial I/O interrupt to the processor, causing a CALL to location 5 in the ROM. When disabled, the presence of an interrupt is still indicated by PIN in S1, allowing the interrupt to be serviced. However, vectored interrupt will not occur.

Table 3 SIO clock pulse frequency control when using a 6 MHz and a 10 MHz crystal

hexadecimal S20-S24 code	divisor	f_{XTAL} (6 MHz) f_{SCLK} (kHz)▲	f_{XTAL} (10 MHz) f_{SCLK} (kHz)▲
0	not allowed		
1	39	*154	*256
2	45	*133	*222
3	51	*118	*196
4	63	95	*159
5	75	80	*133
6	87	69	*115
7	99	61	*101
8	123	49	81
9	147	41	68
A	171	35	58
B	195	31	51
C	243	25	41
D	291	21	34
E	339	18	29
F	387	16	26
10	483	12	21
11	579	10	17
12	675	8,9	15
13	771	7,8	13,4
14	963	6,2	10,4
15	1155	5,2	8,7
16	1347	4,5	7,4
17	1539	3,9	6,5
18	1923	3,1	5,2
19	2307	2,6	4,3
1A	2691	2,2	3,7
1B	3075	2,0	3,3
1C	3843	1,6	2,6
1D	4611	1,3	2,2
1E	5379	1,1	1,9
1F	6147	1,0	1,6

DEVELOPMENT DATA

* Not permitted for I^2C operation.

▲ The maximum clock frequency in the I^2C systems is 100 kHz.

FUNCTIONAL DESCRIPTION (continued)

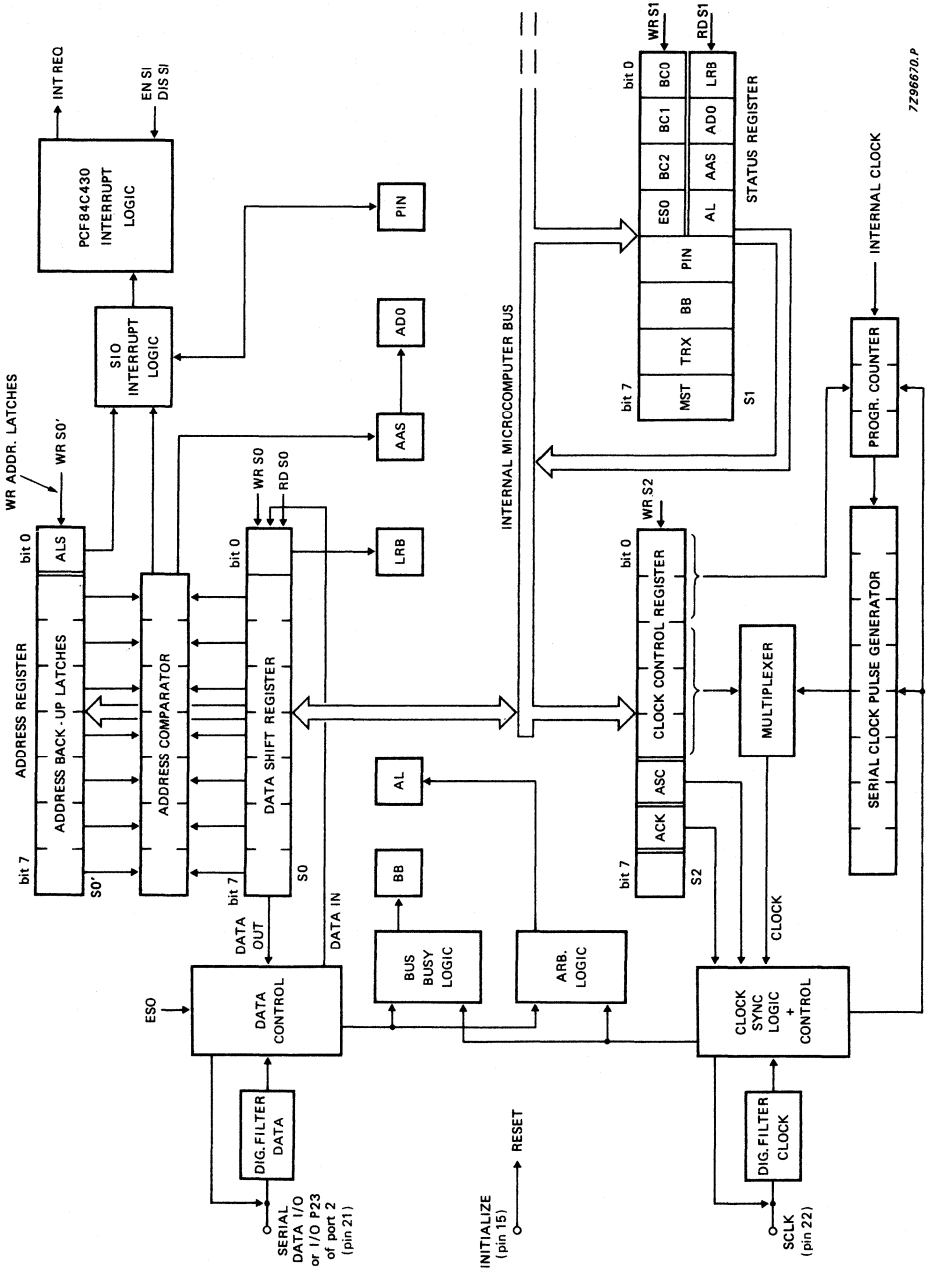


Fig. 13 Serial I/O interface.

Interrupts

Upon entering an interrupt routine, the contents of the program counter and bits 4, 6 and 7 of the PSW are saved in the program counter stack. The contents of the accumulator must be saved by user software. Interrupt acknowledgement may be carried out by software via I/O ports. All interrupt routines must reside in memory bank 0; the SEL MB instructions may not be used within an interrupt routine. An interrupt routine can only be terminated by the RETR (return and restore) instruction. During an interrupt routine, further subroutine calls must be terminated using the RET instruction. Using the RETR instruction to terminate such a nested subroutine would terminate the interrupt routine prematurely.

External interrupt

When the external interrupt is enabled and no interrupt routine is in progress, a HIGH-to-LOW transition on the $\overline{\text{INT}}$/T0 pin sets the External Interrupt Flag (EIF) and invokes the external interrupt routine by forcing a CALL to location 3*. The program counter points to the external interrupt vector address (003 H) between 2,6 and 3,6 machine cycles after the transition occurs. Interrupt latency will depend upon the instruction that is being executed when the transition occurs. If an interrupt routine is already in progress, an external interrupt request is stored in the External Interrupt flag (EIF). When the external interrupt is disabled the request is still latched into the digital filter. Execution of a DIS I instruction cancels a stored interrupt request by clearing both the digital filter and the (EIF).

Another external interrupt can be created by enabling the timer/event counter interrupt and loading FFH into the counter (one less than overflow). The STRT CNT instruction is then executed in user software, this enables the event counter mode and a LOW-to-HIGH transition on the T1 input will then initiate an interrupt subroutine and invoke a call to the timer/counter interrupt vector location 7.

SIO Interrupt

An interrupt request from the SIO hardware will set the PIN flag to its active LOW state. This action is fully independant of the Enable SIO interrupt flag. When the SIO interrupt is enabled and no interrupt routine is in progress, the PIN flag at active LOW will invoke the SIO interrupt routine by forcing a CALL to program memory location 5. After the SIO interrupt is initiated, the PIN flag is **not** automatically set back to a HIGH state, this must be done as part of the interrupt subroutine.

Timer/counter Interrupt

When no interrupt routine is in progress and the timer/counter is enabled, a timer/counter overflow sets the Timer interrupt flag (TIF). This initiates the timer interrupt routine by forcing a CALL to program memory location 7**. If an interrupt routine is in progress, the interrupt request is stored in the (TIF) only if the timer interrupt has been enabled. Execution of a DIS TCNTI instruction deletes a previously stored interrupt request. The timer flag (TF) is set every time the timer/counter overflows and is not automatically reset after the timer counter routine is called. It can only be cleared by either the JTF or JNTF or by a hardware RESET.

Interrupt Priority

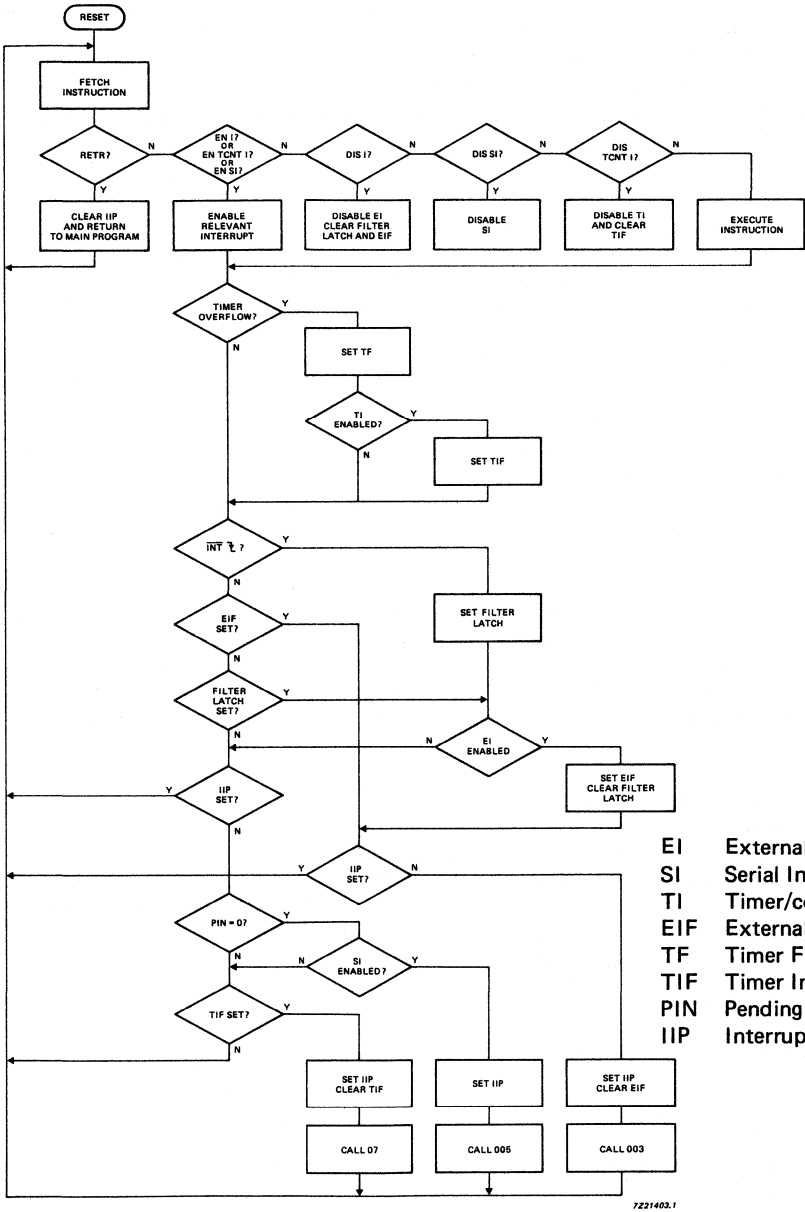
If simultaneous interrupts occur, their priority is as follows:

- External (highest)
- SIO
- Timer/Counter (lowest)

An interrupt routine can only be interrupted by a hardware RESET and cannot be interrupted by other interrupts (which will be latched). When the interrupt routine is terminated by the RETR instruction, at least one instruction of the main program will be executed before another interrupt routine is entered.

* This CALL clears the EIF flag.
** This CALL clears the TIF flag.

FUNCTIONAL DESCRIPTION (continued)



EI External Interrupt
SI Serial Interrupt
TI Timer/counter Interrupt
EIF External Interrupt Flag
TF Timer Flag
TIF Timer Interrupt Flag
PIN Pending Interrupt NOT (SIO)
IIP Interrupt In Progress flag

Fig. 14(a) Flow chart illustrating the interrupt handling sequence.

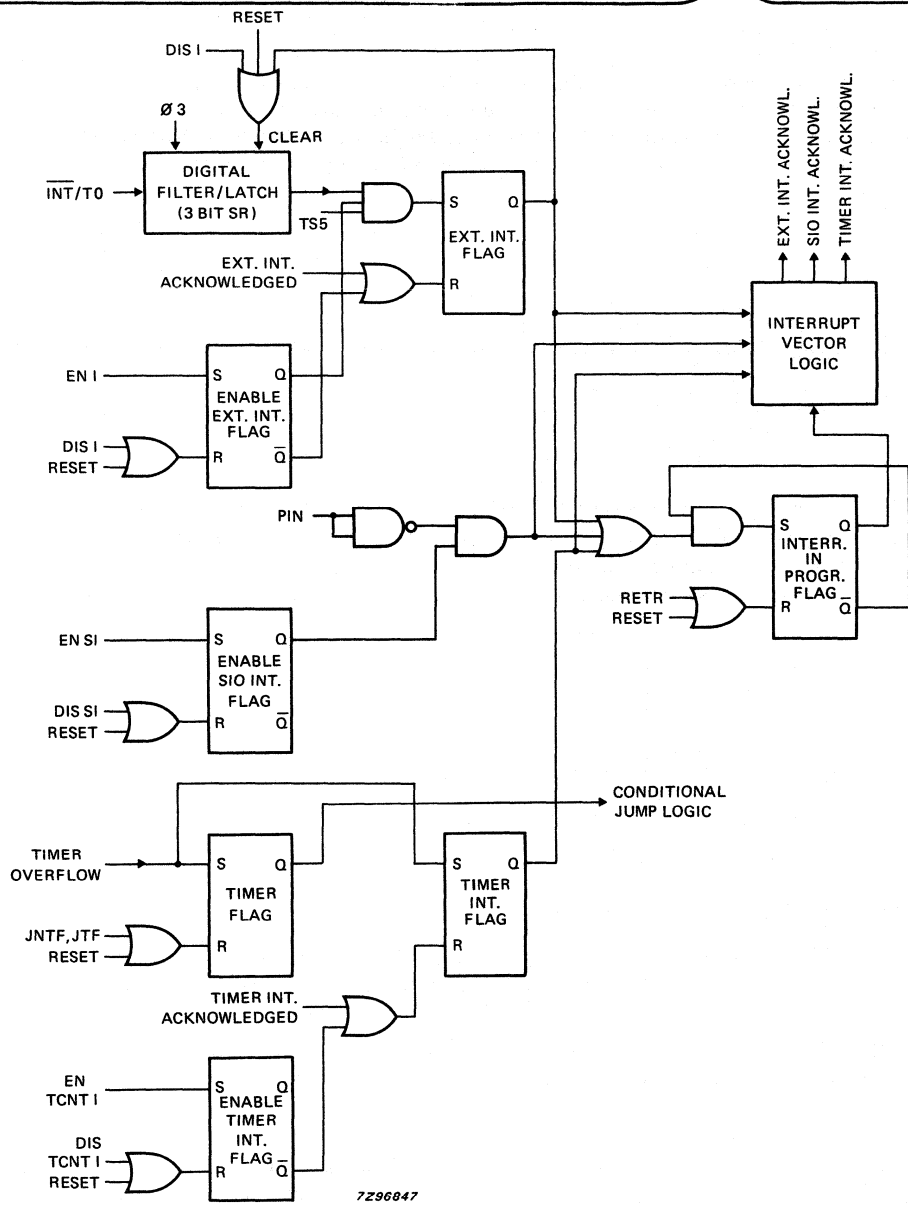


Fig. 14(b) Simplified schematic of interrupt logic, to be used in conjunction with the functional description.

Note to figure

1. $\overline{INT}$/T0 negative edge is always latched in the digital filter/latch.
2. Correct interrupt timing is ensured when $\overline{INT}$/T0 is HIGH for > 4 CP followed by a LOW for > 7 CP.
3. When the interrupt in progress flat is set, further external and timer interrupts are latched but ignored, until RETR is executed.
4. A DIS I instruction always clears a pending external interrupt.
5. For all flip-flops, RESET overrules SET.

Oscillator (see Fig. 15)

The oscillator can be inhibited by the STOP instruction under software control. It is also inhibited when a low-supply voltage condition is present to prevent discharge of a weak back-up battery. Provided the supply voltage is within the operating range the oscillator will be restarted after a STOP instruction by a LOW level at the $\overline{INT}$/T0 pin or a HIGH level at the RESET pin.

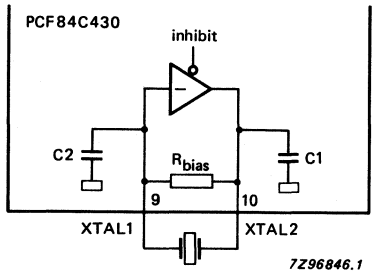


Fig. 15 Oscillator with integrated elements.

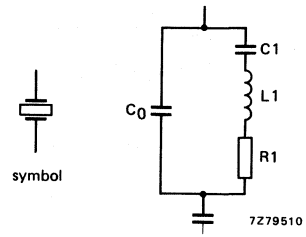


Fig. 16 Crystal unit equivalent circuit.

The values of crystal series resistance R1 and the crystal's total load capacitance C_L (C_o + wiring + external capacitors) must not be above the curve (Fig. 17) for the corresponding frequency.
Note: if external capacitors are connected to XTAL 1 and XTAL 2 they must be of equal value.

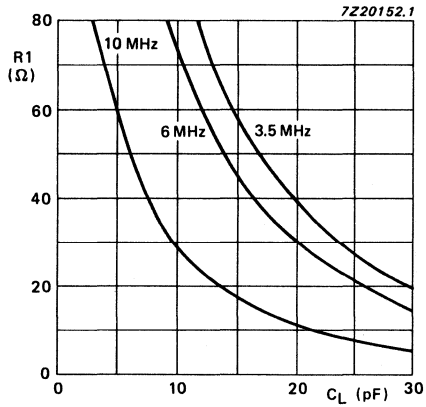


Fig. 17 Crystal circuit series resistance (R1) as a function of load capacitance (C_L).

The oscillator has the output drive capability via pin 10 (XTAL 2). An external clock can be applied to pin 9 (XTAL 1). A machine cycle consists of 10 time slots, each time slot being 3 oscillator periods.

Timer/event counter (see Fig. 18)

An internal 8-bit up-counter is provided. It can count external events, modulo-32 machine cycles, or machine cycles directly. Table 4 gives the instructions that control the counter and the prescaler, and the functions performed.

When used as a timer, the input to the counter is either the overflow or input of a 5-bit prescaler. When used as an event counter, LOW-to-HIGH transitions on pin T1 are counted. The counter is incremented during a machine cycle only if the falling edge occurs during the first 7 time slots; otherwise it is incremented during the next cycle. The maximum rate at which the counter may be incremented is once every machine cycle. When the counter overflows, the Timer flag is set. The flag can be tested and reset using the JTF (jump if Timer flag = logic 1) or JNTF (jump if Timer flag = logic 0) instruction. Overflow also generates an interrupt request to the processor via setting of the Timer Interrupt Flag when the timer/event counter interrupt is enabled.

Table 4 Timer/event counter control

function	timer mode modulo-1, modulo-32*	counter mode
CLEAR	MOV T,A (A) = 0 or RESET	MOV T,A (A) = 0 or RESET
PRESET	MOV T,A	MOV T,A
START	STRT T	STRT CNT
STOP	STOP TCNT or RESET	STOP TCNT or RESET
TEST	JTF/JNTF	JTF/JNTF
READ**	MOV A,T	MOV A,T

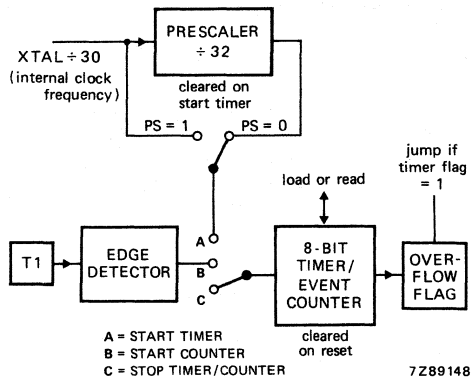


Fig. 18 Timer/event counter.

