Programmable Pulse Generator

PM5781

Operators Manual

GPIB-bus operation

12 Programming the PM 5781

13 Command Reference

14 Command Set, Alphabetical

15 Program Examples

16 Error Messages

17 Appendix

18 Index

Safety Instructions

Introduction

Read these pages carefully before you install and use the instrument.

This instrument has been designed and tested in accordance with IEC publication 348 and has been supplied in a safe condition. The user of this instrument should be knowledgeable of its use. This knowledge can be gained by thoroughly studying this manual.

This equipment is designed to be used by trained personnel only. Repair, maintenance and adjustment of exposed components must be carried out by qualified personnel who are aware of the hazards involved.

Safety Precautions

To ensure the correct and safe operation of this instrument, it is essential that operators and service technicians follow generally accepted safety procedures in addition to the safety precautions specified in this manual.

Caution and Warning Statements

CAUTION: indicates where incorrect operating procedures can cause damage to, or destruction of, equipment or other property.

WARNING: indicates a potential danger that requires correct procedures or practices in order to prevent personal injury.

Symbols

Indicates where the protective ground terminal is connected inside the instrument. Never remove or loosen this screw.

Indicates that special attention must be taken to preserve the instrument from damage. The operator must refer to an explanation in the instruction manual.

If in doubt about safety

Whenever you suspect that it is unsafe to use the instrument, you must make it inoperative, clearly mark it to prevent its further operation, and inform your local service department. The instrument is likely to be unsafe if it is visibly damaged.

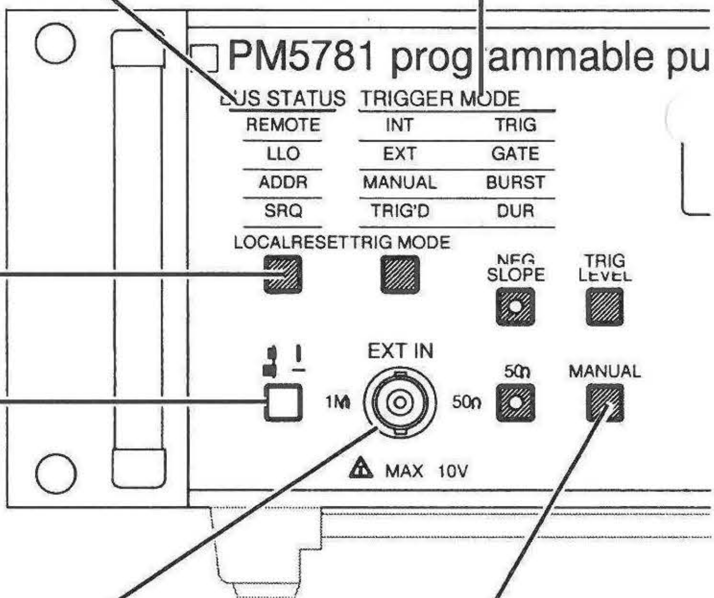
This page is intentionally left blank.

Product Presentation

BUS STATUS This display indicates bus status of the generator

- Remote controlled,
- Local Lock Out,
- Addressed as listener, or talker
- Service Requested.

TRIGGER MODE
The trigger mode display indicates the selected trigger source and trigger function. The Trig Mode key is used together with the keyboard to select these functions.



LOCAL/RESET
The generator is reset or switched to local operation with the LOCAL / RESET key.

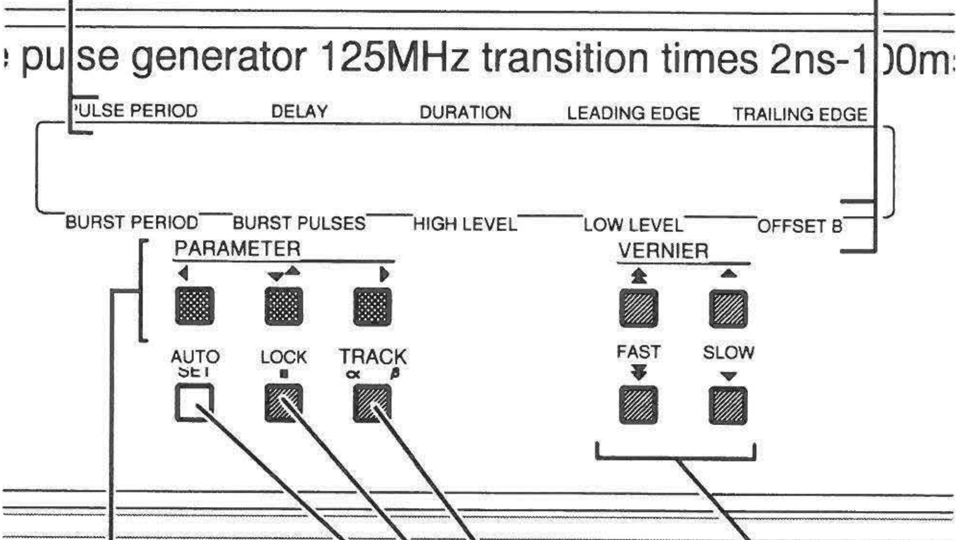
POWER
The power ON/OFF function is a two-pole switch. The position is mechanically indicated on the front panel.

EXT IN
External input for triggering of the generator from an external source. The input impedance and trigger slope are selectable. The Trig Level may be set with the keyboard or with the Vernier keys.

MANUAL
The MANUAL key is used to manually stimulate the pulse generator in a manner equivalent to an external source.

DISPLAY, upper
The upper line of the display is used for setting: PERIOD, DELAY, DURATION, LEADING EDGE and TRAILING EDGE., (see also page 2-8)

DISPLAY, lower
The lower line of the display is used for setting BURST PERIOD, BURST PULSES,. HIGH LEVEL and LOW LEVEL for both channels, and OFFSET for the B channel.



PARAMETER
The three keys in this pad are used to position the cursor on the parameter to be set.