Program status word (see Fig. 19)

The program status word (PSW) is an 8-bit word (1 byte) in the CPU which stores information about the current status of the microcontroller.
The PSW bits are:

- Bits 0 to 2 — stack pointer bits (SP_0, SP_1, SP_2)
- Bit 3 — prescaler select (PS);
  0 = modulo-32; 1 = modulo-1 (no prescaling)
- Bit 4 — working register bank select (RBS);
  0 = register bank 0; 1 = register bank 1
- Bit 5 — not used (1)
- Bit 6 — auxiliary carry (AC); half-carry bit generated by an ADD instruction and used by the decimal adjust instruction DA A
- Bit 7 — carry (CY); the carry flag indicates that previous operation has resulted in an overflow of the accumulator.

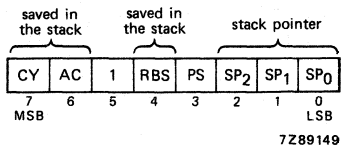


Fig. 19 Program status word.

* With prescaler select, PS = logic 0, the timer is incremented every 32 machine sycles; with PS = logic 1 the timer will be incremented every machine cycle (prescaler not used); the prescaler is cleared by the STRT T instruction and is not readable.

** READ does not disturb the counting process.

FUNCTIONAL DESCRIPTION (continued)

Program status word (continued)

All bits can be read using the MOV A, PSW instruction. Bits 7 and 6 are set and cleared by CPU operation. Bit 4 can be changed by a SEL RB instruction, bit 3 by the MOV PSW, A instruction, and bits 0, 1 and 2 by the CALL, RET or RETR instructions in the event of an interrupt. Bits 7, 6 and 4 are stored in the program counter stack during subroutine and interrupt calls. These bits are restored in the PSW with a RETR (return and restore) instruction which must be used at the end of an interrupt and can be used at the end of a normal routine, which is not part of an interrupt subroutine. The RET instruction has no restore feature and must not be used at the end of an interrupt.

Program counter (see Fig. 20)

The 12-bit program counter is able to address 4 K bytes of ROM. The arrangement of the bits is shown in Fig. 20. During an interrupt subroutine PC_{11} is forced to logic 0. All 12 bits are saved in the stack during CALL and interrupt routines.

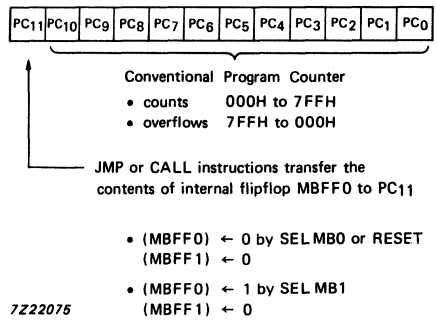


Fig. 20 Program counter.

Central processing unit

The PCF84C430 has arithmetic, logical and branching capabilities. The DA A, SWAP A and XCHD instructions simplify BCD arithmetic and the handling of nibbles. The MOVP A,@A instruction permits efficient table look-up from the current ROM page.

Conditional branch logic

The conditional branch logic within the processor enables several conditions, internal and external to the processor, to be tested by the user's program. Table 5 lists the conditional jump instructions used to change the program sequence. The DJNZ instruction decrements a designated register or data memory location and branches if the contents are not zero. This instruction is useful for looping control. The JMPP@A instruction allows multiway branches to destinations determined by the contents of the accumulator.

Table 5 Conditional branches

test	jump condition	jump instruction
accumulator	all bits zero	JZ
	any bit non-zero	JNZ
accumulator bit test	1	JB0 to JB7
carry flag	1	JC
	0	JNC
timer overflow flag	1	JTF
	0	JNTF
test input T0	1	JT0
	0	JNT0
test input T1	1	JT1
	0	JNT1
register	non-zero	DJNZ

Test input T1 (pin 7)

The T1 input line can be used as:

- A test input for branch instructions JT1 and JNT1
- An external input to the event counter

When used as a test input:

- JT1 instruction tests for logic 1 level
- JNT1 instruction tests for logic 0 level

When used as an input to the event counter, T1 must be LOW for > 4 CP, followed by a HIGH for > 4 CP. A transition can be recognized every 30 oscillator clock periods (1 machine cycle).

There is no internal pull-up or pull-down resistor connected to the T1 input. If required it must be externally connected to a resistor ($R = \leqslant 100\ k\Omega$). When T1 is not used pin 7 must be connected V_{DD} or V_{SS}.

Reset (pin 15)

A positive-going signal on the RESET input/output

- Sets the program counter to zero
- Selects location 0 of memory bank 0 and register bank 0
- Sets the stack pointer to zero (000); pointing to RAM address 8
- Disables the interrupts (external, timer and serial I/O)
- Stops the timer/event counter, then sets it to zero
- Sets the timer prescaler to divide by 32
- Resets the timer flag
- Sets all ports except P2.3 to input mode (see serial I/O section)
- Sets the serial I/O to slave receiver mode and disables the serial I/O
- Cancels IDLE and STOP mode
- Re-defines the LCD control byte

A negative-going signal on the RESET input/output:

- Sets P2.3/SDA and SCLK to HIGH after a maximum of 30 clock pulses
- Sets the serial I/O to slave receiver mode and disables the serial I/O after a maximum of 30 clock pulses
- Starts program execution after 1866 clock pulses

DEVELOPMENT DATA

FUNCTIONAL DESCRIPTION (continued)

Power-on reset

The internal power-on reset circuit monitors the PCF84C430 supply voltage V_{DD}. For as long as the supply voltage remains below the internal reference level V_{ref} (typically 1,5 V) the oscillator is inhibited and RESET (pin 15) has an undefined level. When V_{DD} rises above the internal reference level, the oscillator is released and RESET is pulled high to V_{DD} by TR1 for a period t_D (typically 50 μs).

N.B. Because of the narrow bandwidth of the crystal, the start-up time of the oscillator is typically 10 ms.

Three modes of power-on reset are possible:

1. If V_{DD} can be switched with a fast rise time i.e. V_{DD} reaches its minimum operating value (corresponding to the selected oscillator frequency) before the RESET signal (t_D) has finished, then no extra components are required (see Fig. 21 and 22). Note that the first instruction is executed after the oscillator start-up time plus 1866 clock periods have elapsed.
2. If V_{DD} has a slow rise time then the RESET signal should be stretched by an external RC circuit (see Fig. 23 and 24). In the event of a short drop in the supply voltage, the diode path rapidly discharges the capacitor to ensure a reliable power-on reset. To ensure a correct reset, the RESET signal should reach at least 70% of the final value of V_{DD}. Given that the RESET voltage and V_{DD} rise exponentially, the above requirement is satisfied when the time constant τ of the RESET pulse is >8 times the time constant of V_{DD}. If V_{DD} rises linearly, then a RESET time constant >2 times the rise time of V_{DD} is required.

   When a reset is completed (RESET goes LOW) before the oscillator has started up, program execution begins after the oscillator start-up time plus 1866 clock periods have elapsed (see Fig. 24). If the oscillator is started-up prior to the completion of RESET, then program execution begins 1866 clock periods after RESET goes LOW.
3. Figure 25 shows an external reset to the PCF84C430 during power-on. The external reset signal must remain HIGH until V_{DD} has reached its minimum operating value corresponding to the selected oscillator frequency. When a reset is completed (RESET goes LOW) before the oscillator has started up, program execution begins after the oscillator start-up time plus 1866 clock periods have elapsed (see Fig. 26). If the oscillator is started-up prior to the completion of RESET, then program execution begins 1866 clock periods after RESET goes LOW.

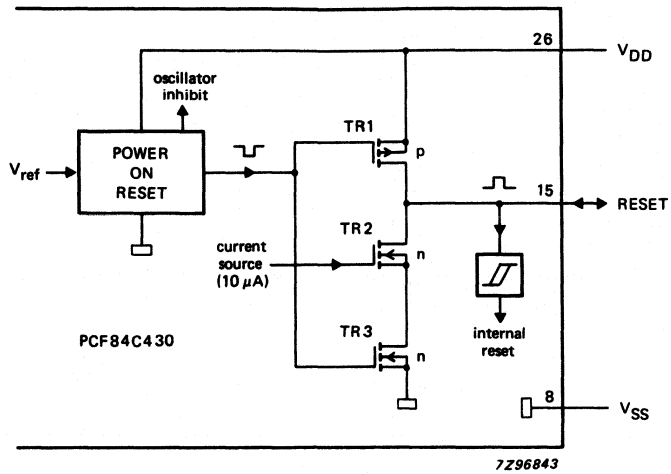


Fig. 21 Power-on-reset configuration.

DEVELOPMENT DATA

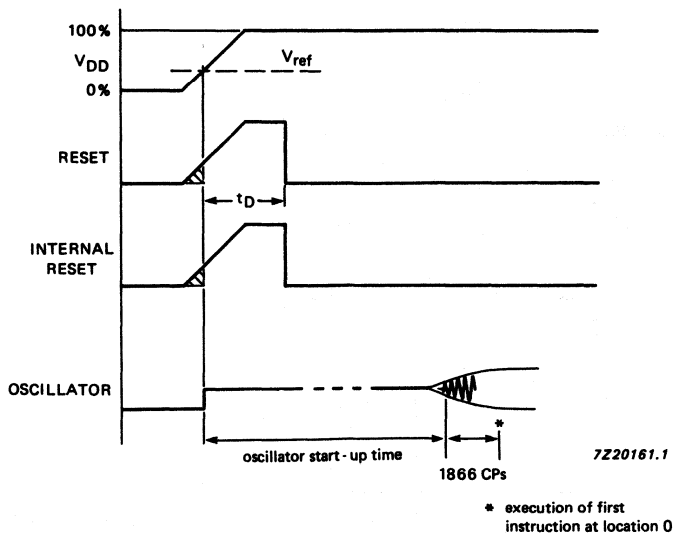


Fig. 22 Timing of power-on-reset with fast rise time.

FUNCTIONAL DESCRIPTION (continued)

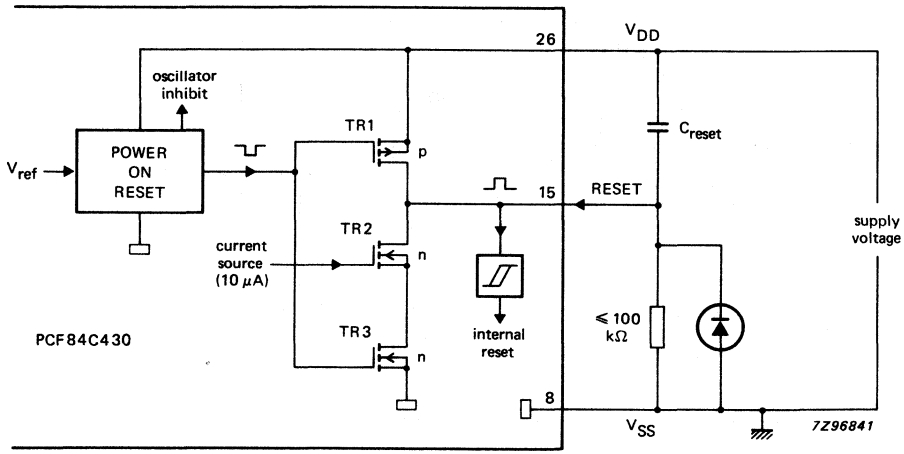


Fig. 23 Stretched power-on-reset with external components.

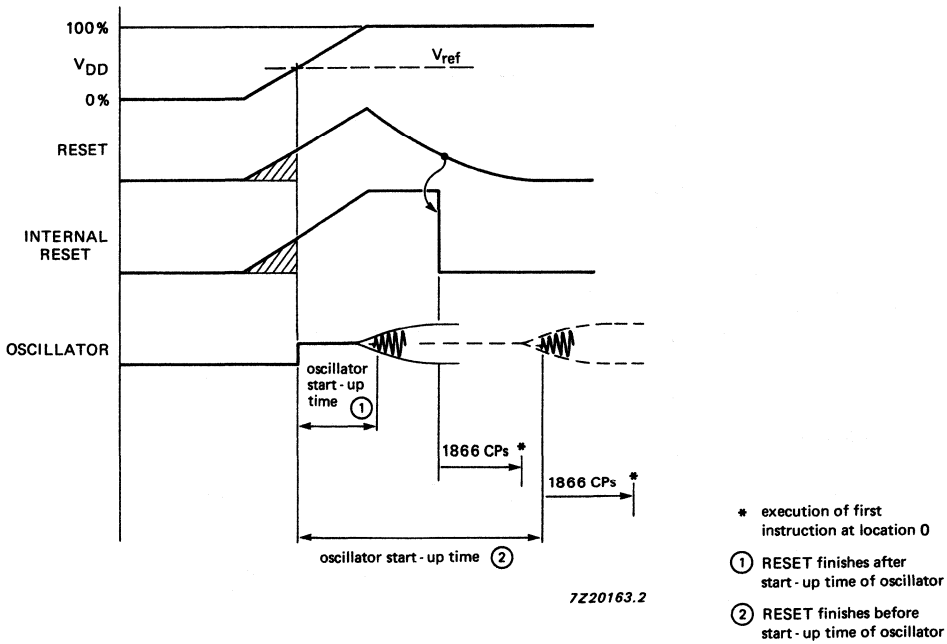


Fig. 24 Timing of power-on-reset with a slowly rising V_{DD} and a stretched RESET pulse.

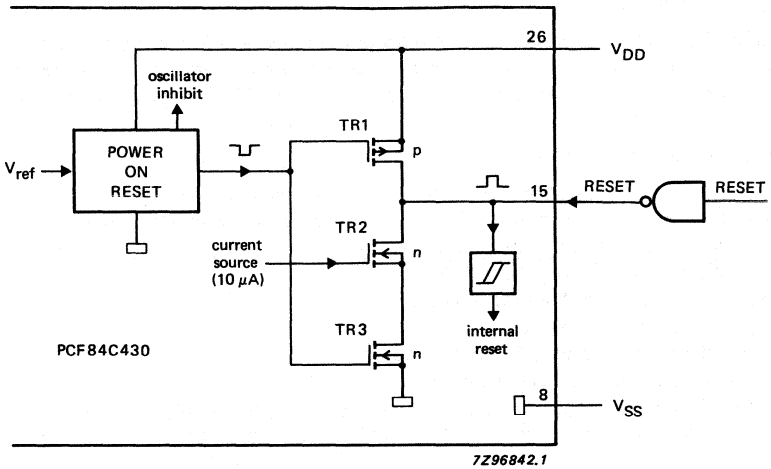


Fig. 25 External power-on-reset configuration.

DEVELOPMENT DATA

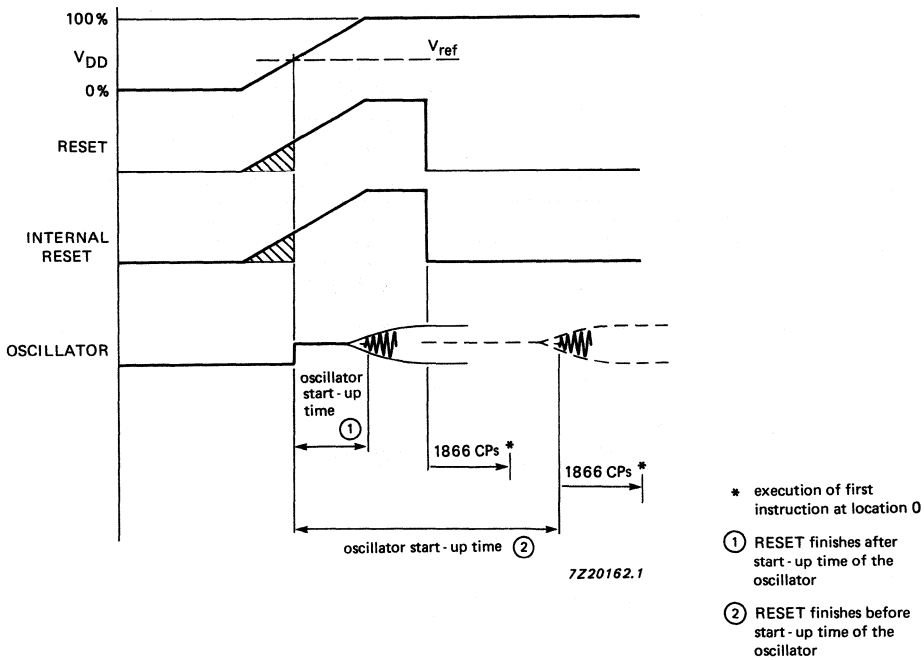


Fig. 26 Timing of external power-on-reset.

FUNCTIONAL DESCRIPTION (continued)

Liquid crystal display driver

The PCF84C430 has a display driver which interfaces to almost any liquid crystal display (LCD) which has a low multiplex rate. The interface delivers drive signals for any static or multiplexed LCD panel that contains up to four backplanes and up to 24 segments. Fig. 27 shows a block diagram of the LCD driver.

The following features are incorporated:

- Selectable backplane drive configuration; static or 2/3/4 backplane multiplexing
- Selectable display bias configuration; 1/2 or 1/3 internal LCD bias generation
- 24 individual segment drivers can be used to provide
  - up to twelve 8-segment numeric characters
  - up to six 15+1 segment alphanumeric characters
  - graphics using up to 96 elements
  - twelve 8-bit derivative registers for display data bits
- LCD and logic voltage supplies may be separated
- An LCD operating voltage range of between V_{SS} to V_{DD} −2,5 V ($V_{SS} \leqslant V_{LCD} < V_{DD}$)
- A low-power zero-component oscillator keeps the LCD display running in the STOP mode

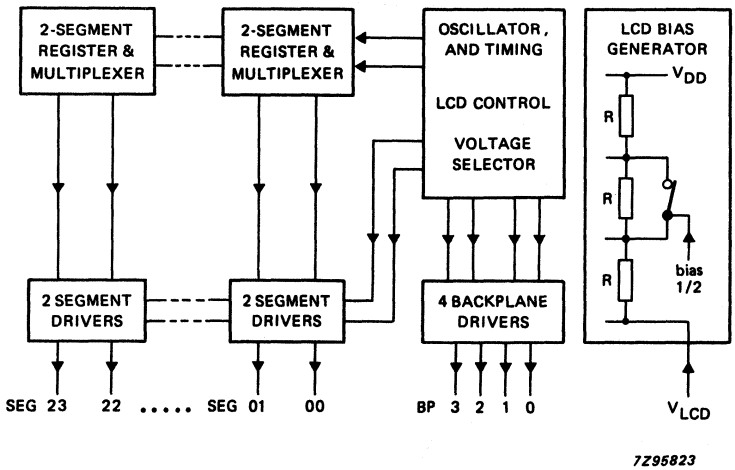


Fig. 27 Block diagram of the LCD driver.

The display configurations possible with the PCF84C430 depend upon the number of active backplane outputs required. A selection of display configurations is given in Table 6.

Table 6 Selection of display configurations

number of backplanes	number of segments	7-segment numeric	14-segment alphanumeric	dot matrix
4	96	12 digits + 12 indicator symbols	6 characters + 12 indicator symbols	96 dots (4 x 24)
3	72	9 digits + 9 indicator symbols	5 characters + 2 indicator symbols	72 dots (3 x 24)
2	48	6 digits + 6 indicator symbols	3 characters + 6 indicator symbols	48 dots (2 x 24)
1	24	3 digits + 3 indicator symbols	1 character + 10 indicator symbols	24 dots (1 x 24)

All of the display configurations given in Table 6 can be implemented in the typical system shown in Fig. 28. The appropriate biasing voltages for the multiplexed LCD wave forms are generated internally.

At power-on all the LCD driver control register bits are cleared. The LCD display is not affected by executing a STOP instruction.