VERNIER
These keys are used to decrease or increase parameter values with fast or slow speed.

AUTO SET
This function automatically corrects erroneous settings. After use of the Auto Set function the new corrected parameter values are indicated in the display.

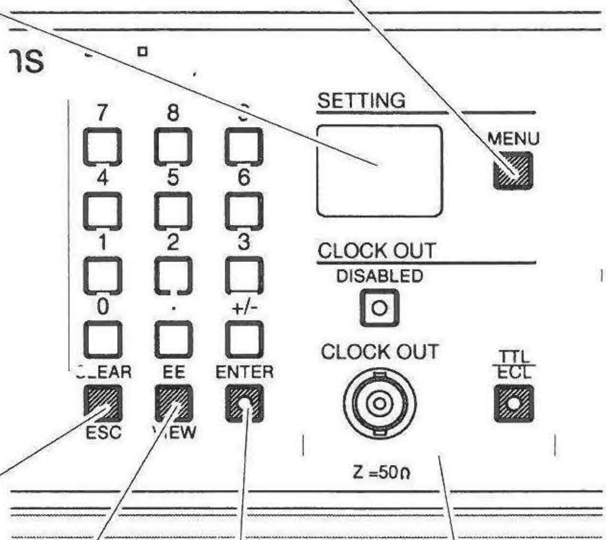
LOCK
Numerical parameter values may be frozen to prevent unintentional changes.

TRACK
Forces selected parameters to follow other parameters proportionally.

SETTING
The display indicates the memory number of the recalled instrument setting.

MENU
The MENU key works with the main display and gives access to functions as: save/recall front panel settings, voltage limits, front panel display modes, format of the exponent, beeper ON / OFF, GPIB -address, debug on/off, test programs, period control ON/OFF, internal calibration (optional)...

KEYBOARD
The keyboard is used: to:
- set numerical parameter values,
- select trigger source and function and to set trigger level,
- select alternatives in menu mode.



CLEAR/ESC
- clears the display entries
- steps back in the menu tree
- gives some second level functions

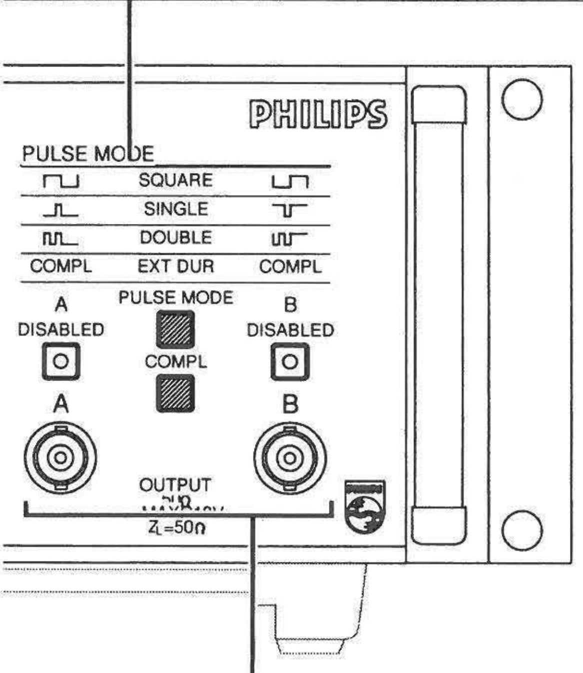
EE/VIEW
- EE stands for exponent entry
- VIEW allows you to study any of the saved front panel settings in some special menu modes.

ENTER key is used to confirm the set values.

CLOCK OUT
The CLOCK output carries in internal trigger modes a symmetrical square wave. The levels may be set for TTL or ECL. The key DISABLE disconnects the signal source.

PULSE MODE

The pulse modes: SQUARE, SINGLE and DOUBLE are selected with the PULSE MODE key. EXTERNAL DURATION is selected in the TRIGGER MODE function menu. The selected pulse mode is indicated in the PULSE MODE display.



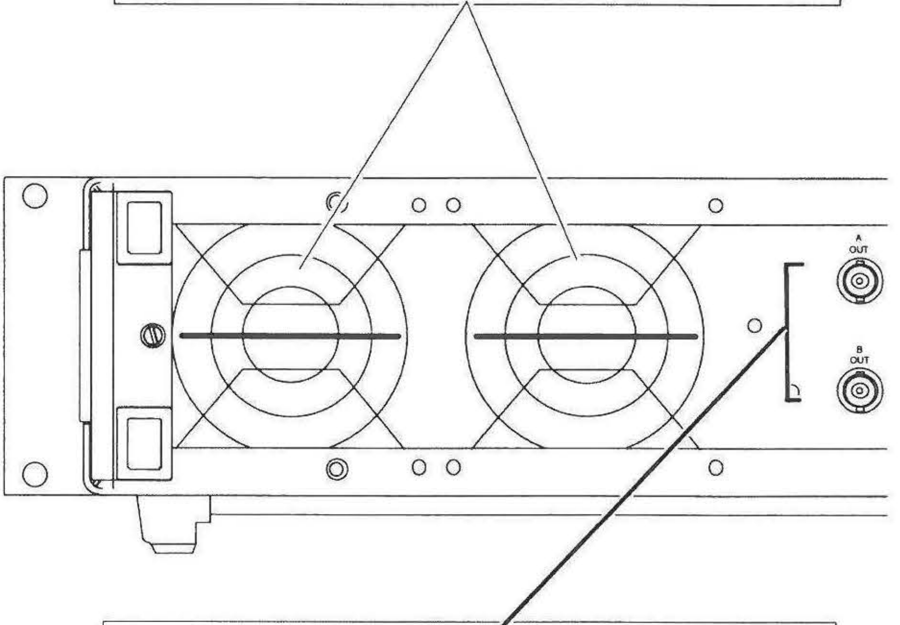
OUTPUTS A, B

The two outputs provides a normal and a complementary output pulse. The ON / OFF ratio may be interchanged with the COMPL key. The signal sources are disconnected when the DISABLED keys are lit.