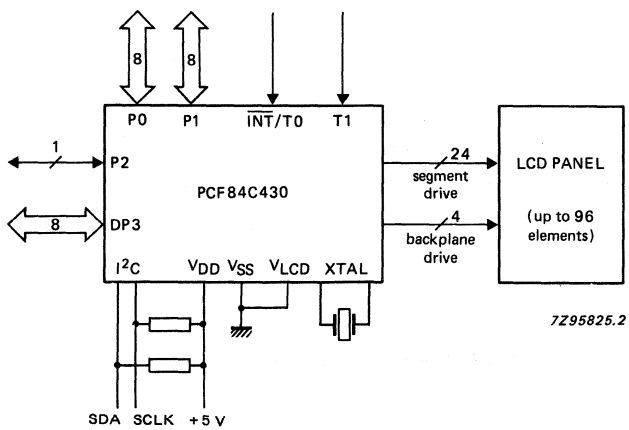


Fig. 28 Typical system configuration.

DEVELOPMENT DATA

FUNCTIONAL DESCRIPTION (continued)

LCD bias generation

The LCD operating voltage (V_{op}) is the result of V_{DD} - V_{LCD}. V_{op} should be chosen so that the off voltage ($V_{off(rms)}$) is just below the threshold voltage (V_{th}), typically when the LCD exhibits 10% contrast. The LCD voltage may be temperature compensated externally through the LCD supply. Fractional LCD biasing voltages are obtained from an internal voltage divider of three resistors connected between V_{DD} and V_{LCD}. The centre resistor may be switched out of circuit to provide a ½ bias voltage level for a 1:2 multiplex configuration.

LCD voltage selector

The LCD voltage selector coordinates the multiplexing of the LCD according to the selected drive configuration. The operation of the voltage selector is controlled by the MODE bits in the LCD control byte. The biasing configurations that apply to the preferred mode of operation, together with the biasing characteristics as functions of $V_{op} = V_{DD} - V_{LCD}$ and resulting discrimination ratios (D), are given in Table 7.

Table 7 LCD drive modes and characteristics

LCD drive mode	number of backplanes	LCD bias configuration	number of levels	$\frac{V_{off(rms)}}{V_{op}}$	$\frac{V_{on(rms)}}{V_{op}}$	$D = \frac{V_{on(rms)}}{V_{off(rms)}}$
static	1	static	2	0	1	∞
1:2	2	1/2	3	0,354	0,791	2,236
1:2	2	1/3	4	0,333	0,745	2,236
1:3	3	1/3	4	0,333	0,638	1,915
1:4	4	1/3	4	0,333	0,577	1,7321

Multiplex drive ratios of 1:3 and 1:4 with ½ bias are possible, but the discrimination and contrast ratios are reduced thus;

(1,732 for 1:3 or 1,528 for 1:4)

There is an advantage however, that these modes lead to a reduction in V_{op} as follows:

1:3 multiplex (½ bias). $V_{op} = 2{,}449\ V_{off(rms)}$

1:4 multiplex (½ bias). $V_{op} = 2{,}309\ V_{off(rms)}$

This compares with $V_{op} = 3\ V_{off(rms)}$ when 1/3 bias is used.

LCD driver (continued)

LCD drive mode waveforms

The static LCD drive mode is used when a single backplane is provided in the LCD. Backplane and segment drive waveforms are shown in Fig. 29.

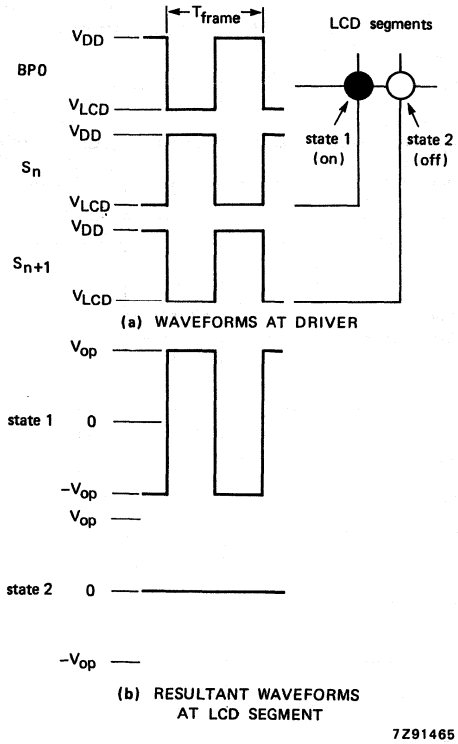


Fig. 29 Static drive mode waveforms.

DEVELOPMENT DATA

FUNCTIONAL DESCRIPTION (continued)

When two backplanes are provided in the LCD; the 1:2 multiplex drive mode applies. The PCF84C430 allows use of 1/2 or 1/3 bias in this mode as shown in Figs 30 and 31.

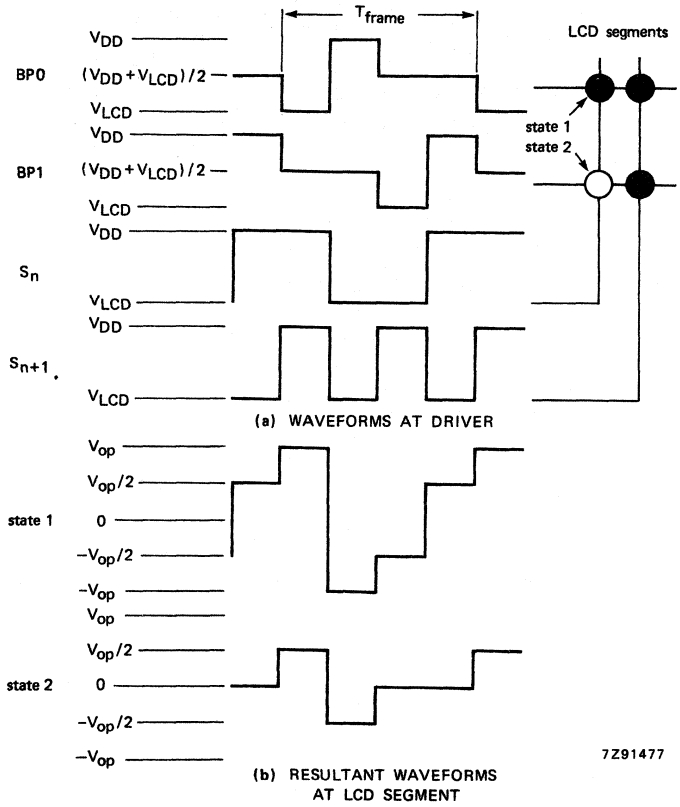


Fig. 30 Waveforms for the 1:2 multiplex drive mode with 1/2 bias.

LCD driver (continued)

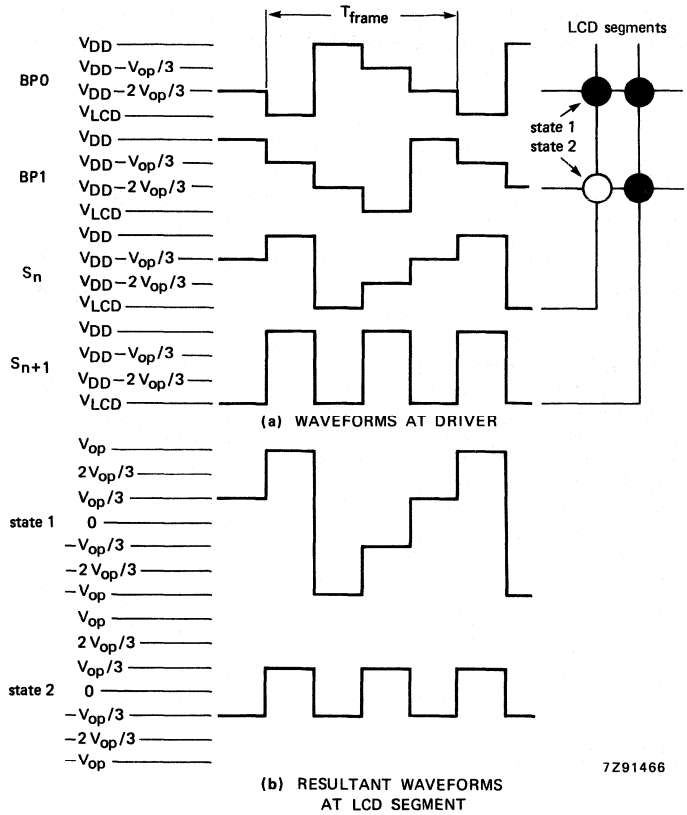


Fig. 31 Waveforms for the 1:2 multiplex drive mode with 1/3 bias.

DEVELOPMENT DATA

FUNCTIONAL DESCRIPTION (continued)

The backplane and segment drive waveforms for the 1:3 multiplex drive mode (three LCD backplanes) and for the 1:4 multiplex drive mode (four LCD backplanes) are shown in Figs 32 and 33 respectively.

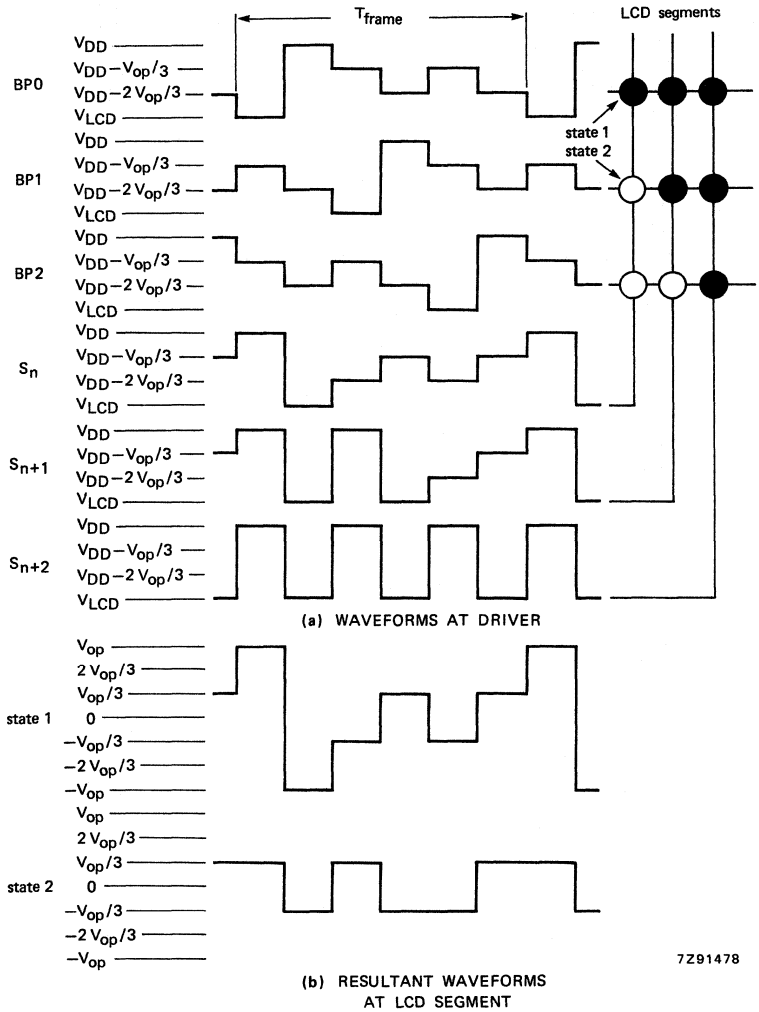


Fig. 32 Waveforms for the 1:3 multiplex drive mode.

LCD driver (continued)

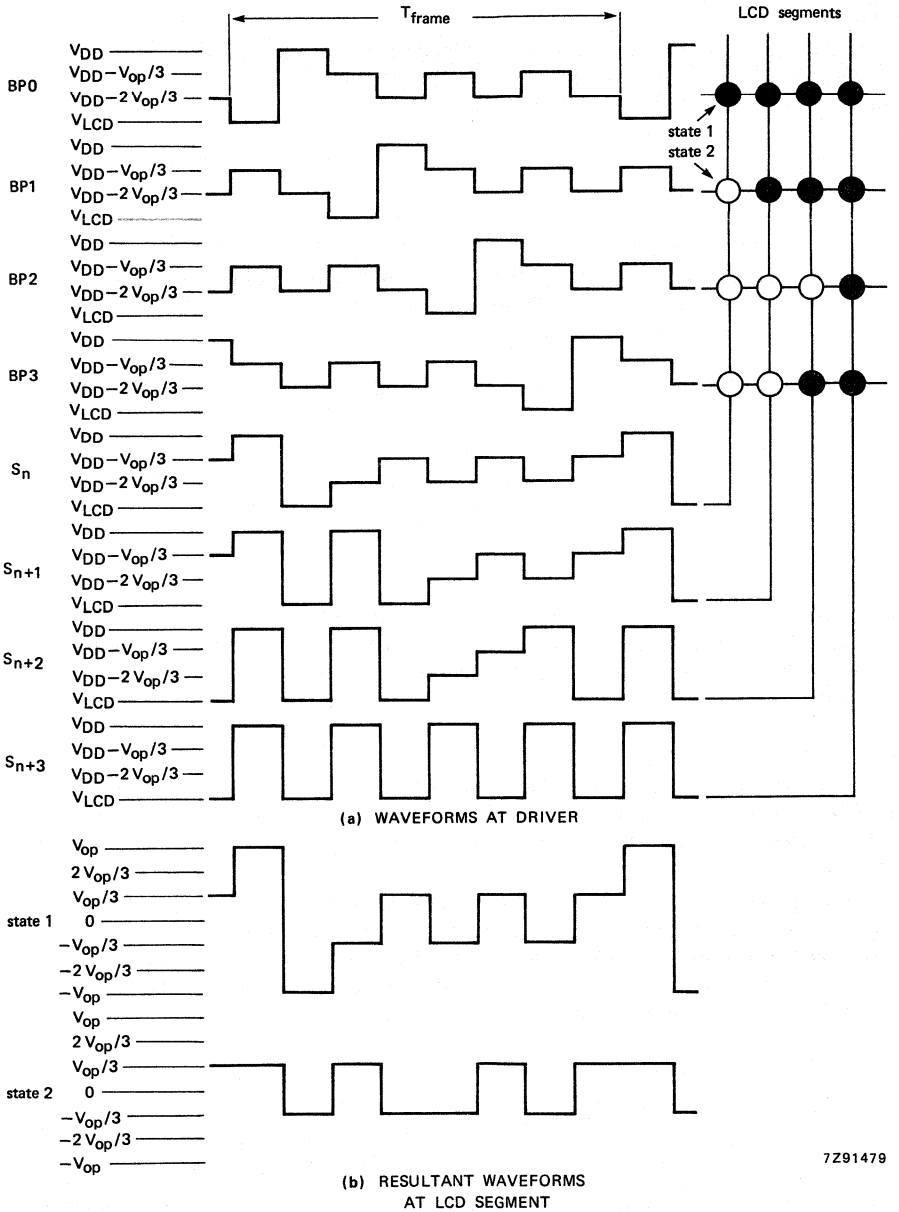


Fig. 33 Waveforms for the 1:4 multiplex drive mode.

FUNCTIONAL DESCRIPTION (continued)

LCD timing

The LCD clocking is performed by a separate, self-contained, on-chip oscillator with a nominal frequency of 30 kHz. The display may be kept active while in the STOP mode using a very low supply current. Fig. 34 shows the LCD timing control.

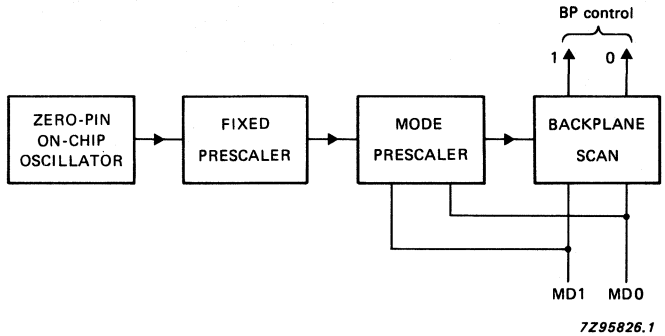


Fig. 34 LCD timing control.

LCD segment driver outputs

The LCD drive section includes 24 segment outputs (S0 to S23) which should be connected directly to the LCD. The segment data bits (one nibble per segment) are multiplexed to the outputs in accordance with the backplane signals. If less than the 24 segment outputs are required then the unused driver outputs should be left open.

Backplane outputs

The LCD drive section includes 4 backplane outputs (BP0-BP3) which should be connected directly to the LCD. These backplane output signals are generated in accordance with the selected LCD drive mode. If less than four backplane outputs are required then the unused driver outputs should be left open-circuit.

In the 1:3 multiplex drive mode, BP3 carries the same signal as BP0, therefore these two outputs can be tied together to give enhanced drive capabilities. In the 1:2 multiplex drive mode, BP0 and BP3, BP1 and BP2 respectively carry the same signals and may also be paired to increase the drive capabilities. In the static drive mode the same signal is carried by all four backplane outputs and may be connected in parallel to give a very high drive capability.

LCD driver (continued)

LCD segment display registers

The 12 segment display registers are 8-bit derivative (read/write) registers which store LCD segment data. A segment register bit which is set to a logic 1 indicates the 'on' state of the corresponding LCD segment, similarly, a logic 0 indicates the 'off' state. There is a one-to-one relationship between the LCD segment register bits and the segment outputs. Every byte is divided into two nibbles. Each nibble corresponds to a segment driver; the first byte contains the bits earmarked for segment 1 and 2 etc. The first bit of a nibble will correspond to backplane 0, and the second to backplane 1 and so on. Fig. 35 shows the display register bit map. Each bit is shown in the form Sxx.n, where (xx) is the segment output and (n) the backplane output.

LCDD : LCD display data

17	R/W	S 0.0	S 0.1	S 0.2	S 0.3	S 1.0	S 1.1	S 1.2	S 1.3
18	R/W	S 2.0	S 2.1	S 2.2	S 2.3	S 3.0	S 3.1	S 3.2	S 3.3
19	R/W	S 4.0	S 4.1	S 4.2	S 4.3	S 5.0	S 5.1	S 5.2	S 5.3
20	R/W	S 6.0	S 6.1	S 6.2	S 6.3	S 7.0	S 7.1	S 7.2	S 7.3
21	R/W	S 8.0	S 8.1	S 8.2	S 8.3	S 9.0	S 9.1	S 9.2	S 9.3
22	R/W	S10.0	S10.1	S10.2	S10.3	S11.0	S11.1	S11.2	S11.3
23	R/W	S12.0	S12.1	S12.2	S12.3	S13.0	S13.1	S13.2	S13.3
24	R/W	S14.0	S14.1	S14.2	S14.3	S15.0	S15.1	S15.2	S15.3
25	R/W	S16.0	S16.1	S16.2	S16.3	S17.0	S17.1	S17.2	S17.3
26	R/W	S18.0	S18.1	S18.2	S18.3	S19.0	S19.1	S19.2	S19.3
27	R/W	S20.0	S20.1	S20.2	S20.3	S21.0	S21.1	S21.2	S21.3
28	R/W	S22.0	S22.1	S22.2	S22.3	S23.0	S23.1	S23.2	S23.3

Fig. 35 Display register bit map.

In the static drive mode the eight display data bits are placed in bit 0 of the nibbles of four successive derivative display registers. In the 1:2 multiplex drive mode, these eight bits are placed in bits 0 and 1 of the nibbles of two successive registers. In the 1:3 multplex drive mode the eight bits are placed in

FUNCTIONAL DESCRIPTION (continued)

bits 0, 1 and 2 of three successive nibbles. In the 1:4 multiplex drive mode the eight bits are placed in bits 0, 1, 2 and 3 of the nibbles of the first display register. Fig. 36 shows the relationship between LCD segment layout, drive mode and the LCD segment derivative register bit pattern.