Rear Panel

FANS

There are two fans on the rear panel. Make sure that there are no obstacles for the air streams. The air inlets are located in the side panels.



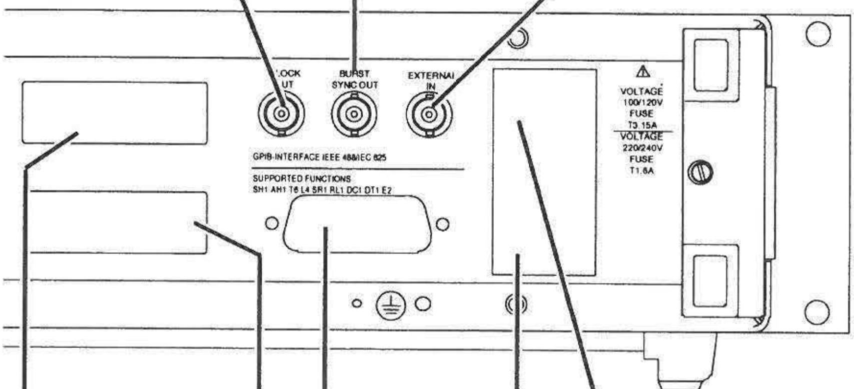
PULSE OUTPUTS

(Optional) The rear panels of PM 5781/012 and PM 5781/022 have connectors for the pulse outputs A and B. The front panel outputs are not connected in these versions.

CLOCK OUT
(Optional) The clock signal is connected to this connector instead of the front panel connector on PM 5781/012 and PM 5781/022

BURST SYNC OUT
This output carries a pulse that starts at the leading edge of the first pulse, and stops at the trailing edge of the last pulse is the burst.

EXTERNAL IN
(Optional) The External Input is connected to this connector instead of the front panel connector on the PM 5781/012 and PM 5781/022 .



TYPE LABEL
The type label indicates the serial number and the version number of the pulse generator.

GPIB CONNECTOR
The GPIB cable is connected here.

LINE VOLTAGE
Line voltage inlet and line voltage selector. Check the voltage setting before connection is made.

BATTERY LABEL indicating when the battery must be exchanged.

FUSE
A fuse is placed inside the line voltage selector, check the sign next to the selector for correct fuse rating on each voltage range.

Display

Pulse Parameters The display is usually used for setting the pulse parame-

PULSE PERIOD	DELAY	DURATION	LEADING EDGE	TRAILING EDGE
45.0 -9	10.0 -9	20.0 -9	3.00 -9	3.00 -9
[..........]	[............]	+ 6.00	+ 3.00	0.00
BURST PERIOD	BURST PULSES	HIGH LEVEL	LOW LEVEL	OFFSET B

ters. All parameters are written above and below the display. The display shows the set value for each parameter.

Error Message When changing the pulse parameters, there are several

PULSE PERIOD	DELAY	DURATION	LEADING EDGE	TRAILING EDGE
#35.0 -9	10.0 -9	#20.0 -9	3.00 -9	3.00 -9
		Period > = 40.0 -9		
BURST PERIOD	BURST PULSES	HIGH LEVEL	LOW LEVEL	OFFSET B

ways to make an illegal combination. The display will then show an error message telling you the limit of the allowed range, (it is also indicating the other conflicting parameters). The error message is combined with a caution beep. The illegal value is not accepted by the generator and the outputs are not updated until all errors have been corrected.

Menus When using saved front panels and auxiliary functions or selecting the trigger source and trigger function, the operator is always guided by information on the display.

Pulse Terminology

As with other types of generators, characteristic signal purity is an important aspect in pulse generator applications.

Imperfections in the circuit under test cause the output pulse to exhibit waveform aberrations such as preshoot, overshoot, ringing, or tilt. Meaningful measurements require that the test signal does not display such aberrations or at least that these are within the limits of the specifications.

To provide familiarity with the commonly used pulse terminology, fig. 2-1 shows the various pulse parameters, and the following list defines most pulse terms.

- *Pulse repetition period*

Pulse repetition period of a repetitive pulse is the time between consecutive leading edges (measured at the 50 % amplitude points).

- *Pulse repetition frequency*

Pulse repetition frequency is the inverse of pulse repetition period.

- *Pulse delay*

Pulse delay is the interval between the leading edges of a double pulse (measured at the 50 % amplitude points).

For pre-trigger purposes, PM 5781 supplies a trigger pulse (CLOCK OUT) synchronous with the first pulse of a double pulse.

In single pulse mode, the PULSE DELAY is measured from the CLOCK OUT pulse to the SINGLE pulse.

The delay between the CLOCK OUT pulse and the MAIN OUTPUT pulse differs (about 8 ns) from the set pulse delay due to internal propagation delays.

- *Pulse duration* Pulse duration is sometimes called pulse width. It is the duration of a pulse between its leading and trailing edges (measured at the 50 % amplitude points).

  In PM 5781, pulse duration is set as the time from the start of the leading edge to the start of the trailing edge.

- *Transition time* Transition time is the interval between the 10 and 90% amplitude levels on the leading or trailing edge.

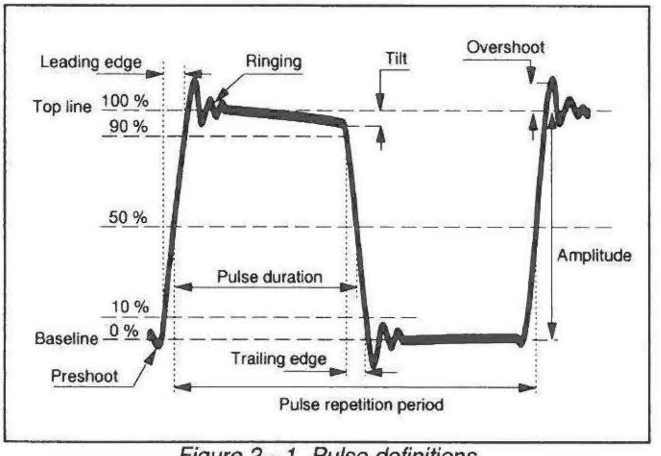


Figure 2 - 1 Pulse definitions

The terms rise and fall times are ambiguous, since they do not indicate what you can expect. Rise time is the transition of the leading edge of a pulse, irrespective of polarity or logic level. However in practice it is often (mis) used to indicate the time for a transition from a low level to a high level.