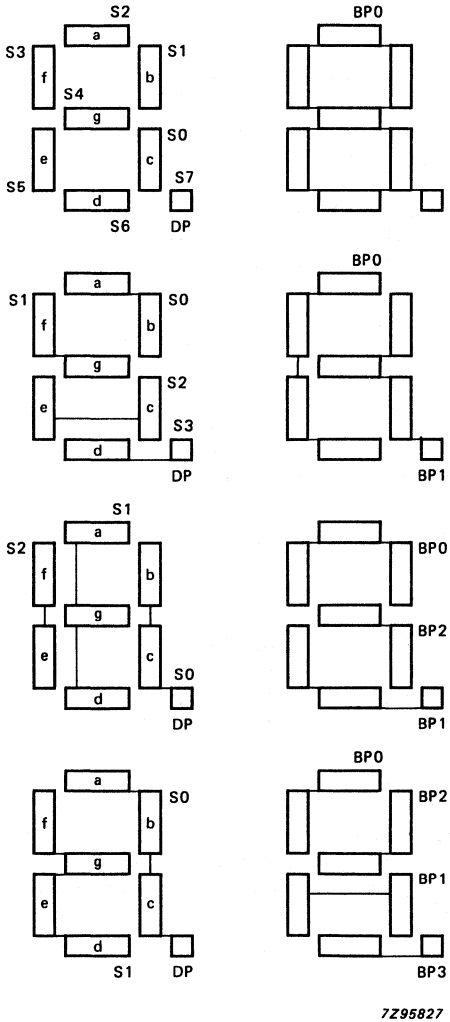


Fig. 36 LCD segment layout and register bit pattern.

LCD derivative register bit map

byte

17	c	–	–	–	b	–	–	–
18	a	–	–	–	f	–	–	–
19	g	–	–	–	e	–	–	–
20	d	–	–	–	Dp	–	–	–

static mode

byte

17	a	b	–	–	f	g	–	–
18	e	c	–	–	d	Dp	–	–
19	–	–	–	–	–	–	–	–
20	–	–	–	–	–	–	–	–

1:2 multiplex mode

byte

17	b	Dp	c	–	a	d	g	–
18	f	e	–	–	–	–	–	–
19	–	–	–	–	–	–	–	–
20	–	–	–	–	–	–	–	–

1:3 multiplex mode

byte

17	a	c	b	Dp	f	e	g	d
18	–	–	–	–	–	–	–	–
19	–	–	–	–	–	–	–	–
20	–	–	–	–	–	–	–	–

1:4 multiplex mode

LCD driver (continued)

LCD control register

The LCD operating mode is governed by four bits of the LCD control register (address 16 D). The LCD control register is an 8-bit derivative read/write register. The function of each bit is given in Table 8.

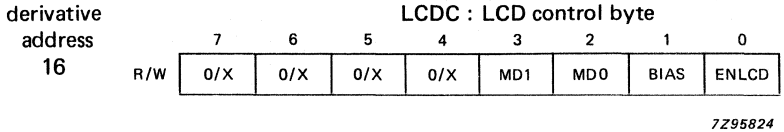


Table 8 Control register bit definition

bit	name	function		
0	ENLCD	0 = LCD display disabled 1 = LCD display enabled		
1	BIAS	0 = 1/3 V_{DD} 1 = 1/2 V_{DD}		
2, 3	MD0/MD1	MD1	MD0	multiplex mode
		0 0 1 1	0 1 0 1	static 1:2 1:3 1:4
4-7	unused			

ENLCD: When ENLCD is reset to 0, the LCD is disabled. Consequently all segment and backplane drivers are set to the V_{DD} level. When ENLCD is set to a 1, the LCD is enabled and character display is possible.

BIAS: The BIAS bit sets the LCD voltage bias generator to either 1/2 or 1/3 of V_{DD}, see Fig. 27.

MD0/MD1: Mode bits MD0 and MD1 determine the multiplex rate. Four multiplex rates are available; static, 1:2, 1:3 and 1:4.

DEVELOPMENT DATA

FUNCTIONAL DESCRIPTION (continued)

LCD control byte after RESET

After an external or power-on reset the LCD control register is set with 0C H.

	7	6	5	4	3	2	1	0
	0/X	0/X	0/X	0/X	MD1	MD0	BIAS	ENLCD
reset value	0	0	0	0	1	1	0	0

7Z97990

- The LCD is disabled and segment and backplane drives are switched to the V_{DD} level.
- BIAS is set to generate 1/3 V_{DD}.
- Bits MD0 and MD1 reset the multiplex mode to the 1:4 mode.

INSTRUCTION SET

The PCF84C430 instruction set consists of over 80 one and two byte instructions. Program code efficiency is high because all RAM locations and all ROM locations on a 256 byte page require only a single byte address.

Table 11 gives the instruction set of the PCF84C430. Table 10 shows the instruction map and Table 9 details the symbols and definition descriptions that are used.

Table 9 Symbols and definitions used in Table 11

symbol	definition description
A	accumulator
addr	program memory address
Bb	bit designation (b = 0-7)
RBS	register bank select
C	carry bit (bit CY)
CNT	event counter
Dx	Derivative register designation (x = 4,5,16,17-28)
data	8-bit number or expression
I	interrupt
MB	memory bank
MBFF	memory bank flip-flop
P	mnemonic for 'in-page' operation
PC	program counter
Pp	port designation (p = 0, 1 or 2)
PSW	program status word
RB	register bank
Rr	register designation (r = 0-7)
Sn	serial I/O register
SP	stack pointer
T	timer
TF	timer flag
T0, T1	test 0 and 1 inputs
#	immediate data prefix
@	indirect address prefix
(X)	contents of X
((X))	contents of location addressed by X
←	is replaced by
↔	is exchanged with

DEVELOPMENT DATA

INSTRUCTION SET (continued)

Table 10 PCF84C430 instruction map

first hexadecimal character of opcode

second hexadecimal character of opcode

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	NOP	IDLE		ADD A,#data	JMP page 0	EN I	JNTF addr	DEC A	IN A,Pp				MOV A,Sn			
									0	1	2		0	1		
1	INC @Rr		JBO addr	ADDC A,#data	CALL page 0	DIS I	JTF addr	INC A	INC Rr							
	0	1							0	1	2	3	4	5	6	7
2	XCH A,@Rr		STOP	MOV A,#data	JMP page 1	EN TCNTI	JNTO addr	CLR A	XCH A,Rr							
	0	1							0	1	2	3	4	5	6	7
3	XCHD A,@Rr		JB1 addr		CALL page 1	DIS TCNTI	JTO addr	CPL A	OUTL Pp,A				MOV Sn,A			
	0	1							0	1	2		0	1	2	
4	ORL A,@Rr		MOV A,T	ORL A,#data	JMP page 2	STRT CNT	JNT1 addr	SWAP A	ORL A,Rr							
	0	1							0	1	2	3	4	5	6	7
5	ANL A,@Rr		JB2 addr	ANL A,#data	CALL page 2	STRT T	JT1 addr	DA A	ANL A,Rr							
	0	1							0	1	2	3	4	5	6	7
6	ADD A,@Rr		MOV T,A		JMP page 3	STOP TCNT		RRC A	ADD A,Rr							
	0	1							0	1	2	3	4	5	6	7
7	ADDC A,@Rr		JB3 addr		CALL page 3			RR A	ADDC A,Rr							
	0	1							0	1	2	3	4	5	6	7
8				RET	JMP page 4	EN SI			ORL Pp,#data				MOV A,Dx	MOV Dx,A	ANL Dx,A	ORL Dx,A
									0	1	2					
9			JB4 addr	RETR	CALL page 4	DIS SI	JNZ addr	CLR C	ANL Pp,#data				MOV Sn,#data			
									0	1	2		0	1	2	
A	MOV @Rr,A			MOVP A,@A	JMP page 5			CPL C	MOV Rr,A							
	0	1							0	1	2	3	4	5	6	7
B	MOV @Rr,#data		JB5 addr	JMPP @A	CALL page 5				MOV Rr,#data							
	0	1							0	1	2	3	4	5	6	7
C	DEC @Rr				JMP page 6	SEL RB0	JZ addr	MOV A,PSW	DEC Rr							
	0	1							0	1	2	3	4	5	6	7
D	XRL A,@Rr		JB6 addr	XRL A,#data	CALL page 6	SEL RB1		MOV PSW,A	XRL A,Rr							
	0	1							0	1	2	3	4	5	6	7
E	DJNZ @Rr,addr				JMP page 7	SEL MB0	JNC addr	RL A	DJNZ Rr, addr							
	0	1							0	1	2	3	4	5	6	7
F	MOV A,@Rr		JB7 addr		CALL page 7	SEL MB1	JC addr	RLC A	MOV A,Rr							
	0	1							0	1	2	3	4	5	6	7

Table 11 Instruction set

	mnemonic	opcode (hex.)	bytes/ cycles	description	function		notes
ACCUMULATOR	ADD A, Rr	6*	1/1	Add register contents to A	(A)←(A) + (Rr)	r = 0–7	1
	ADD A, @Rr	60 61	1/1	Add RAM data, addressed by Rr, to A	(A)←(A) + ((R0)) (A)←(A) + ((R1))		1
	ADD A, #data	03 data	2/2	Add immediate data to A	(A)←(A) + data		1
	ADDC A, Rr	7*	1/1	Add carry and register contents to A	(A)←(A) + (Rr) + (C)	r = 0–7	1
	ADDC A, @Rr	70 71	1/1	Add carry and RAM data, addressed by Rr, to A	(A)←(A) + ((R0)) + (C) (A)←(A) + ((R1)) + (C)		1
	ADDC A, #data	13 data	2/2	Add carry and immediate data to A	(A)←(A) + data + (C)		1
	ANL A, Rr	5*	1/1	'AND' Rr with A	(A)←(A) AND (Rr)	r = 0–7	
	ANL A, @Rr	50 51	1/1	'AND' RAM data, addressed by Rr, with A	(A)←(A) AND ((R0)) (A)←(A) AND ((R1))		
	ANL A, #data	53 data	2/2	'AND' immediate data with A	(A)←(A) AND data		
	ORL A, Rr	4*	1/1	'OR' Rr with A	(A)←(A) OR (Rr)	r = 0–7	
	ORL A, @Rr	40 41	1/1	'OR' RAM data, addressed by Rr, with A	(A)←(A) OR ((R0)) (A)←(A) OR ((R1))		
	ORL A, #data	43 data	2/2	'OR' immediate data with A	(A)←(A) OR data		
	XRL A, Rr	D*	1/1	'XOR' Rr with A	(A)←(A) XOR (Rr)	r = 0–7	
	XRL A, @Rr	D0 D1	1/1	'XOR' RAM, addressed by Rr, with A	(A)←(A) XOR ((R0)) (A)←(A) XOR ((R1))		
	XRL A, #data	D3 data	2/2	'XOR' immediate data with A	(A)←(A) XOR data		
	INC A	17	1/1	increment A by 1	(A)←(A) + 1		
	DEC A	07	1/1	decrement A by 1	(A)←(A) −1		
	CLR A	27	1/1	clear A to zero	(A)←0		
	CPL A	37	1/1	one's complement A	(A)←NOT(A)		
	RL A	E7	1/1	rotate A left	$(A_{n+1})\leftarrow(A_n)$ $(A_0)\leftarrow(A_7)$	n = 0–6	

INSTRUCTION SET (continued)

ACCUMULATOR (cont.)	RLC A	F7	1/1	rotate A left through carry	$(A_{n+1})\leftarrow A_n$ $(A_0)\leftarrow(C)$, $(C)\leftarrow(A_7)$	n = 0–6	2
	RR A	77	1/1	rotate A right	$(A_n)\leftarrow(A_{n+1})$ $(A_7)\leftarrow(A_0)$	n = 0–6	
	RRC A	67	1/1	rotate A right through carry	$(A_n)\leftarrow(A_{n+1})$ $(A_7)\leftarrow(C)$, $(C)\leftarrow(A_0)$	n = 0–6	2
	DA A	57	1/1	decimal adjust A			2
	SWAP A	47	1/1	swap nibbles of A	$(A_{4-7})\leftrightarrow(A_{0-3})$		
DATA MOVES	MOV A, Rr	F*	1/1	move register contents to A	(A)←(Rr)	r = 0–7	
	MOV A, @Rr	F0 F1	1/1	move RAM data, addressed by Rr, to A	(A)←((R0)) (A)←((R1))		
	MOV A, #data	23 data	2/2	move immediate data to A	(A)←data		
	MOV Rr, A	A*	1/1	move accumulator contents to register	(Rr)←(A)	r = 0–7	
	MOV @Rr, A	A0 A1	1/1	move accumulator contents to RAM location addressed by Rr	((R0))←(A) ((R1))←(A)		
	MOV Rr, #data	B* data	2/2	move immediate data to Rr	(Rr)←data		
	MOV @Rr, #data	B0 data B1 data	2/2	move immediate data to RAM location addressed by Rr	((R0))←data ((R1))←data		
	XCH A, Rr	2*	1/1	exchange accumulator contents with Rr	(A)↔(Rr)	r = 0–7	
	XCH A, @Rr	20 21	1/1	exchange accumulator contents with RAM data addressed by Rr	(A)↔((R0)) (A)↔((R1))		
	XCHD A, @Rr	30 31	1/1	exchange lower nibbles of A and RAM data addressed by Rr	$(A_{0-3})\leftrightarrow((R0_{0-3}))$ $(A_{0-3})\leftrightarrow((R1_{0-3}))$		
	MOV A, PSW	C7	1/1	move PSW contents to accumulator	(A)←(PSW)		
	MOV PSW, A	D7	1/1	move accumulator bit 3 to PSW_3	$(PSW_3)\leftarrow(A_3)$		3
	MOVP A, @A	A3	1/2	move indirectly addressed data in current page to A	$(PC_{0-7})\leftarrow(A)$, (A)←((PC))		
FLAGS	CLR C	97	1/1	clear carry bit	(C)←0		2
	CPL C	A7	1/1	complement carry bit	(C)←NOT(C)		2

	mnemonic	opcode (hex.)	bytes/ cycles	description	function	notes
REGISTER	INC Rr	1*	1/1	increment register by 1	(Rr)←(Rr) + 1 r = 0–7	
	INC @Rr	10 11	1/1	increment RAM data, addressed by Rr, by 1	((R0))←((R0)) + 1 ((R1))←((R1)) + 1	
	DEC Rr	C*	1/1	decrement register by 1	(Rr)←(Rr) – 1 r = 0–7	
	DEC @Rr	C0 C1	1/1	decrement RAM data, addressed by Rr, by 1	((R0))←((R0)) – 1 ((R1))←((R1)) – 1	
BRANCH	JMP addr	● 4 addr	2/2	unconditional jump within a 2 K bank	(PC_{8-10})←$addr_{8-10}$ (PC_{0-7})←$addr_{0-7}$ (PC_{11-12})←MBFF 0–1	
	JMPP @A	B3	1/2	indirect jump within a page	(PC_{0-7})←((A))	
	DJNZ Rr, addr	E* addr	2/2	decrement Rr by 1 and jump if not zero to addr	(Rr)←(Rr) – 1 r = 0–7 if (Rr) not zero (PC_{0-7})←addr	
	DJNZ @Rr, addr	E0 addr	2/2	decrement RAM data, addressed by Rr by 1 and jump if not zero to addr	((R0))←((R0)) – 1 if ((R0)) not zero (PC_{0-7})←addr	
		E1 addr			((R1))←((R1)) – 1 if ((R1)) not zero (PC_{0-7})←addr	
	JBb addr	▲ 2 addr	2/2	jump to addr if Acc. bit b = 1	if b = 1 : (PC_{0-7})←addr b = 0–7	
	JC addr	F6 addr	2/2	jump to addr if C = 1	if C = 1 : (PC_{0-7})←addr	
	JNC addr	E6 addr	2/2	jump to addr if C = 0	if C = 0 : (PC_{0-7})←addr	
	JZ addr	C6 addr	2/2	jump to addr if A = 0	if A = 0 : (PC_{0-7})←addr	
	JNZ addr	96 addr	2/2	jump to addr if A is NOT zero	if A≠0 : (PC_{0-7})←addr	
	JT0 addr	36 addr	2/2	jump to addr if T0 = 1	if T0 = 1: (PC_{0-7})←addr	
	JNT0 addr	26 addr	2/2	jump to addr if T0 = 0	if T0 = 0: (PC_{0-7})←addr	
	JT1 addr	56 addr	2/2	jump to addr if T1 = 1	if T1 = 1: (PC_{0-7})←addr	
	JNT1 addr	46 addr	2/2	jump to addr if T1 = 0	if T1 = 0: (PC_{0-7})←addr	
	JTF addr	16 addr	2/2	jump to addr if Timer Flag = 1	if TF = 1: (PC_{0-7})←addr	4
	JNTF addr	06 addr	2/2	jump to addr if Timer Flag = 0	if TF = 0: (PC_{0-7})←addr	

INSTRUCTION SET (continued)

TIMER/EVENT COUNTER	MOV A, T	42	1/1	move timer/event counter contents to accumulator	(A)←(T)	
	MOV T, A	62	1/1	move accumulator contents to timer/event counter	(T)←(A)	
	STRT CNT	45	1/1	start event counter		
	STRT T	55	1/1	start timer		
	STOP TCNT	65	1/1	stop timer/event counter		
	EN TCNTI	25	1/1	enable timer/event counter interrupt		
	DIS TCNTI	35	1/1	disable timer/event counter interrupt		
CONTROL	EN I	05	1/1	enable external interrupt		
	DIS I	15	1/1	disable external interrupt		
	SEL RB0	C5	1/1	select register bank 0	(RBS)←0	5
	SEL RB1	D5	1/1	select register bank 1	(RBS)←1	5
	SEL MB0	E5	1/1	select program memory bank 0	(MBFF0)←0, (MBFF1)←0	10
	SEL MB1	F5	1/1	select program memory bank 1	(MBFF0)←1, (MBFF1)←0	10
	STOP	22	1/1	enter STOP mode		
	IDLE	01	1/1	enter IDLE mode		
SUBROUTINE	CALL addr	▲ 4 addr	2/2	jump to subroutine	((SP))←(PC), ($PSW_{4, 6, 7}$) 6 (SP)←(SP) + 1 (PC_{8-10})←$addr_{8-10}$ (PC_{0-7})←$addr_{0-7}$ (PC_{11-12})←MBFF 0–1	6
	RET	83	1/2	return from subroutine	(SP)←(SP) – 1 (PC)←((SP))	6
	RETR	93	1/2	return from interrupt and restore bits 4, 6, 7 of PSW	(SP)←(SP) – 1 ($PSW_{4, 6, 7}$) + (PC)←((SP))	6

DEVELOPMENT DATA

	mnemonic	opcode (hex.)	bytes/ cycles	description	function	notes
PARALLEL INPUT/OUTPUT	IN A, Pp	08 09 0A	1/2	input port p data to accumulator	(A)←(P0) (A)←(P1) (A)←(P2)	7
	OUTL Pp, A	38 39 3A	1/2	output accumulator data to port p	(P0)←(A) (P1)←(A) (P2)←(A)	
	ANL Pp, #data	98 data 99 data 9A data	2/2	AND port p data with immediate data	(P0)←(P0) AND data (P1)←(P1) AND data (P2)←(P2) AND data	
	ORL Pp, #data	88 data 89 data 8A data	2/2	OR port p data with immediate data	(P0)←(P0) OR data (P1)←(P1) OR data (P2)←(P2) OR data	
DERIVATIVE INPUT/OUTPUT	MOV A, Dx	8C Dx	2/2	Move derivative register/port contents addressed by Dx to accumulator	(A)←(Dx) x = 4,5,16,17-28	8
	MOV Dx, A	8D Dx	2/2	Move contents of accumulator to derivative register addressed by Dx	(Dx)←(A) x = 4,16,17-28	8
	ANL Dx, A	8E Dx	2/2	AND contents of accumulator with derivative register addressed by Dx	(Dx)←(Dx) AND (A) x = 4,16,17-28	8
	ORL Dx, A	8F Dx	2/2	OR contents of accumulator with derivative register addressed by Dx	(Dx)←(Dx) OR (A) x = 4,16,17-28	8

INSTRUCTION SET (continued)

	mnemonic	opcode (hex.)	bytes/ cycles	description	function	notes
SERIAL INPUT/OUTPUT	MOV A, S_n	0C 0D	1/2	move serial I/O register contents to accumulator	(A)←(S0) (A)←(S1)	9
	MOV S_n, A	3C 3D 3E	1/2	move accumulator contents to serial I/O register	(S0)←(A) (S1)←(A) (S2)←(A)	
	MOV S_n, #data	9C data 9D data 9E data	2/2	move immediate data to serial I/O register	(S0)←data (S1)←data (S2)←data	
	EN SI	85	1/1	enable serial I/O interrupt		
	DIS SI	95	1/1	disable serial I/O interrupt		
	NOP	00	1/1	no operation		

Notes to Table 8

1. PSW CY, AC affected
2. PSW CY affected
3. PSW PS affected
4. Execution of JTF and JNTF instructions resets the Timer Flag (TF).