- *Jitter* RMS Jitter RMS is the short term instability of the specified time parameters such as pulse repetition period, pulse duration, or pulse delay. Jitter RMS is equal to the standard deviation.

- *High level* High level is the DC level of the most positive part of the pulse (top line or base line).

  In PM 5781 the set value is valid for both channel A and channel B when the OFFSET B = 0V and at 50 Ω load.

- *Low level* Low level is the DC level of the most negative part of the pulse (top line or base line).

  In PM 5781, the set value is valid for both channel A and channel B when the OFFSET B = 0V and has a 50 Ω load.

- *Pulse amplitude* Pulse amplitude is the level difference between the topline and the baseline of the pulse waveform.

- *Offset B*

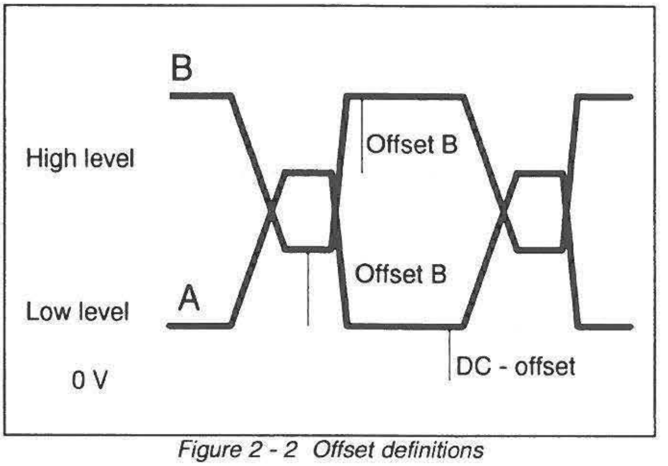


Figure 2 - 2 Offset definitions

  Offset B is the difference between the output pulse levels of channel A and channel B (at 50 Ω load).

- *DC - Offset* DC-offset is the offset between the pulse baseline and the 0V level. This term is not used in PM 5781.

- *Preshoot* Preshoot is a distortion preceeding a major transition.

- *Overshoot* Overshoot is a distortion following a major transition.

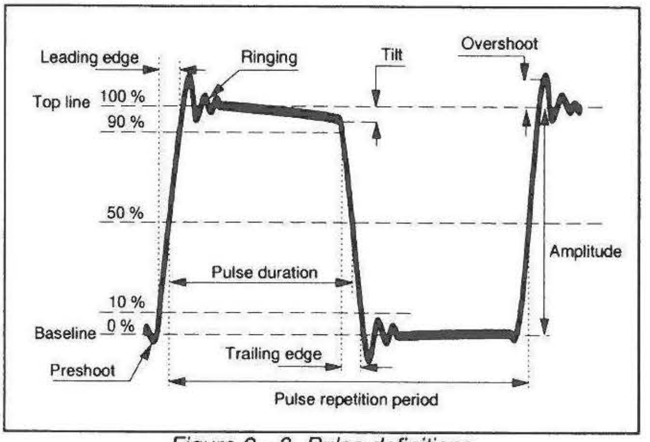


Figure 2 - 3 Pulse definitions

- *Ringing* — Ringing is a superimposed damped oscillatory waveform which, when present, usually follows a major transition; it does not include overshoot.

- *Repeatability* — Repeatability error is the difference between the values measured by a series of consecutive measurements during a short time. The measurements shall be made only on increasing or decreasing values over the whole measuring range. Repeatability error is expressed as a percentage of the measured value and does not include hysteresis and drift.

- *Tilt* — Tilt is a pulse-top distortion. Its value is taken as the fractional loss in pulse amplitude at the point just prior to the trailing edge.Tilt may be of either polarity.

- *Ramp non - linearity* — Ramp non linearity is the deviation from a straight line of the leading and trailing edges, mostly defined between 10% and 90% of pulse amplitude.

- *Baseline* — Baseline is the output level between the presence of pulses.

Installation

Unpacking

Confirm that the shipment is complete and check that no damage has occurred during transportation. If the contents are incomplete or damaged, immediately file a claim with the carrier. The Fluke sales or service organization should then be notified to repair or replace the pulse generator.

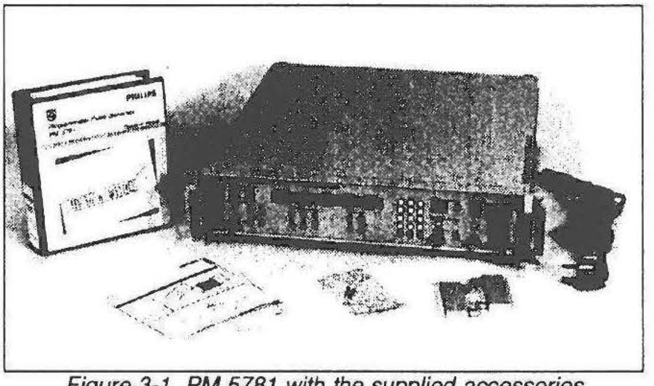


Figure 3-1 PM 5781 with the supplied accessories

Check list

The shipment should contain:

-Line cord
-Operators manual.
-IEEE-488 pocket guides, 3 pcs.
-Two extra fuses 1.6 A and 3.15 A, (5 x 20 mm).
-The pulse generator marked PM 5781.
-Bench-top conversion kit.