* : 8,9,A,B,C,D,E,F
● : 0,2,4,6,8,A,C,E
▲ : 0,3,5,7,9,B,D,F

5. PSW RBS affected
6. PSW SP_0, SP_1, SP_2 affected
7. (A) = 0000 (P23) 111
8. Dx = 04, 05, 16, 17-28
9. (S1) has a different function in read and write operations, see serial I/O interface.
10. SEL MB0 and SEL MB1 must not be used within interrupt routines.

RATINGS

Limiting values in accordance with the Absolute Maximum System (IEC 134)

parameter	symbol	min.	max.	unit
Supply voltage (pin 26)	V_{DD}	−0,8	+ 8	V
All input voltages	V_I	−0,8	V_{DD} + 0,8	V
DC current into input or output	$\pm I_I \pm I_O$	–	10	mA
Power dissipation per output	P_O	–	50	mW
Storage temperature	T_{stg}	−65	+ 150	°C
Ambient operating temperature range (if P_{tot}max. = 100 mW)	T_{amb}	−40	+ 70	°C
Ambient operating temperature range (if P_{tot}max. = 30 mW)	T_{amb}	−40	+ 85	°C
Operating junction temperature	T_j	–	90	°C

HANDLING

Inputs and outputs are protected against electrostatic discharge in normal handling. However, to be totally safe, it is desirable to take normal precautions appropriate to handling MOS devices (see 'Handling MOS devices').

DEVELOPMENT DATA

LCD DRIVER CHARACTERISTICS

V_{DD} = 2,5 to 5,5 V; V_{SS} = 0 V; T_{amb} = −40 to + 85 °C. All voltages with respect to V_{SS} unless otherwise specified.

parameter	symbol	min.	typ.	max.	unit
LCD supply voltage (note 1)	V_{LCD}	V_{SS}	–	V_{DD}−2,5	V
DC voltage component (BP0 to BP3) at V_{DD} = 5 V	$+/-V_{BP}$	0	20	–	mV
DC voltage component (S0 to S23) at V_{DD} = 5 V	$+/-V_S$	–	20	–	mV
Output impedance (BP0 to BP3) at $V_{LCD} = V_{SS}$ (note 2)	R_{BP}	–	–	5	kΩ
Output impedance (S0 to S23) at $V_{LCD} = V_{SS}$ (note 2)	R_S	–	–	7,0	kΩ
Driver delays with test loads at $V_{LCD} = V_{SS}$, I_{OS} = 15 μA, I_{OB} = 25 μA	t_{PLCD}	–	–	30	μs
LCD scan frame frequency	f_{LCD}	–	60	–	Hz

Notes to the LCD driver characteristics

1. $V_{LCD} < V_{DD}$−3 V for 1/3 bias.
2. Outputs measured one at a time.

DC CHARACTERISTICS

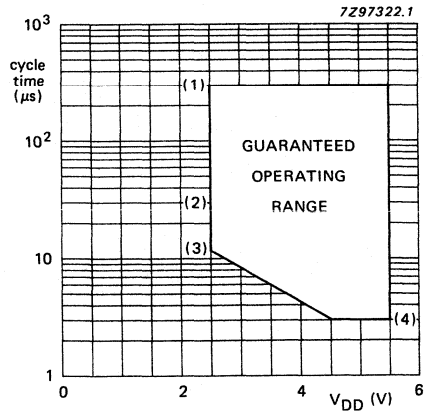
V_{DD} = 2,5 to 5,5 V; V_{SS} = 0 V; T_{amb} = −40 to + 85 °C; all voltages with respect to V_{SS}; unless otherwise specified.

parameter	symbol	min.	typ.	max.	unit
Supply voltage					
operating (see Fig. 37)	V_{DD}	2,5	–	5,5	V
Supply current (note 1)					
operating mode (see Figs 41 and 42)					
at V_{DD} = 5 V; f_{XTAL} = 10 MHz	I_{DD}	–	2,2	4,5	mA
at V_{DD} = 5 V; f_{XTAL} = 6 MHz	I_{DD}	–	1,3	2,6	mA
at V_{DD} = 3 V; f_{XTAL} = 3,58 MHz	I_{DD}	–	0,4	0,8	mA
IDLE mode (see Figs 39 and 40)					
at V_{DD} = 5 V; f_{XTAL} = 10 MHz	I_{DD}	–	0,8	1,6	mA
at V_{DD} = 5 V; f_{XTAL} = 6 MHz	I_{DD}	–	0,5	1	mA
at V_{DD} = 3 V; f_{XTAL} = 3,58 MHz	I_{DD}	–	0,15	0,4	mA
STOP mode (see Fig.38 and note 2)					
at V_{DD} = 2,5 V	I_{DD}	–	26	60	μA
at V_{DD} = 5 V	I_{DD}	–	30	80	μA
Inputs					
Input voltage LOW	V_{IL}	0	–	$0{,}3V_{DD}$	V
Input voltage HIGH	V_{IH}	$0{,}7V_{DD}$	–	V_{DD}	V
Input leakage current					
at $V_{SS} < V_I < V_{DD}$	$\pm I_{IL}$	–	–	1	μA
Outputs					
Output sink current LOW					
at V_{DD} = 5 V ± 10%; V_O = 0,4 V					
except P2.3/SDA, SCLK (see Fig.43)	I_{OL}	1,6	3	–	mA
P2.3/SDA, SCLK (see Fig.44)	I_{OL}	3	–	–	mA
Pull-up output source current HIGH (see Fig.45)					
at V_{DD} = 5 V ± 10%; V_O = $0{,}7V_{DD}$	$-I_{OH}$	40	–	–	μA
at V_{DD} = 5 V ± 10%; V_O = V_{SS}	$-I_{OH}$	–	–	400	μA
Push-pull output source current HIGH					
at V_{DD} = 5 V ± 10%; V_O = V_{DD}−0,4 V	$-I_{OH}$	1,6	3	–	mA

Notes to the DC characteristics

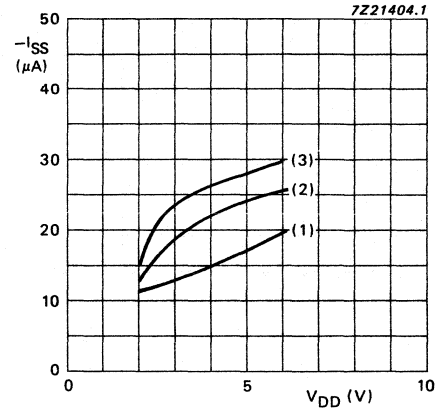
1. V_{IL} = V_{SS}, V_{IH} = V_{DD}; all outputs unloaded; all open drain outputs connected to V_{SS}.
2. Crystal connected between XTAL 1 and XTAL 2; SCLK and SDA pulled to V_{DD} via a 5,6 kΩ resistor; T1 at V_{SS}, $\overline{INT}$ at V_{DD}.

AC CHARACTERISTICS



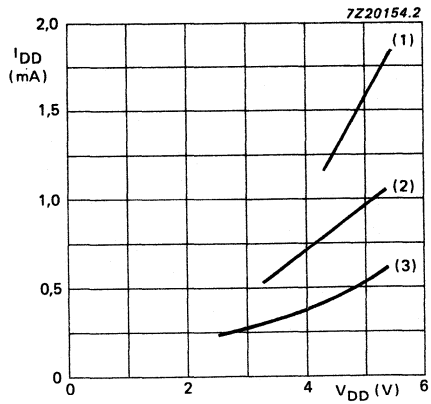
(1) clock frequency = 100 kHz
(2) clock frequency = 1 MHz
(3) clock frequency = 3 MHz
(4) clock frequency = 10 MHz

Fig. 37 Maximum clock frequency (f_{XTAL}) as a function of the supply voltage (V_{DD}).



$f_{CLK} \triangleq 30$ kHz (1) $T_{amb} = +85$ °C
(2) $T_{amb} = +25$ °C
(3) $T_{amb} = -40$ °C

Fig. 38 Typical supply current ($-I_{SS}$) in STOP mode as a function of the supply voltage (V_{DD}).



(1) clock frequency = 10 MHz
(2) clock frequency = 6 MHz
(3) clock frequency = 3,58 MHz

Fig. 39 Maximum supply current (I_{DD}) in IDLE mode as a function of the supply voltage (V_{DD}).

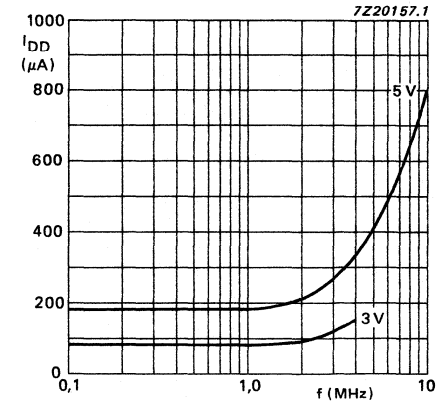
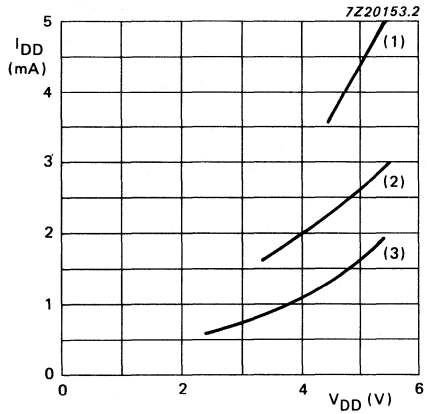


Fig. 40 Typical supply current during IDLE mode as a function of frequency at $V_{DD} = 3$ V and $V_{DD} = 5$ V.

DEVELOPMENT DATA

AC CHARACTERISTICS (continued)



(1) clock frequency = 10 MHz
(2) clock frequency = 6 MHz
(3) clock frequency = 3,58 MHz

Fig. 41 Maximum supply current (I_{DD}) in operating mode as a function of the supply voltage.

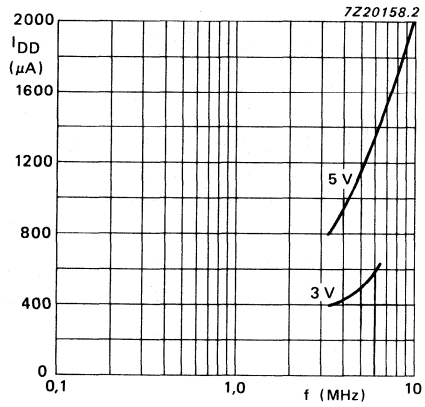


Fig. 42 Typical supply current during operating mode as a function of frequency at V_{DD} = 3 V and V_{DD} = 5 V.

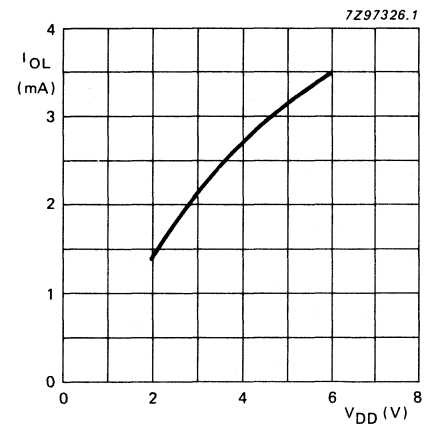


Fig. 43 Typical output sink current (I_{OL}), outputs P0.0 to P0.7, P1.0 to P1.7, DP3.0 to DP3.7, as a function of the supply voltage (V_{DD}); V_O = 0,4 V.

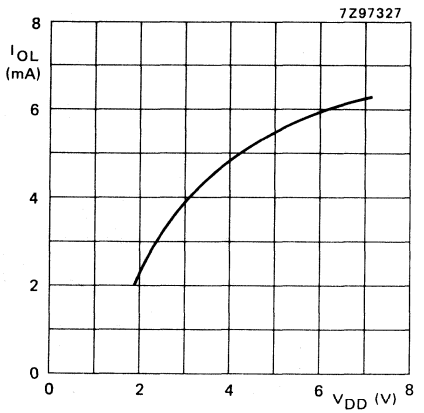
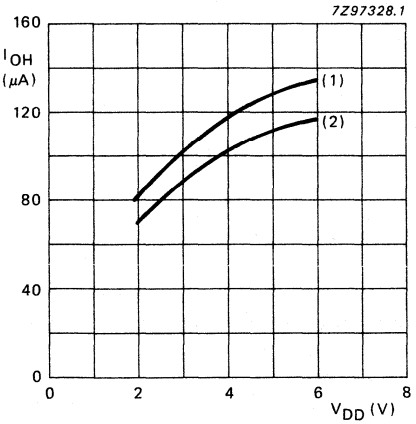
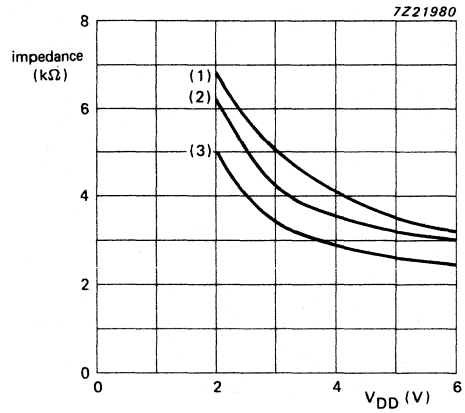


Fig. 44 Typical output sink current (I_{OL}), outputs P2.3/SDA and SCLK, as a function of the supply voltage (V_{DD}); V_O = 0,4 V.

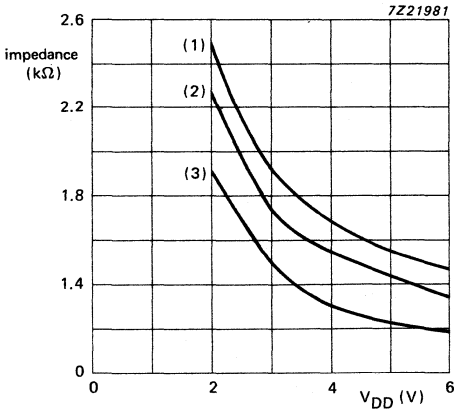


(1) $V_O = V_{SS}$
(2) $V_O = 0{,}7\ V_{DD}$
Fig. 45 Typical output source current ($-I_{OH}$) as a function of the supply voltage (V_{DD}).



(1) + 90 °C
(2) + 25 °C
(3) −50 °C
Fig. 46 Segment output impedance.

DEVELOPMENT DATA



(1) + 90 °C
(2) + 25 °C
(3) −50 °C
Fig. 47 Backplane output impedance.

Table 12 Input timing shown in Fig. 48.

symbol	timing
t_{BUF}	$\geqslant 14t_{XTAL}$
$t_{HD;STA}$	$\geqslant 14t_{XTAL}$
t_{HIGH}	$\geqslant 17t_{XTAL}$
t_{LOW}	$\geqslant 17t_{XTAL}$
$t_{SU;STO}$	$\geqslant 14t_{XTAL}$
$t_{HD;DAT}$	> 0
$t_{SU;DAT}$	$\geqslant 250$ ns
t_{RD}	$\leqslant 1\ \mu s$
t_{RC}	$\leqslant 1\ \mu s$
t_{FD}	$\leqslant 1\ \mu s$
t_{FC}	$\leqslant 0{,}3\ \mu s$

Notes to Table 12

t_{XTAL} = one period of the XTAL input frequency (f_{XTAL})
= 167 ns for f_{XTAL} = 6 MHz.
These figures apply to all modes.

AC CHARACTERISTICS (continued)

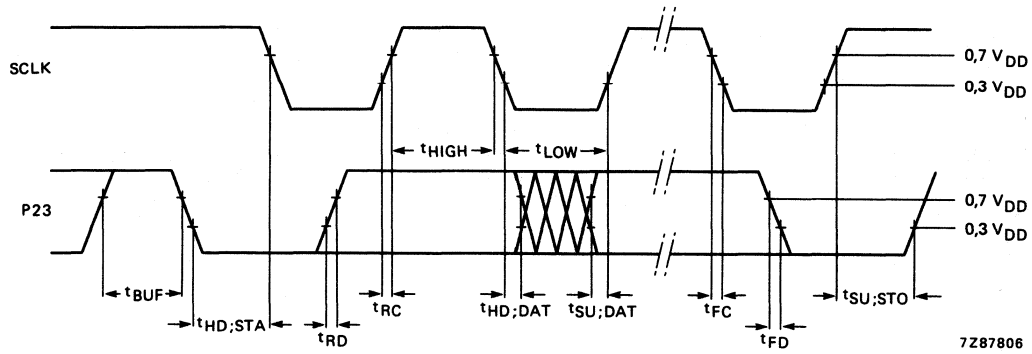


Fig. 48 PCF84C430 timing requirements for the P2.3 and SCLK *input* signals.

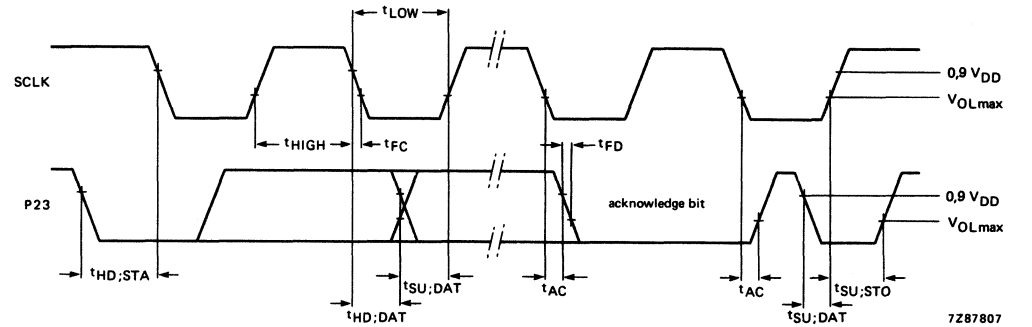


Fig. 49 PCF84C430 timing requirements for the P2.3 and SCLK *output* signals.

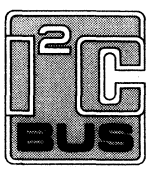
Table 13 Output timing shown in Fig. 49

symbol	timing	
	normal mode (ASC in S2 = 0)	low-speed mode (ASC in S2 = 1
$t_{HD;STA}$	$\frac{1}{2}(DF+9)\,t_{XTAL}$	$\frac{3}{4}(DF+9)\,t_{XTAL}$
t_{HIGH}	$\frac{1}{2}(DF)\,t_{XTAL}$	$\frac{3}{4}(DF)\,t_{XTAL}$
t_{LOW}	$\frac{1}{2}(DF)\,t_{XTAL}$	$\frac{1}{4}(DF)\,t_{XTAL}$
$t_{SU;STO}$	$\frac{1}{2}(DF-3)\,t_{XTAL}$	$\frac{1}{4}(DF-3)\,t_{XTAL}$
$t_{HD;DAT}$ (slave transmitter)		
any DF	$\geqslant 9t_{XTAL}$ $\leqslant 12t_{XTAL}$	$\geqslant 9t_{XTAL}$ $\leqslant 12t_{XTAL}$
$t_{HD;DAT}$ (master transmitter)		
for DF $\leqslant$ 51	$\geqslant 9t_{XTAL}$ $\leqslant 12t_{XTAL}$	– –
for DF $\leqslant$ 99	– –	$\geqslant 9t_{XTAL}$ $\leqslant 12t_{XTAL}$
$t_{SU;DAT}$ (master transmitter)		
for DF > 51	$\geqslant 15t_{XTAL}$ $\leqslant 24t_{XTAL}$	– –
for DF > 99	– –	$\geqslant 15t_{XTAL}$ $\leqslant 24t_{XTAL}$
t_{AC}	$\geqslant 9t_{XTAL}$ $\leqslant 12t_{XTAL}$	$\geqslant 9t_{XTAL}$ $\leqslant 12t_{XTAL}$
$t_{FD};t_{FC}$	$\leqslant$ 100 ns at C_b = 400 pF	$\leqslant$ 100 ns at C_b = 400 pF

DEVELOPMENT DATA

Notes to Table 13

t_{XTAL} = one period of the XTAL input frequency (f_{XTAL})
= 167 ns for f_{XTAL} = 6 MHz.
DF = divisor (see Table 3 Serial I/O section).
C_b = the maximum bus capacitance for each line.