Line Voltage

Setting

The pulse generator must be connected only to an AC supply. Before connecting the pulse generator to the line, make sure it is set to your local line voltage. The line voltage selector at the rear of the generator can be set to the following ranges:

Position	Range	Fuse rating
100 V	90...110 V	3.15 A
120 V	108... 132 V	3.15 A
220 V	198...242 V	1.6 A
240 V	216...264 V	1.6 A

Please note that a fuse of different rating may be required if the voltage setting is altered.

WARNING: The pulse generator must be disconnected from all voltage sources when a fuse is to be replaced, or when the pulse generator is to be changed to a different line voltage.

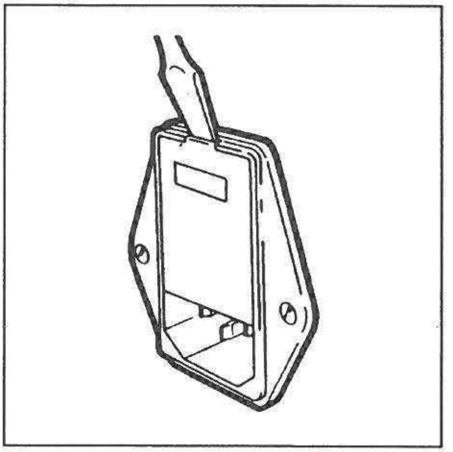


Figure 3-2 The voltage selector cover being released

- *To adjust the voltage setting,*

- Disconnect the power cable from the pulse generator.
- Release the voltage selector cover with a screwdriver
- Remove the voltage indicator camshaft.
- Reposition the camshaft to reveal the required voltage setting .
- Read the following section on "The Fuse" to ascertain the recommended fuse rating.
- Replace the fuse if necessary.
- Reinstall the voltage selector cover.

The Fuse

The pulse generator is protected by a fuse located in the voltage selector unit on the rear panel, see Fig. 3.2. The fuse holder, as shown in Fig. 3.3 is located in the slide on the right-hand side inside the voltage selector.

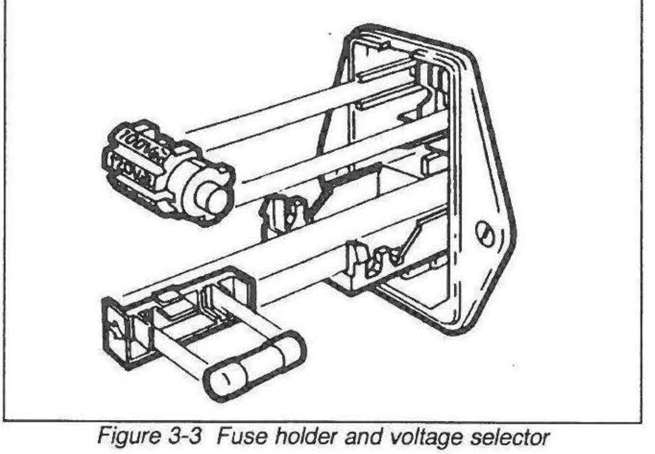


Figure 3-3 Fuse holder and voltage selector

On delivery, the pulse generator is fitted with fuse holders suitable for both European types 5 x 20 mm fuses as well as U.S. types 6.3 x 32 mm. The fuse holder positioned in the right- hand side position is in use while the other one is a spare unit.

- *To change a fuse*

- Disconnect the line cord from the pulse generator.
- Release the voltage selector cover with a screwdriver.
- Select the correct fuse by refering to the table below.

Fuse type	Service code
For 220/240 V - 1.6 A "slow-blow"	4822 253 30024
For 100/120 V - 3.15 A "slow-blow"	4822 253 30027

Grounding

Faults in the supply will make the pulse generator connected to it dangerous. Before the instrument is connected to the power line, you must satisfy yourself of the correct functioning of the protective ground in the line. Only then can the instrument be connected to the power line and only by using a three-conductor line cord. No other grounding method is permitted. Extension cables must always have a protective earth conductor.

WARNING: Never interrupt the protective grounding intentionally. Any interruption of the protective ground connection inside or outside the instrument, or disconnection of the protective ground terminal is likely to make the instrument dangerous.

If the instrument is transferred from a cold to a warm environment, condensation may cause a hazardous condition. Ensure therefore that the grounding requirements are strictly met.

Operating Position and Cooling

The pulse generator can be operated in any desired position. The generator is cooled by two fans mounted on the rear panel. Make sure that the air flow through the fans in the rear panel and the ventilation slots in both side panels are not impeded.

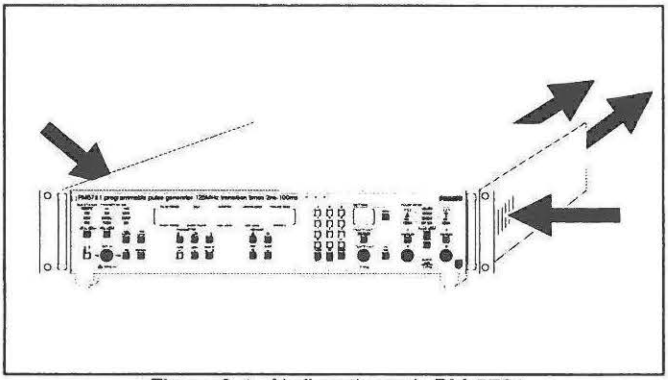


Figure 3-4 Air flow through PM 5781

Check at least once a month to ensure that there are two airstreams through the rear panel.

Changing handles

For table top use, a fold-down support is located underneath the pulse generator. If side pieces without handles have to be mounted, follow the instructions below.

Removing the covers

WARNING: The opening of covers or removal of parts, except those to which access can be gained by hand is likely to expose live parts and accessible terminals which can be dangerous to life.

- *Electrostatic discharge*

Almost all modern components have extremely thin conductors and metal oxide layers. If these layers are exposed to electrostatic discharge (ESD) they will break down or perhaps even worse, be damaged in a way that inevitably will cause a breakdown later on.