Purchase of Philips' I²C components conveys a license under the Philips' I²C patent to use the components in the I²C-system provided the system conforms to the I²C specifications defined by Philips.

PACKAGE INFORMATION

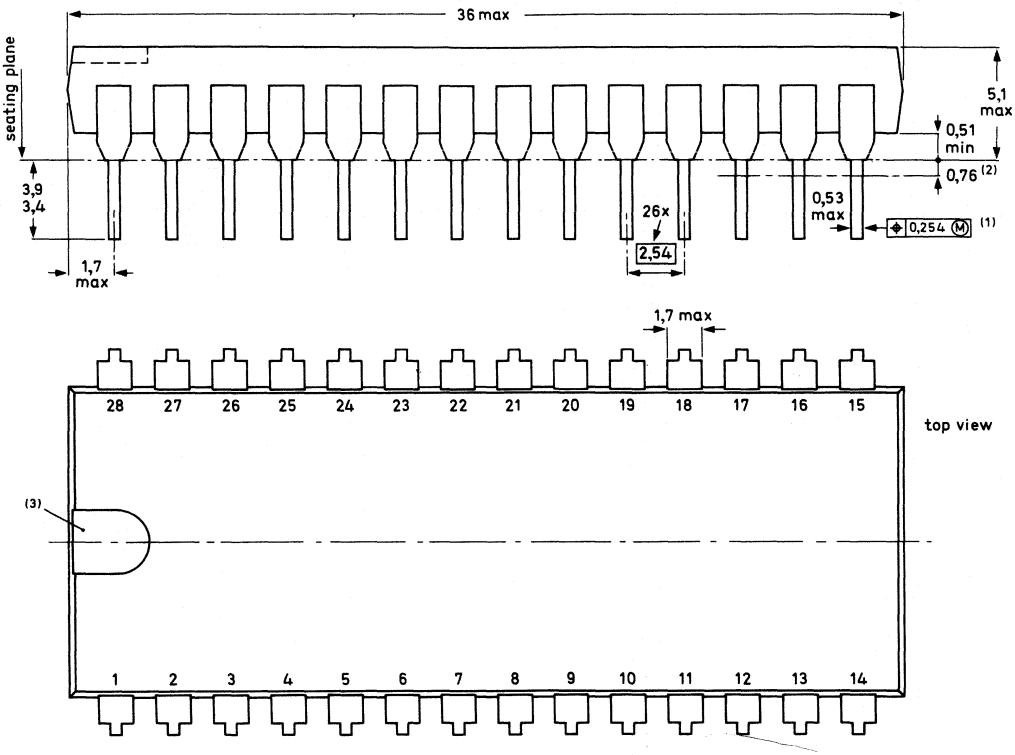
Package outlines **717**
Soldering **733**

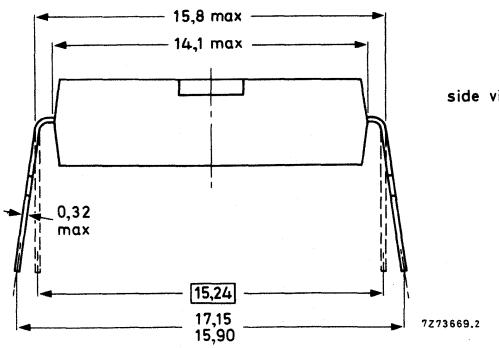
PACKAGE OUTLINES

For product with prefixes: MAB, MAF, PCD, PCA, PCB, PCF

28-lead dual in-line; plastic (SOT117) 719
40-lead dual in-line; plastic (SOT129) 720
28-lead dual in-line; ceramic (cerdip) (SOT135A) 721
28-lead mini-pack; plastic (SO-28; SOT136A) 722
20-lead dual in-line; plastic (SOT146) 723
40-lead mini-pack; plastic (VSO-40; SOT158A) 724
20-lead mini-pack; plastic (SO-20; SOT163A) 725
44-lead (PLCC) 'pocket' version; plastic (SOT187AA) 726
44-lead (PLCC) 'pedestal' version; plastic (SOT187) 727
44-lead quad flat-pack (QFP); plastic (SOT205A) 728
68-lead (PLCC) 'pocket' version; plastic (SOT188AA) 729
68-lead (PLCC) 'pedestal' version; plastic (SOT188) 730
56-lead mini-pack; plastic (VSO-56; SOT190) 731

28-LEAD DUAL IN-LINE; PLASTIC (SOT117)





⊕ Positional accuracy.

Ⓜ Maximum Material Condition.

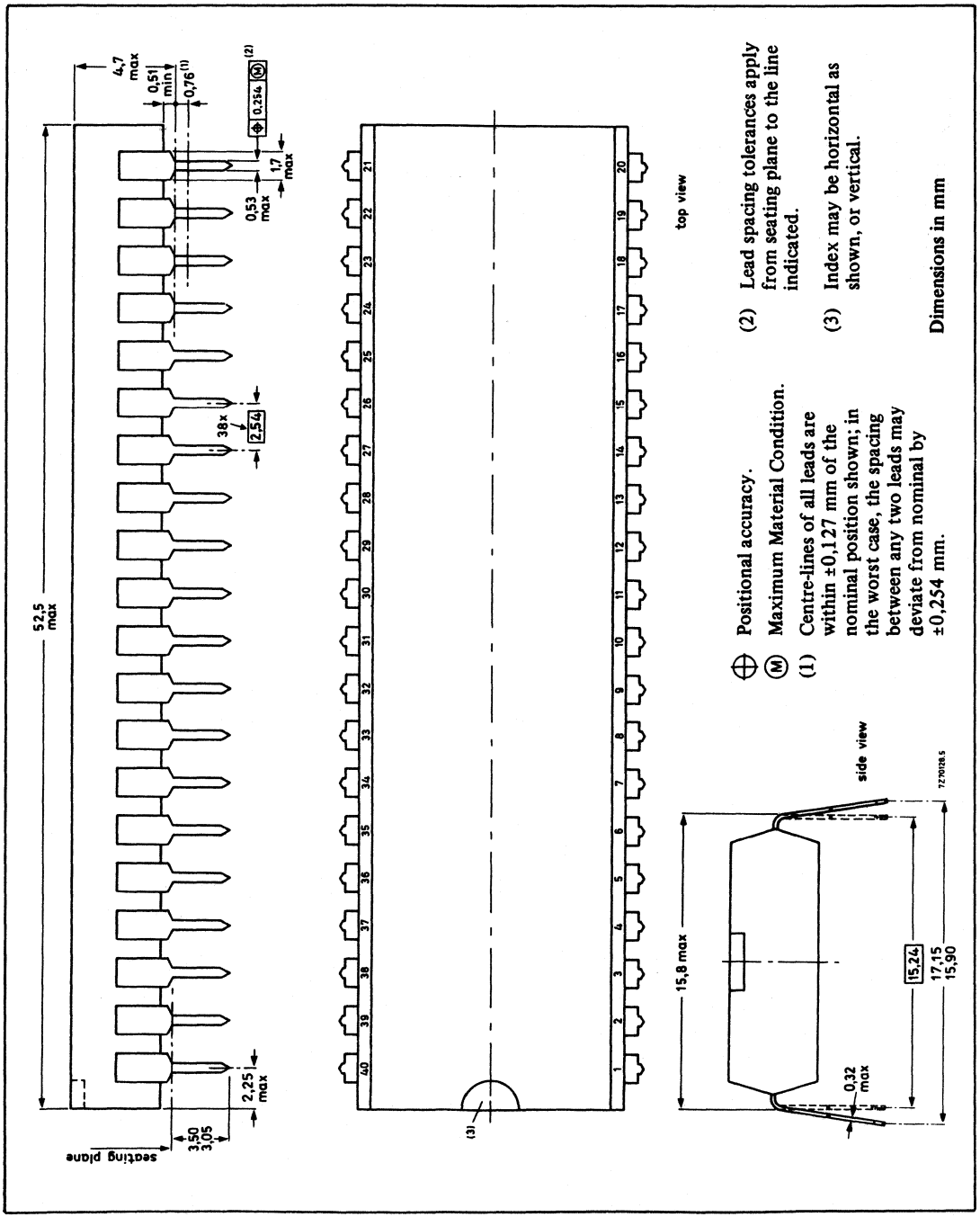
(1) Centre-lines of all leads are within ±0,127 mm of the nominal position shown; in the worst case, the spacing between any two leads may deviate from nominal by ±0,254 mm.

(2) Lead spacing tolerances apply from seating plane to the line indicated.

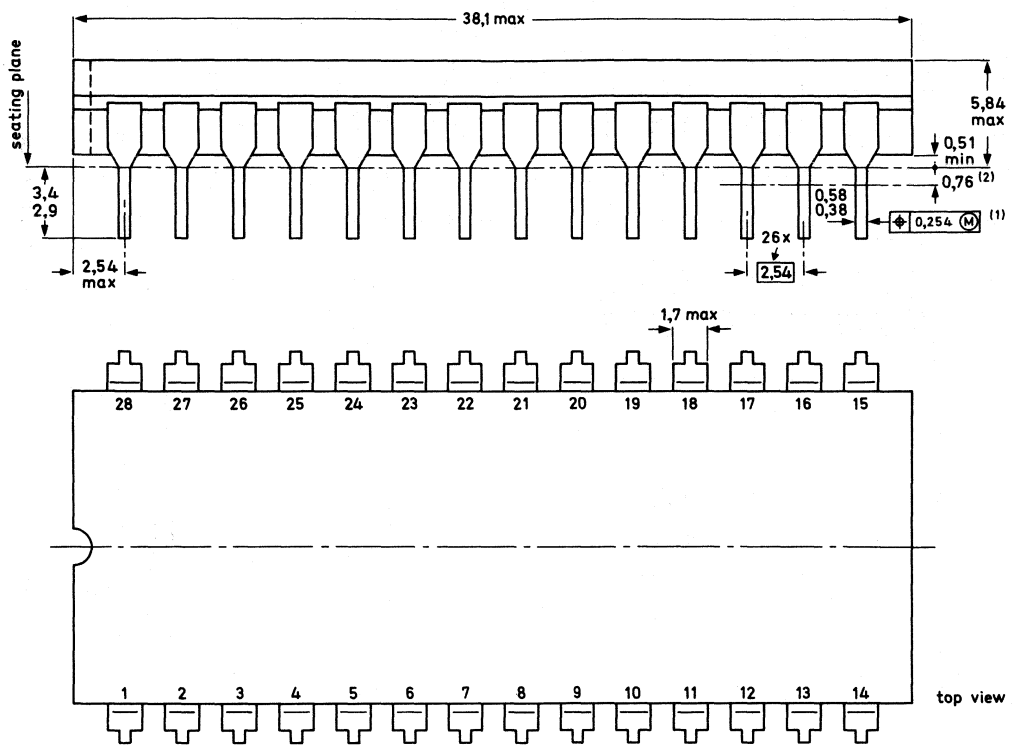
(3) Index may be horizontal as shown, or vertical.

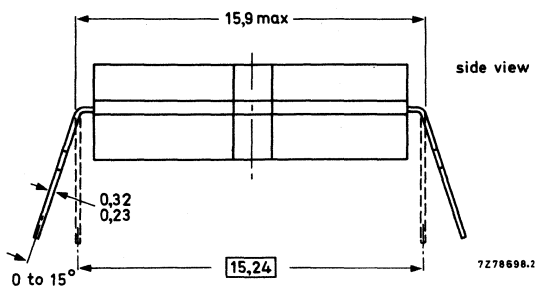
Dimensions in mm

40-LEAD DUAL IN-LINE; PLASTIC (SOT129)



28-LEAD DUAL IN-LINE; CERAMIC (CERDIP) (SOT135A)





⊕ Positional accuracy.

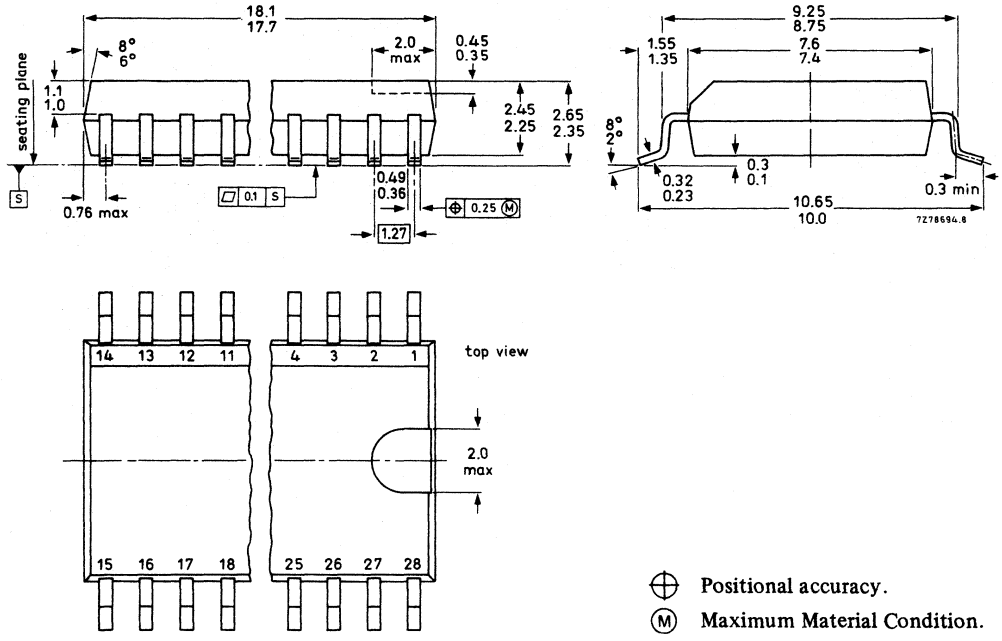
Ⓜ Maximum Material Condition.

(1) Centre-lines of all leads are within ±0,127 mm of the nominal position shown; in the worst case, the spacing between any two leads may deviate from nominal by ±0,254 mm.

(2) Lead spacing tolerances apply from seating plane to the line indicated.

Dimensions in mm

28-LEAD MINI-PACK; PLASTIC (SO28; SOT136A)

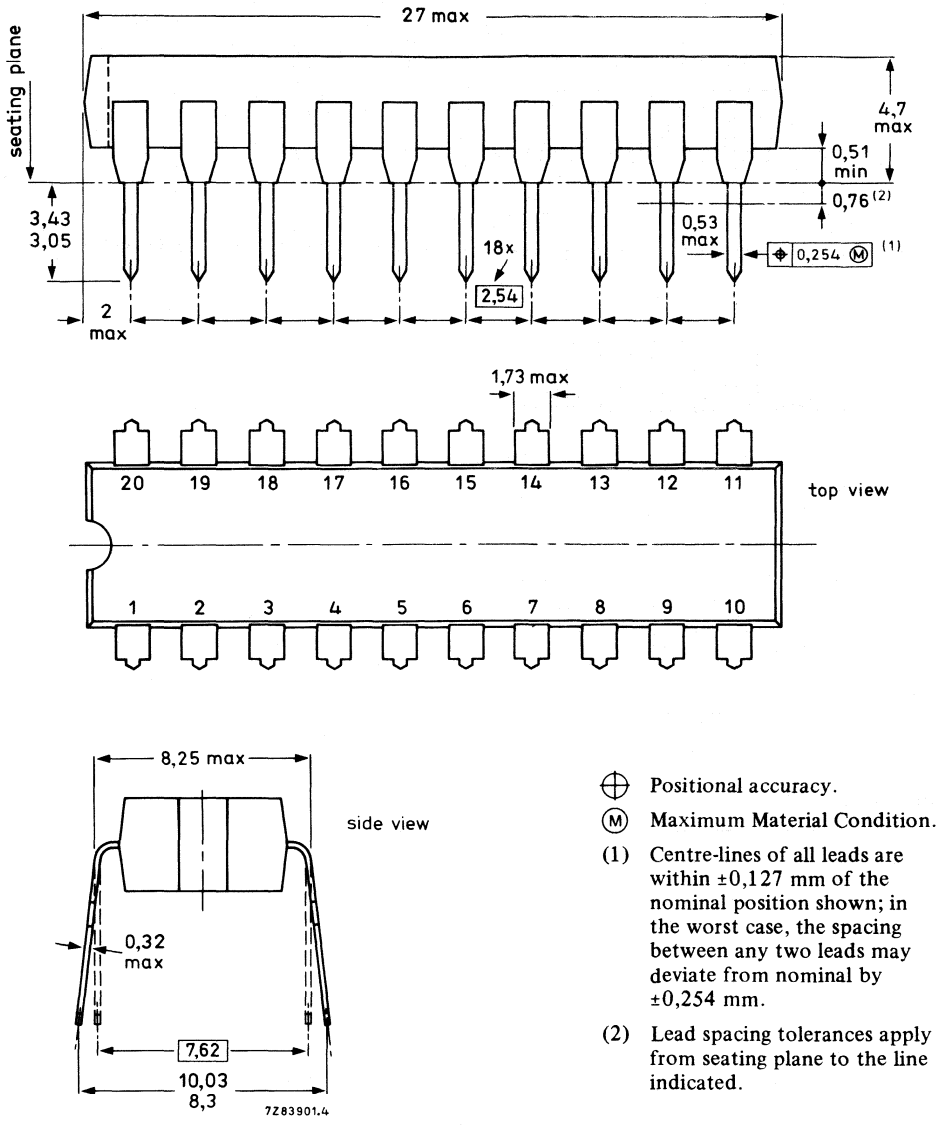


⊕ Positional accuracy.

Ⓜ Maximum Material Condition.

Dimensions in mm

20-LEAD DUAL IN-LINE; PLASTIC (SOT146)



⊕ Positional accuracy.

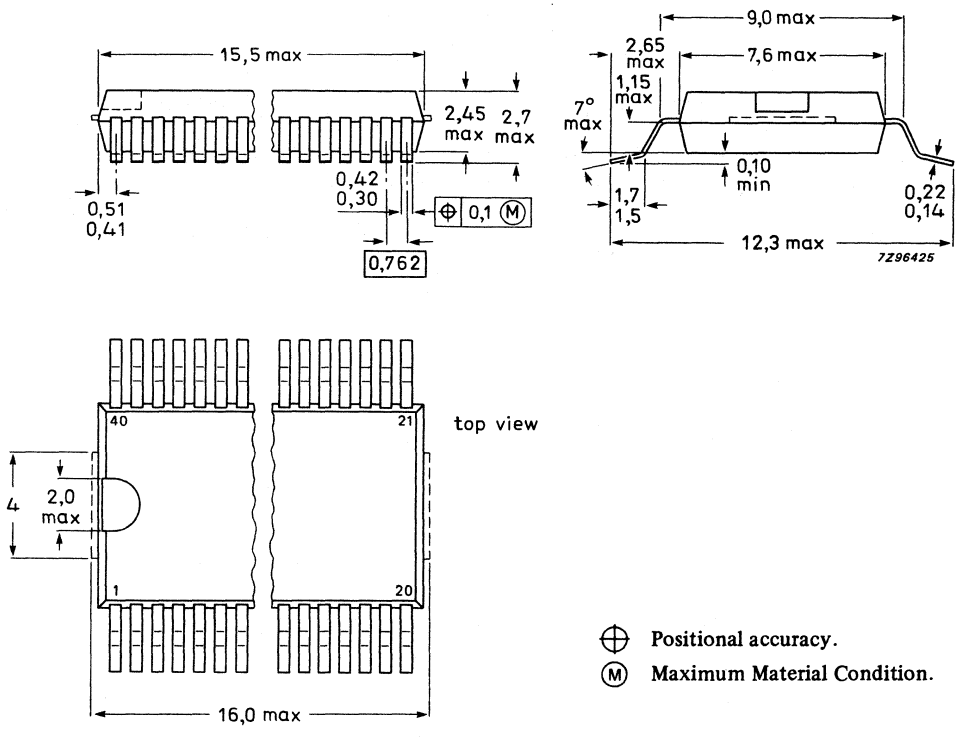
Ⓜ Maximum Material Condition.