This electro-static discharge sensitivity of MOS and CMOS semiconductors has been known for quite a while, but today bipolar semiconductors and even precision resistors are known to be sensitive. So **consider all components, pc-boards and sub-assemblies as being sensitive to electrostatic discharge.** The following text explains how to minimize the risk of destroying components by knowing there is an ESD-problem and learning how to handle it.

ESD-sensitive options are packed in conductive containers marked with this symbol.

- Never open the container unless you are working in an ESD-protected work station.
- Use a wrist strap, grounded via a high resistance.

- Use a grounded tablet on your workbench.
- Never let your clothes get in contact with the sensitive equipment even when you wear a grounded wrist strap.
- Never touch the leads of the components.
- Never touch open connectors.
- Use ESD safe packing materials.
- Use the packing material only once.
- Keep papers and nonconductive plastics etc. away from your workbench, they may block the ground path.

- *Remove top and bottom covers*

1 Make sure that the line cord is disconnected from the generator. Although the power switch is in the off position, the line voltage is present at several places in the generator.

2 Loosen the two screws (A) in the top cover.

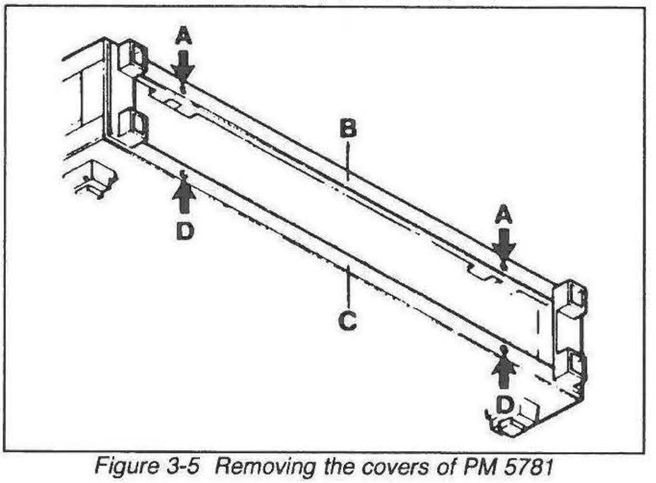


Figure 3-5 Removing the covers of PM 5781

3 Carefully lift the top cover (B) off the generator.

4 Turn the generator upside down.

5 Loosen the two screws (D) in the bottom cover.

6 Carefully lift the bottom cover (C) off the generator.

7 Take away each side piece by removing the two fixing screws. See fig 3-6

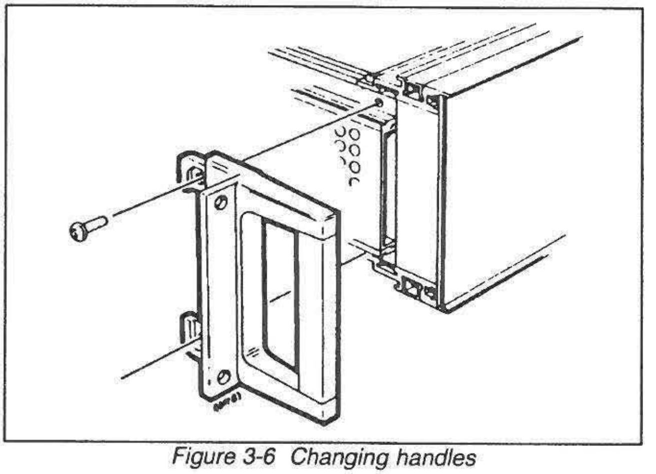


Figure 3-6 Changing handles

8 Mount the new side piece with the two screws.

9 Mount the covers.

Connection

For details about the bus interface cable, BNC- terminations and cables, see ordering information in chapter "specifications".

Getting Started

Introduction

This short introduction will help the first-time user to become acquainted with the PM 5781.

Check that the pulse generator is switched OFF and installed in accordance with the installation instructions (line voltage, grounding and so on).

Power

Press POWER on the left hand side of the panel, and the displays will light up. There are no output signals at power on; you must enable them.

The display now gives the information "Warm up in progress"; 30 seconds later the message changes to "Initial warm up completed"; and a second later the generator is ready for use.

The front panel setting present when the generator was last turned off is now recalled. If that setup is what you want, just enable any of the outputs.

Although the generator is now operating, some warm up time is required to meet published specifications. Thirty minutes after the power up, the generator gives the message "Warm up completed". Units with the selfcal system installed will also confirm "Temperature inside selfcal range", or "Temperature high / low selfcal recommended".

Bus Status

After a power ON, the generator starts in LOCAL mode (front panel controlled). The BUS STATUS display is then OFF. If the power was already ON when you started, the generator might be working in BUS mode.

To get control of it, press LOCAL/RESET, (check first).

Trigger Mode

- *Select trigger source*

The selected TRIGGER MODE is indicated in the display. If you want to change the trigger source, press TRIG MODE and the central display indicates the alternatives. Each alternative is identified by a digit. Select the proper digit with the keyboard on the right hand side of the panel.

PULSE PERIOD	DELAY	DURATION	LEADING EDGE	TRAILING EDGE
Select trig source				
1: Internal	2: External	3: Manual		
BURST PERIOD	BURST PULSES	HIGH LEVEL	LOW LEVEL	OFFSET B

When you touch the key on the keyboard the choice is made and the program continues to the selection of trigger function.

- *Select trigger function*

The display will now ask you to select trigger function.

PULSE PERIOD	DELAY	DURATION	LEADING EDGE	TRAILING EDGE
Select trig function				
1: Trig	2: Gate	3: Burst	4: Duration	
BURST PERIOD	BURST PULSES	HIGH LEVEL	LOW LEVEL	OFFSET B

External Input

When external trigger source is selected, the input must be set to positive or negative slope and 1 MΩ or 50 Ω (input) impedance. Press TRIG LEVEL, and the set level is indicated in the display. You can set a new value with the keyboard or the vernier. When you are ready, press the flashing ENTER key.