(1) Centre-lines of all leads are within ±0,127 mm of the nominal position shown; in the worst case, the spacing between any two leads may deviate from nominal by ±0,254 mm.

(2) Lead spacing tolerances apply from seating plane to the line indicated.

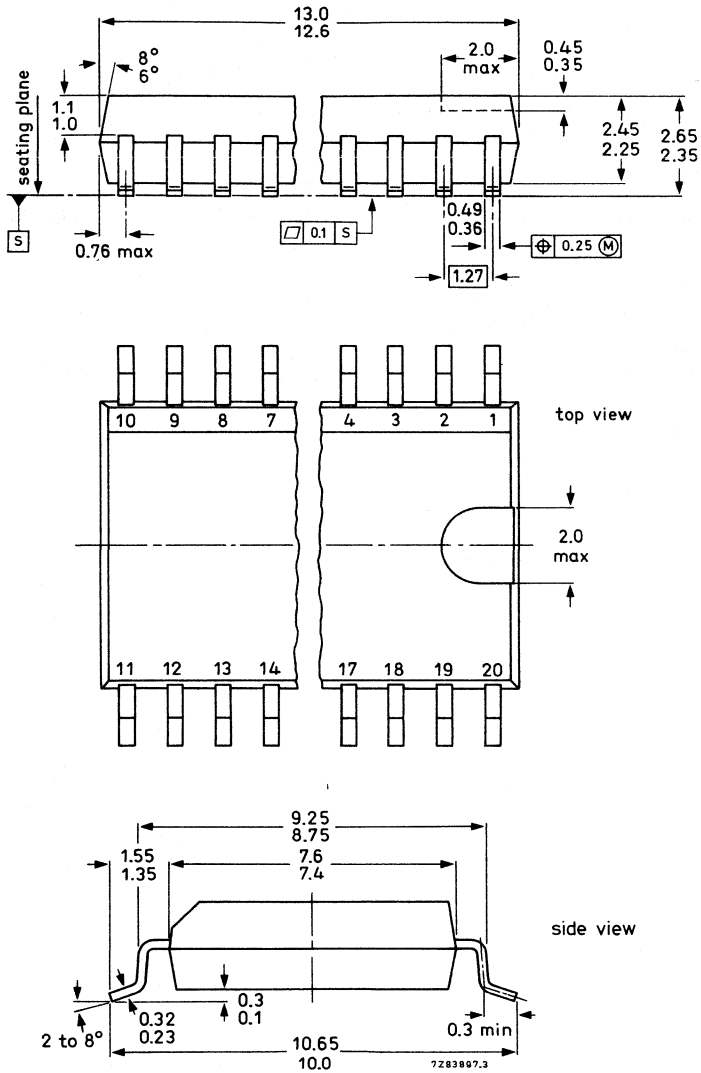
Dimensions in mm

40-LEAD MINI-PACK; PLASTIC (VSO40; SOT158A)



Dimensions in mm

20-LEAD MINI-PACK; PLASTIC (SO20; SOT163A)

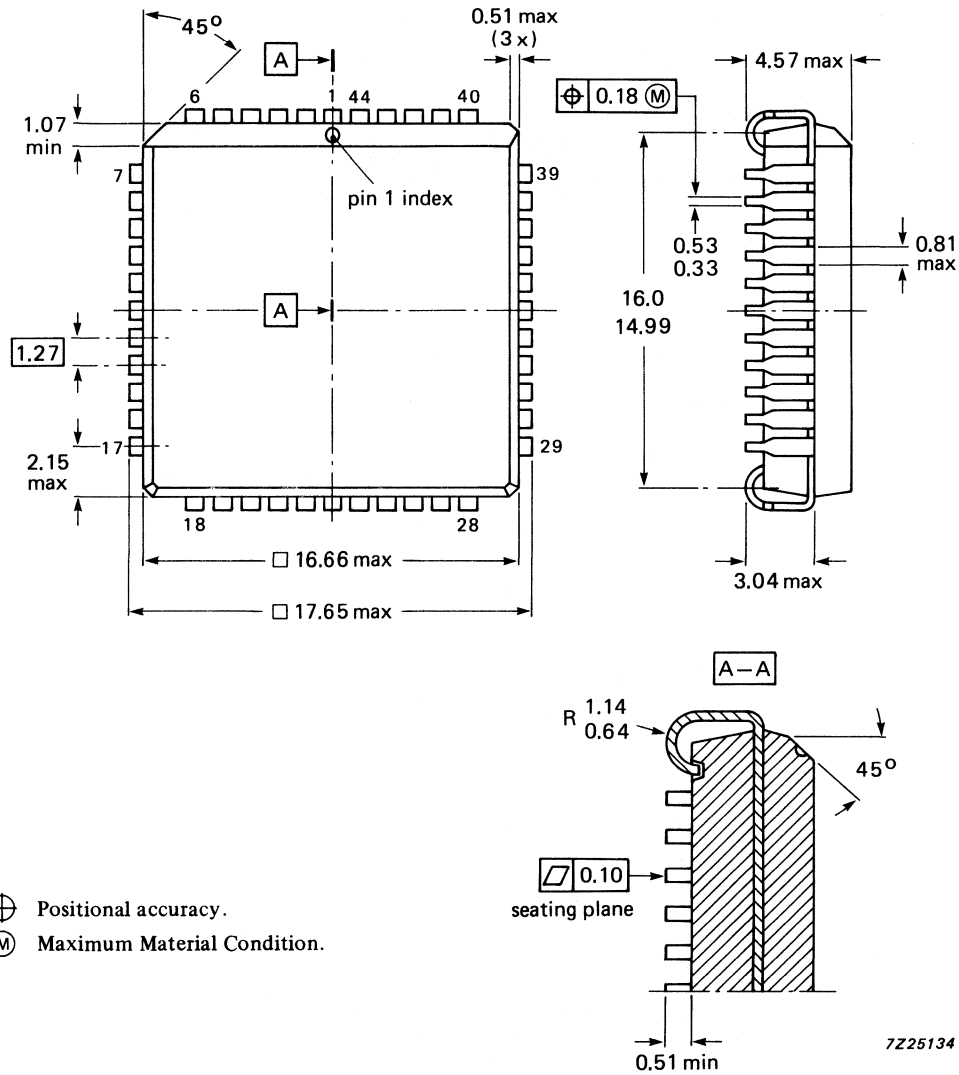


⊕ Positional accuracy.

Ⓜ Maximum Material Condition.

Dimensions in mm

44-LEAD PLASTIC LEADED CHIP CARRIER (PLCC); 'POCKET' VERSION (SOT187AA)

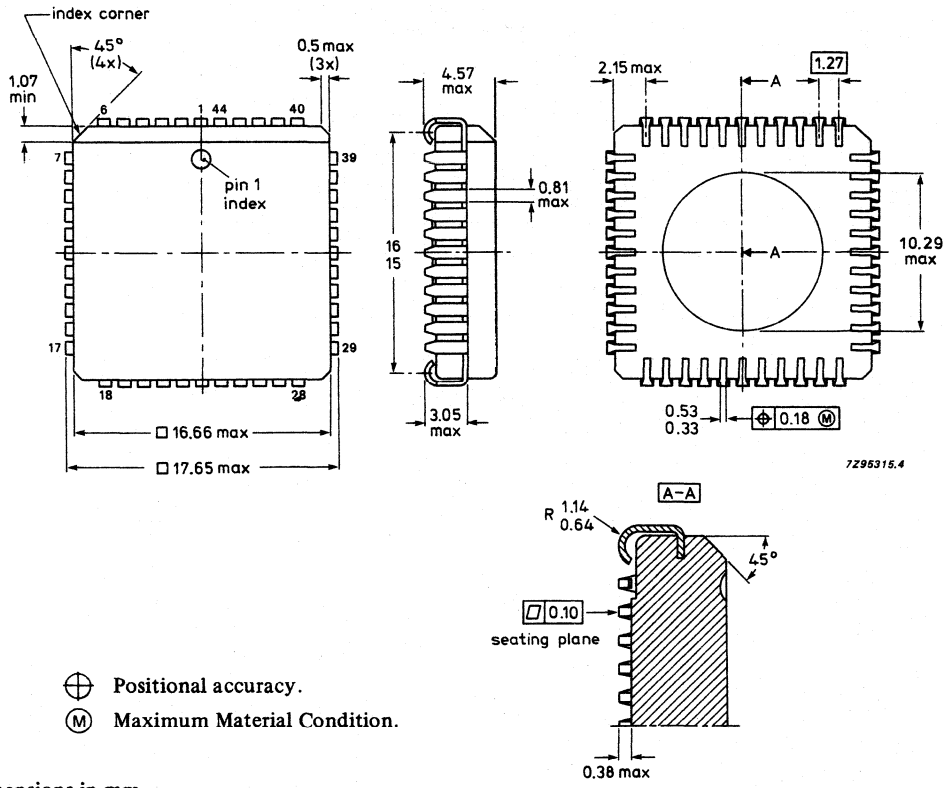


⊕ Positional accuracy.

Ⓜ Maximum Material Condition.

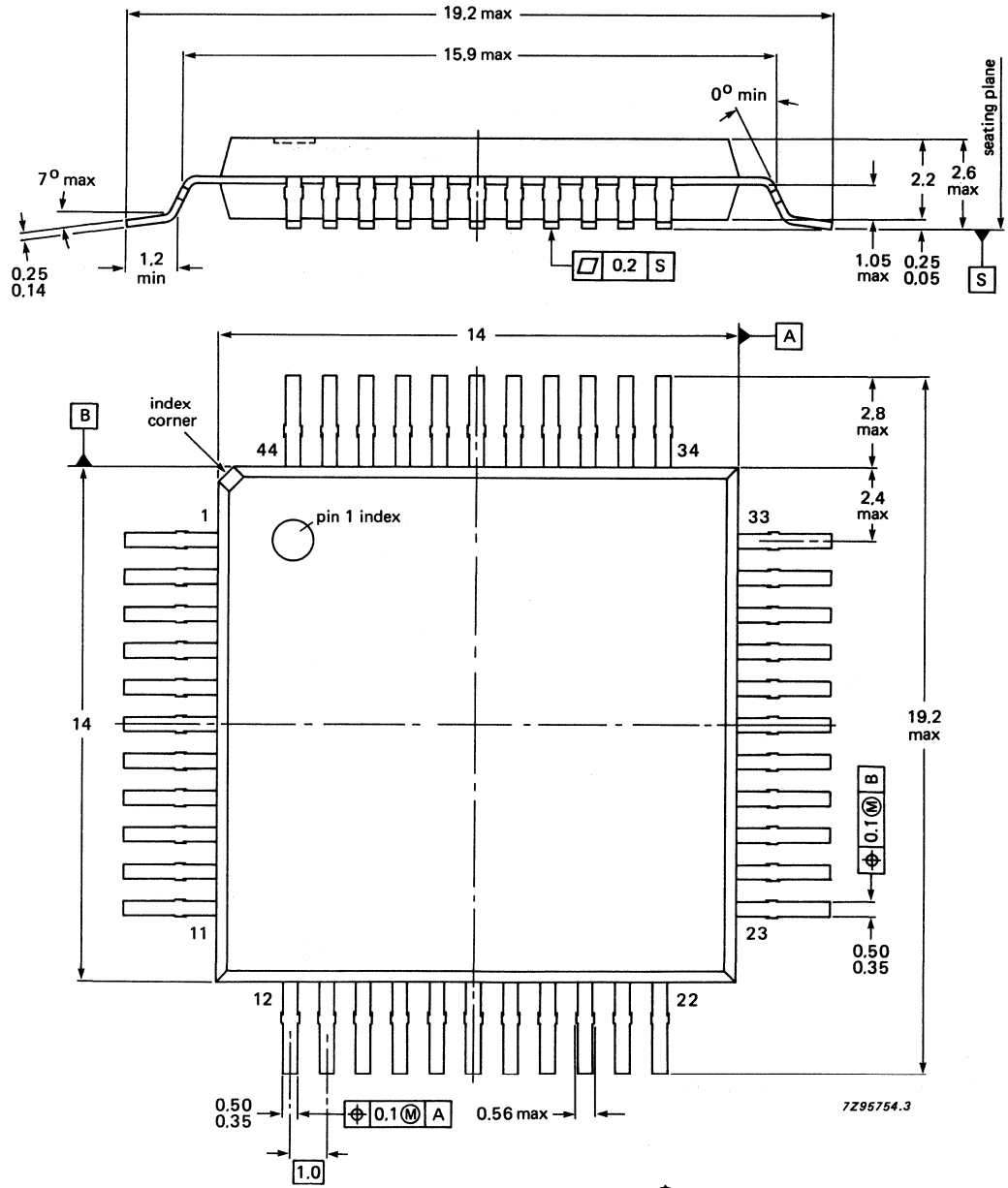
Dimensions in mm

44-LEAD PLASTIC LEADED CHIP CARRIER (PLCC); 'PEDESTAL' VERSION (SOT187)



Dimensions in mm

44-LEAD QUAD FLAT-PACK; PLASTIC (SOT205A)

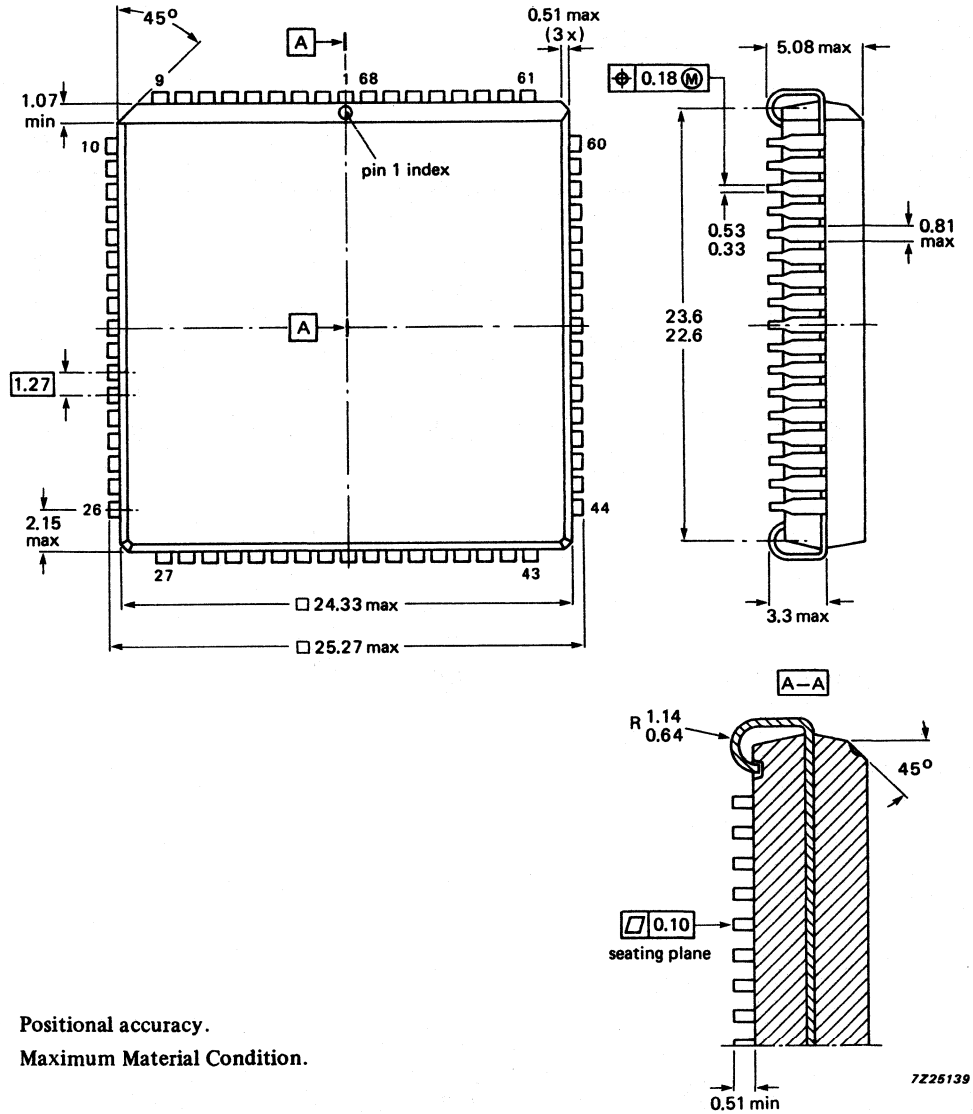


⊕ Positional accuracy.

Ⓜ Maximum Material Condition.

Dimensions in mm

68-LEAD PLASTIC LEADED CHIP CARRIER (PLCC); 'POCKET' VERSION (SOT188AA)

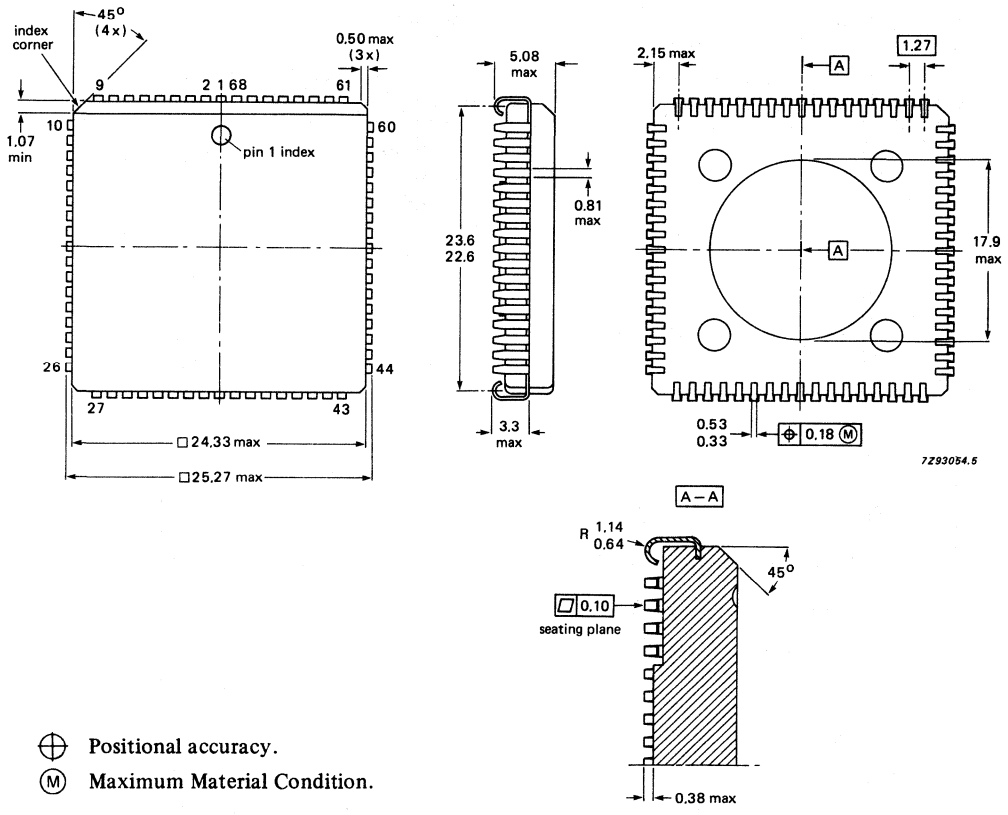


⊕ Positional accuracy.