- *Manual*

The MANUAL key is used in the manual trigger modes and can be used to manually trig or gate the generator or to get manual pulse duration.

Pulse Mode

In the upper right corner of the front panel is a display showing the selected pulse mode. Press PULSE MODE underneath to get SQUARE, SINGLE, or DOUBLE pulses. (EXTERNAL DURATION is selected by the TRIG MODE.)

Pulse Parameters

A cursor on the central display indicates the parameter that can be set with the keyboard or the VERNIER keys. To select another parameter, use the three PARAMETER keys to move the cursor to the desired parameter.

Note: Pulse period and Burst period are set with an accuracy of four digits.

- *#*

A # sign indicates that the parameter setting might give an unstable or even impossible output pulse. The erroneous setting is not accepted by the generator and the output is unchanged from the most recent correct entry. The display indicates the limit for the parameter marked by the cursor.

- *Beeper*

A beeper gives an audible warning if a parameter value does not fit into the current setting.

Auto Set

Press this green key and let the AUTO SET function correct any erroneous setting. The AUTO SET function will attempt to set the pulse parameters to valid values, while holding the parameter at the cursor constant. The AUTO SET function confirms that the correction is successful. (It can fail due to locked parameters).

Lock

The ■ sign on the display indicates that the parameter value is locked and can't be changed. Pressing LOCK marks/unmarks the parameter value. All locked parameters are released with the keys CLEAR plus LOCK.

Track α, β

Allows you to make several parameters follow each other. Time parameters marked with the α sign will get the same relative change. Voltage parameters marked with the β sign changes with the same voltage.

Vernier

The four VERNIER keys are used to change the parameter values continuously. Use either fast or slow rate of change.

Keyboard

The fifteen key- keyboard is for selecting from the menus, mode selection as well as setting parameter values with decimal point and exponent (EE). CLEAR will clear the display value and also allow you to see the fourth digit set for pulse period and burst period. Enter the set value with ENTER. ESC returns you from a menu. VIEW is explained in the reference chapter under SETTING.

Setting

The SETTING display shows the number of a recalled setting (if current). If you change any function mode or parameter value the setting number indication will disappear.

Menu

The MENU key allows you to save and recall complete front panel settings. Saved front panel settings may be combined and recalled as sequences to enable complete testing procedures. See also SETTING in chapter "Reference". The MENU also allows you to reach some auxiliary functions such as SELFCAL (optional), GPIB address, debug ON/OFF and some tests. (See MENU in the Reference chapter.)

Complementary

The A and B outputs are complementary. With COMPL the ON/OFF times are interchanged on outputs A and B.

Outputs Disabled

After power ON the outputs are disabled and must be enabled with the DISABLED keys. All three outputs have 50 Ω source impedance and are protected against shorts and open circuits.

Clock Out

The clock output always gives a square wave out in internal trigger modes. The output levels (amplitude) may be selected to either TTL or ECL.

For Further Information

Read the relevant part of chapter "Reference".

Reference

Introduction

Introduction

In this chapter each part of the front panel is described in detail. The descriptions are grouped in functions according to the list below.

If you cannot find the word you are looking for in this list, check the index at the end of the manual.

Time Parameters

Pulse Period

Pulse Period

The pulse period may be set from 8 ns to 10 s.

To set the pulse period:
- Position the cursor on PULSE PERIOD.
- Set the PULSE PERIOD.

Four Digits

It is possible to set the pulse period with four digits resolution in the internal trigger mode.

To set the pulse period with four digits:
- Position the cursor on PULSE PERIOD.
- Use the keyboard to set the value, or:
- Press CLEAR/ESC and use the VERNIER keys to set the fourth digit.

Regulation

A regulation of the period time is included in the internal trigger mode. This regulation system works with an internal counter on a chip which gives a highly improved accuracy. The regulation system corrects any deviation from the set period time. The regulation (period control) function may be switched ON or OFF, see "Other Functions".

Duration

The duration may be set from 4 ns to 100 ms. The duration is set as the time from the start of the leading edge to the start of the trailing edge; see also fig 5-1.

To set the duration:

- Position the cursor on DURATION.
- Set the duration.

As can be seen from the figure, the set duration time will

not change if the time for leading or trailing edge is changed.

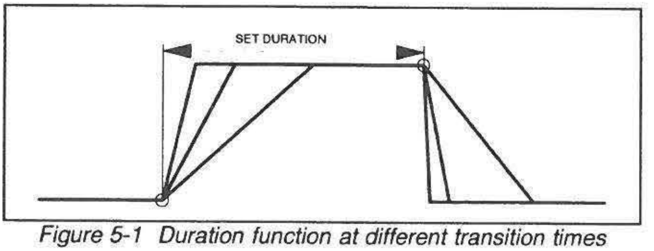


Figure 5-1 Duration function at different transition times

The bending knees used as definition points for the duration time will be slightly delayed for longer transition times.

The set duration is valid for both the first and the second pulse in a double pulse pair.

Delay

Double Pulse

The delay is the time interval between the leading edges of the two pulses in double pulse mode. The delay can be set from 8 ns to 100 ms.

To set the delay:

- Position the cursor on DELAY.
- Set the new DELAY value.

NOTE: In fast applications when the CLOCK OUT pulse is used as time reference the external output cables must be matched to give pulses that coincide at the device under test. The clock out pulse appears approximately 8 ns before the signal the output connectors.

Single Pulse

The DELAY time in single pulse mode is defined as the time from the clock pulse to the leading edge of a single pulse. Approximately 8 ns is added to the time set on the display (see note above), this is due to the fixed delay in the generator.