Ⓜ Maximum Material Condition.

Dimensions in mm

68-LEAD PLASTIC LEADED CHIP CARRIER (PLCC); 'PEDESTAL' VERSION (SOT188)

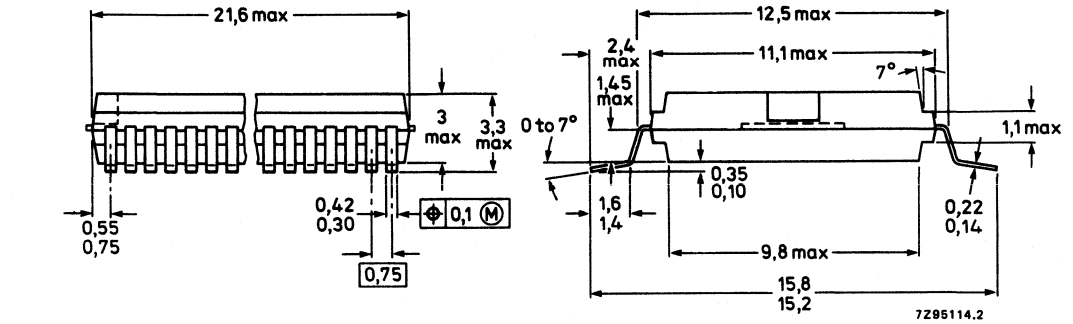


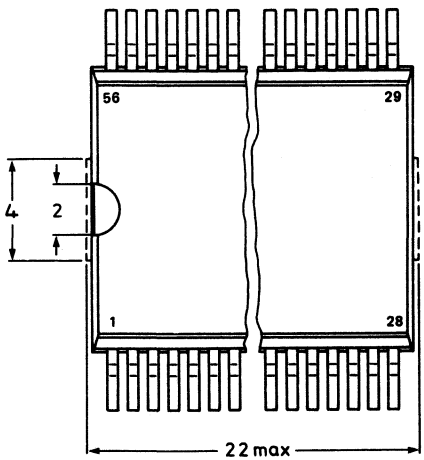
⊕ Positional accuracy.

Ⓜ Maximum Material Condition.

Dimensions in mm

56-LEAD MINI-PACK; PLASTIC (VSO56; SOT190)





top view

⊕ Positional accuracy.

Ⓜ Maximum Material Condition.

Dimensions in mm

Soldering

Plastic dual in-line (DIL) packages **735**
Plastic mini-pack (SO) packages **735**

SOLDERING PLASTIC MINI-PACKS

1. By hand-held soldering iron or pulse-heated solder tool

Fix the component by first soldering two, diagonally opposite end leads. Apply the heating tool to the flat part of the lead only. Contact time must be limited to 10 seconds at up to 300 °C. When using proper tools, all other leads can be soldered in one operation within 2 to 5 seconds at between 270 and 320 °C. (Pulse-heated soldering is not recommended for SO packages).

For pulse-heated solder tool (resistance) soldering of VSO packages, solder is applied to substrate by dipping or by an extra thick tin/lead plating before package placement.

2. By wave

During placement and before soldering, the component must be fixed with a droplet of adhesive. After curing the adhesive, the component can be soldered. The adhesive can be applied by screen printing, pin transfer or syringe dispensing.

Maximum permissible solder temperature is 260 °C, and maximum duration of package immersion in solder bath is 10 seconds, if allowed to cool to less than 150 °C within 6 seconds. Typical dwell time is 4 seconds at 250 °C.

3. By solder paste reflow

Reflow soldering requires the solder paste (a suspension of fine solder particles, flux and binding agent) to be applied to the substrate by screen printing, stencilling or pressure-syringe dispensing before device placement.

Several techniques exist for reflowing, for example, thermal conduction by heated belt, infrared, and vapour-phase reflow. Dwell times vary between 50 and 300 seconds according to method. Typical reflow temperatures range from 215 to 250 °C.

Pre-heating is necessary to dry paste and evaporate binding agent.

Pre-heating duration: 45 minutes at 45 °C.

4. Repairing soldered joints

The same precaution and limits apply as in (1) above.

SOLDERING PLASTIC DUAL IN-LINE PACKAGES

1. By hand

Apply the soldering iron below the seating plane (or not more than 2 mm above it). If its temperature is below 300 °C it must not be in contact for more than 10 seconds; if between 300 and 400 °C, for not more than 5 seconds.

2. By dip or wave

The maximum permissible temperature of the solder is 260 °C; this temperature must not be in contact with the joint for more than 5 seconds. The total contact time of successive solder waves must not exceed 5 seconds.

The device may be mounted up to the seating plane, but the temperature of the plastic body must not exceed the specified storage maximum. If the printed-circuit board has been pre-heated, forced cooling may be necessary immediately after soldering to keep the temperature within the permissible limit.

3. Repairing soldered joints

The same precautions and limits apply as in (1) above.

NOTES

DATA HANDBOOK SYSTEM

DATA HANDBOOK SYSTEM

Our Data Handbook System comprises more than 60 books with specifications on electronic components, subassemblies and materials. It is made up of six series of handbooks:

INTEGRATED CIRCUITS

DISCRETE SEMICONDUCTORS

DISPLAY COMPONENTS

PASSIVE COMPONENTS*

PROFESSIONAL COMPONENTS**

MATERIALS*

The contents of each series are listed on pages iii to viii.

The data handbooks contain all pertinent data available at the time of publication, and each is revised and reissued periodically.

When ratings or specifications differ from those published in the preceding edition they are indicated with arrows in the page margin. Where application is given it is advisory and does not form part of the product specification.

Condensed data on the preferred products of Philips Components is given in our Preferred Type Range catalogue (issued annually).

Information on current Data Handbooks and how to obtain a subscription for future issues is available from any of the Organizations listed on the back cover.
Product specialists are at your service and enquiries will be answered promptly.

* Will replace the Components and materials (green) series of handbooks.
** Will replace the Electron tubes (blue) series of handbooks.

INTEGRATED CIRCUITS

This series of handbooks comprises:

code	handbook title
IC01	**Radio, audio and associated systems** Bipolar, MOS
IC02a/b	**Video and associated systems** Bipolar, MOS
IC03	**ICs for Telecom** Bipolar, MOS Subscriber sets, Cordless Telephones
IC04	**HE4000B logic family** CMOS
IC05	**not yet issued**
IC06	**High-speed CMOS; PC74HC/HCT/HCU** Logic family
IC07	**not yet issued**
IC08	**ECL 10K and 100K logic families**
IC09N	**TTL logic series**
IC10	**Memories** MOS, TTL, ECL
IC11	**Linear Products**
Supplement to IC11	**Linear Products**
IC12	**I^2C-bus compatible ICs**
IC13	**Semi-custom** Programmable Logic Devices (PLD)
IC14	**Microcontrollers** Bipolar, MOS
IC15	**FAST TTL logic series**
IC16	**CMOS integrated circuits for clocks and watches**
IC17	**ICs for Telecom** Bipolar, MOS Radio pagers Mobile telephones ISDN
IC18	**Microprocessors and peripherals**
IC19	**Data communication products**

DISCRETE SEMICONDUCTORS

This series of data handbooks comprises:

current code	new code	handbook title
S1	**SC01***	**Diodes** High-voltage tripler units
S2a	**SC02***	**Power diodes**
S2b	**SC03***	**Thyristors and triacs**
S3	**SC04***	**Small-signal transistors**
S4a	**SC05***	**Low-frequency power transistors and hybrid IC power modules**
S4b	**SC06**	**High-voltage and switching power transistors**
S5	**SC07***	**Small-signal field-effect transistors**
S6	**SC08***	**RF power transistors**
	SC09*	**RF power modules**
S7	**SC10***	**Surface mounted semiconductors**
S8a	**SC11***	**Light emitting diodes**
S8b	**SC12***	**Optocouplers**
S9	**SC13***	**PowerMOS transistors**
S10	**SC14***	**Wideband transistors and wideband hybrid IC modules**
S11	**SC15**	**Microwave transistors**
S15**	**SC16**	**Laser diodes**
S13	**SC17**	**Semiconductor sensors**
S14	**SC18***	**Liquid crystal displays and driver ICs for LCDs**

* Not yet issued with the new code in this series of handbooks.
** New handbook in this series; will be issued shortly.

DISPLAY COMPONENTS

This series of data handbooks comprises:

current code	new code	handbook title
T8	**DC01**	**Colour display systems**
T16	**DC02***	**Monochrome tubes and deflection units**
C2	**DC03***	**Television tuners, coaxial aerial input assemblies**
C3	**DC04***	**Loudspeakers**
C20	**DC05***	**Wire-wound components for TVs and monitors**

* These handbooks are currently issued in another series; they are not yet issued in the Display Components series of handbooks.

PASSIVE COMPONENTS

This series of data handbooks comprises:

current code	new code	handbook title
C14	**PA01**	**Electrolytic capacitors; solid and non-solid**
C11	**PA02***	**Varistors, thermistors and sensors**
C12	**PA03***	**Potentiometers, encoders and switches**
C7	**PA04***	**Variable capacitors**
C22	**PA05***	**Film capacitors**
C15	**PA06***	**Ceramic capacitors**
C9	**PA07***	**Piezoelectric quartz devices**
C13	**PA08***	**Fixed resistors**

* Not yet issued with the new code in this series of handbooks.

PROFESSIONAL COMPONENTS

This series of data handbooks comprises:

current code	new code	handbook title
T1	*	**Power tubes for RF heating and communications**
T2a	*	**Transmitting tubes for communications, glass types**
T2b	*	**Transmitting tubes for communications, ceramic types**
T3	**PC01****	**High-power klystrons**
T4	*	**Magnetrons for microwave heating**
T5	**PC02****	**Cathode-ray tubes**
T6	**PC03****	**Geiger-Müller tubes**
T9	**PC04****	**Photo and electron multipliers**
T10	**PC05****	**Plumbicon camera tubes and accessories**
T11	**PC06****	**Microwave diodes and sub-assemblies**
T12	**PC07****	**Vidicon and Newvicon camera tubes and deflection units**
T13	**PC08****	**Image intensifiers and infrared detectors**
T15	**PC09****	**Dry reed switches**
C8	**PC10**	**Variable mains transformers; annular fixed transformers**

* These handbooks will not be reissued.
** Not yet issued with the new code in this series of handbooks.

MATERIALS

This series of data handbooks comprises:

current code	new code	handbook title
C4, C5	MA01*	**Soft Ferrites**
C16	MA02**	**Permanent magnet materials**
C19	MA03**	**Piezoelectric ceramics**

* Handbooks C4 and C5 will be reissued as one handbook having the new code MA01.

** Not yet issued with the new code in this series of handbooks.

Philips Components – a worldwide Group of Companies

Argentina: PHILIPS ARGENTINA S.A., Div. Components, Vedia 3892, 1430 BUENOS AIRES, Tel. (01) 541 - 7141 to 7747.
Australia: PHILIPS COMPONENTS PTY Ltd, 11 Waltham Street, ARTARMON, N.S.W. 2064, Tel. (02) 439 3322.
Austria: ÖSTERREICHISCHE PHILIPS INDUSTRIE G.m.b.H., UB Bauelemente, Triester Str. 64, 1101 WIEN, Tel. (0222) 60 101-820.
Belgium: N.V. PHILIPS PROF. SYSTEMS – Components Div., 80 Rue Des Deux Gares, B-1070 BRUXELLES, Tel. (02) 52 56 111.
Brazil: CONSTANTA-IBRAPE; (Active Devices): Av. Brigadeiro Faria Lima, 1735-SAO PAULO-SP, Tel. (011) 211-2600. CONSTANTA-IBRAPE; (Passive Devices & Materials): Av. Francisco Monteiro, 702 – RIBEIRAO PIRES-SP, Tel. (011) 459-8211.
Canada: PHILIPS ELECTRONICS LTD., Philips Components, 601 Milner Ave., SCARBOROUGH, Ontario, M1B 1M8, Tel. (416) 292-5161.
Chile: PHILIPS CHILENA S.A., Av. Santa Maria 0760, SANTIAGO, Tel. (02) 77 38 16.
Colombia: IND. PHILIPS DE COLOMBIA S.A., c/o IPRELENSO LTD., Cra. 21, No. 56-17, BOGOTA, D.E., Tel. (01) 249 76 24.
Denmark: PHILIPS COMPONENTS A/S, Prags Boulevard 80, PB 1919, DK-2300 COPENHAGEN S, Tel. 01-54 11 33.
Finland: PHILIPS COMPONENTS, Sinikalliontie 3, SF-2630 ESPOO HELSINKI 10, Tel. 358-0-50 261.
France: RTC-COMPELEC, 117 Quai du Président Roosevelt, 92134 ISSY-LES-MOULINEAUX Cedex, Tel. (01) 40 93 80 00.
Germany (Fed. Republic): VALVO, UB Bauelemente der Philips G.m.b.H., Valvo Haus, Burchardstrasse 19, D-2 HAMBURG 1, Tel. (040) 3296-0.
Greece: PHILIPS HELLENIQUE S.A., Components Division, No. 15, 25th March Street, GR 17778 TAVROS, Tel. (01) 4894 339/4894 911.
Hong Kong: PHILIPS HONG KONG LTD., Components Div., 15/F Philips Ind. Bldg., 24-28 Kung Yip St., KWAI CHUNG, Tel. (0)-24 51 21.
India: PEICO ELECTRONICS & ELECTRICALS LTD., Components Dept., Band Box Building, 254-D Dr. Annie Besant Rd., BOMBAY – 400 025, Tel. (022) 493 0311/493 0590.
Indonesia: P.T. PHILIPS-RALIN ELECTRONICS, Components Div., Setiabudi II Building, 6th Fl., Jalan H.R. Rasuna Said (P.O. Box 223/KBY) Kuningan, JAKARTA 12910, Tel. (021) 51 79 95.
Ireland: PHILIPS ELECTRONICS (IRELAND) LTD., Components Division, Newstead, Clonskeagh, DUBLIN 14, Tel. (01) 69 33 55.
Italy: PHILIPS S.p.A., Philips Components, Piazza IV Novembre 3, I-20124 MILANO, Tel. (02) 6752.1.
Japan: NIHON PHILIPS CORP., Shuwa Shinagawa Bldg., 26-33 Takanawa 3-chome, Minato-ku, TOKYO (108), Tel. (03) 448-5611. (IC Products) SIGNETICS JAPAN LTD., 8-7 Sanbancho Chiyoda-ku, TOKYO 102, Tel. (03) 230-1521.
Korea (Republic of): PHILIPS ELECTRONICS (KOREA) LTD., Elcoma Div., Philips House, 260-199 Itaewon-dong, Yongsan-ku, SEOUL, Tel. (02) 794-5011.
Malaysia: PHILIPS MALAYSIA SDN BHD, Components Div., 3 Jalan SS15/2A SUBANG, 47500 PETALING JAYA, Tel. (03) 734 5511.
Mexico: ELECTRONICA, S.A de C.V., Carr. México-Toluca km. 62.5, TOLUCA, Edo. de México 50140, Tel. Toluca 91 (721) 613-00.
Netherlands: PHILIPS NEDERLAND, Marktgroep Philips Components, Postbus 90050, 5600 PB EINDHOVEN, Tel. (040) 78 37 49.
New Zealand: PHILIPS NEW ZEALAND LTD., Components Division, 110 Mt. Eden Road, C.P.O. Box 1041, AUCKLAND, Tel. (09) 605-914.
Norway: NORSK A/S PHILIPS, Philips Components, Box 1, Manglerud 0612, OSLO, Tel. (02) 68 02 00.
Pakistan: PHILIPS ELECTRICAL CO. OF PAKISTAN LTD., Philips Markaz, M.A. Jinnah Rd., KARACHI-3, Tel. (021) 72 57 72.
Peru: CADESA, Av. Pardo y Aliaga No. 695, 6th Floor, San Isidro, LIMA 27, Tel. (014) 70 70 80.
Philippines: PHILIPS INDUSTRIAL DEV. INC., 2246 Pasong Tamo, P.O. Box 911, Makati Comm. Centre, MAKATI-RIZAL 3116, Tel. (02) 86 89 51 to 59.
Portugal: PHILIPS PORTUGUESA S.A.R.L., Av. Eng. Duarte Pacheco 6, 1009 LISBOA Codex, Tel. (019) 68 31 21.
Singapore: PHILIPS PROJECT DEV. (Singapore) PTE LTD., Components Div., Lorong 1, Toa Payoh, SINGAPORE 1231, Tel. 35 02 000.
Spain: PHILIPS Components, Balmes 22, 08007 BARCELONA, Tel. (03) 301 63 12.
Sweden: PHILIPS Components, A.B., Tegeluddsvägen 1, S-11584 STOCKHOLM, Tel. (0)8-782 1 000.
Switzerland: PHILIPS A.G., Components Dept., Allmendstrasse 140-142, CH-8027 ZÜRICH, Tel. (01) 488 22 11.
Taiwan: PHILIPS TAIWAN LTD., 150 Tun Hua North Road, P.O. Box 22978, TAIPEI, Taiwan, Tel. (02) 71 20 500.
Thailand: PHILIPS ELECTRICAL CO. OF THAILAND LTD., 283 Silom Road, P.O. Box 961, BANGKOK, Tel. (02) 233-6330-9.
Turkey: TÜRK PHILIPS TICARET A.S., Philips Components, Talatpasa Cad. No. 5, 80640 LEVENT/ISTANBUL, Tel. (01) 179 27 70.
United Kingdom: PHILIPS COMPONENTS LTD., Mullard House, Torrington Place, LONDON WC1E 7HD, Tel. (01) 580 6633.
United States: AMPEREX ELECTRONIC CORP., 230 Duffy Avenue, P.O. Box 560, HICKSVILLE, L.I.N.Y. 11802, Tel. (516) 931-6200. (Active Devices & Materials) AMPEREX SALES CORP., Providence Pike, SLATERSVILLE, R.I. 02876, Tel. (401) 762-9000. (Colour picture tubes – Monochrome & Colour Display Tubes) PHILIPS DISPLAY COMPONENTS, 50 Johnston St., SENECA FALLS, N.Y. 13148, Tel. (315) 568-5881. (IC Products) SIGNETICS CORPORATION, 811 East Arques Avenue, SUNNYVALE, CA 94088-3409, Tel. (408) 991-2000. (Passive & Electromech. Dev.) MEPCO/CENTRALAB, INC., 2001 West Blue Heron Blvd, RIVIERA BEACH, Florida 33404, Tel. (305) 881-3200.
Uruguay: LUZILECTRON S.A., Avda Uruguay 1287, P.O. Box 907, MONTEVIDEO, Tel. (02) 985 395.
Venezuela: IND. VENEZOLANAS PHILIPS S.A., c/o MAGNETICA S.A., Calle 6, Ed. Las Tres Jotas, App. Post. 78117, CARACAS, Tel. (02) 239 39 31.

For all other countries apply to: Philips Components Division, International Business Relations, P.O. Box 218, 5600 MD EINDHOVEN, The Netherlands, Telex 35000 phtcnl

AS67
© Philips Export B.V. 1989

All rights are reserved. Reproduction in whole or in part is prohibited without the prior written consent of the copyright owner.

The information presented in this document does not form part of any quotation or contract, is believed to be accurate and reliable and may be changed without notice. No liability will be accepted by the publisher for any consequence of its use. Publication thereof does not convey nor imply any license under patent- or other industrial or intellectual property rights.

Printed in The Netherlands 9398 153 30011

Philips Components

PHILIPS

PHILIPS