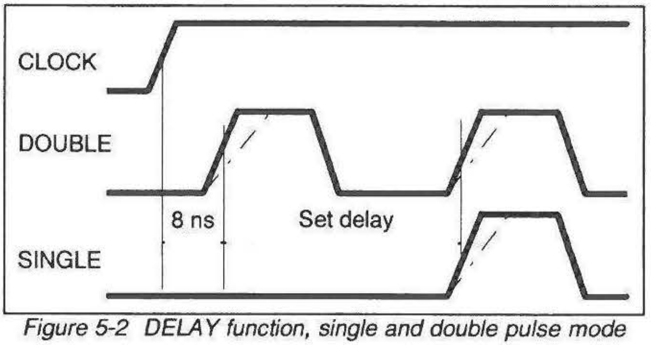


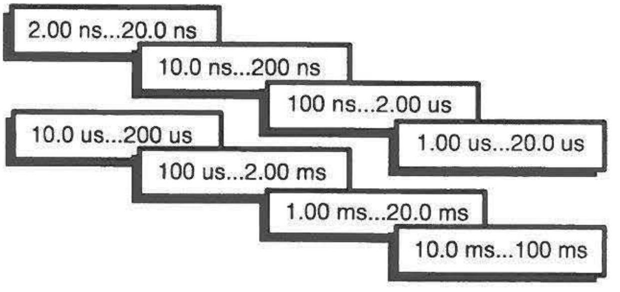
Figure 5-2 DELAY function, single and double pulse mode

The delay time may be set from 8 ns to 100 ms. The procedure for setting the value is the same as for double pulse.

Leading and Trailing Edges

The specification of the transition times are from the 10% to the 90% level of the pulse amplitude and vice versa.

The leading and trailing edges are individually adjustable but both must be within the same of the following ranges:



To set the transition time:

- Position the cursor on leading or trailing edge.
- Set the desired value.

If any of the transition time ranges are exceeded during the setting procedure, there will be an error message. This error message will disappear when the other transition time is set within the same range (if no other parameters are included in the error).

The leading and trailing edges have their starting points as indicated in fig. 5-1. As a consequence the pulse duration measured at 50% levels of the pulse amplitude changes when the transition times are not equal.

Burst

Burst

In this mode the internal oscillator is gated by a burst counter which counts up to the set number of output pulses and then waits for the next trigger pulse from the trigger source.

To select the burst mode:

- Press TRIG MODE.
- Select trigger source :
  Internal, the burst is initiated by the internal burst period generator

  External, the burst is initiated by the external trigger pulse

  Manual, the burst is initiated by the MANUAL key.
- Select 3 for BURST function

Internal Burst Mode

The internal burst mode allows output pulses to pass during the first 50% of the burst repetition period. In this time interval the burst counter will open the output gate, count up to the set number of pulses and then close the gate again. Note that the burst counter has a resolution of 4 digits why short burst periods may allow less then the expected number of burst pulses.

To set the burst (repetition) period:

- Position the cursor on BURST PERIOD
- Set the burst period.
- Position the cursor on BURST PULSES.
- Set the number of burst pulses.

To set the burst (repetition) period with four digits:

- Position the cursor on BURST PERIOD.
- Set the burst period with the keyboard, or:
- Press CLEAR/ESC and use the VERNIER keys.

External burst mode

In the external burst mode is the pulse period oscillator gated by the burst counter which is started by the external trigger pulse.

To set the burst counter:

- Position the cursor on BURST PULSES.
- Set the number of burst pulses.

Manual burst mode

In the manual burst mode, the pulse period oscillator is gated by the burst counter which is started by the key MAN-UAL.

Definitions

The BURST period time is the time between the starting point of two consecutive bursts. This is also indicated in fig. 5-3.

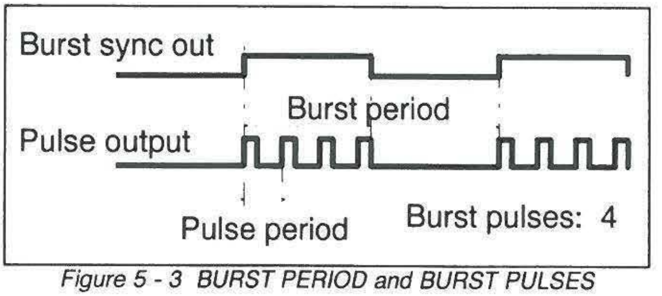


Figure 5 - 3 BURST PERIOD and BURST PULSES

Burst period is sometimes called burst repetition period.

Track α

During many kinds of measurements it is a demand for change of period time with preserved relations between delay, duration, and transition times.

The α-TRACK function makes two or more time parameter values follow each other. The values tied together with TRACK are marked on the display with the α sign.

To set tracking on a parameter:

- Position the cursor on a time parameter.
- Press TRACK, and the parameter is marked with "α".
- Repeat the above on the other time parameter.

The next time you change one of the marked parameters, the other marked parameters will follow with a constant ratio. When the pulse mode or the trig mode is changed with some of the parameters tracked, it might happen that an error occurs between tracked time parameters. In such a case the tracking is automatically removed.

If an inactive parameter gets out of range due to tracking, then tracking will be removed from it.

To remove tracking:

- Position the cursor on the parameter.
- Press TRACK, and the parameter is no longer marked with "α".

Lock

The LOCK function freezes one or several of the parameter values to avoid unwanted changes.

To set locking:

- Position the cursor on the parameter.
- Press LOCK and the value is locked.

The locked parameters are marked with the sign ■ on the display.

NOTE: If a parameter with tracking on is locked, the other parameters with the same tracking sign will be locked as well.

To remove locking:

- Position the cursor on the parameter.
- Press LOCK (pressing and holding CLEAR when pressing LOCK removes all locking signs).

This page is intentionally left blank.

Voltage Parameters

High / Low Levels

High / Low Levels

The output pulses from PM 5781 are set as high and low levels. The levels can be set within the range ±10 V with a maximum difference of 10 V.

To set the high level:

- Position the cursor on HIGH LEVEL.
- Set the new HIGH LEVEL.

The low level is set in the same way. The set high and low levels are always referring to the A channel. See also fig. 5-4.

NOTE: To protect the loads during a shift of output level, the output amplitude goes towards zero during the change over when the internal attenuators are shifting range.

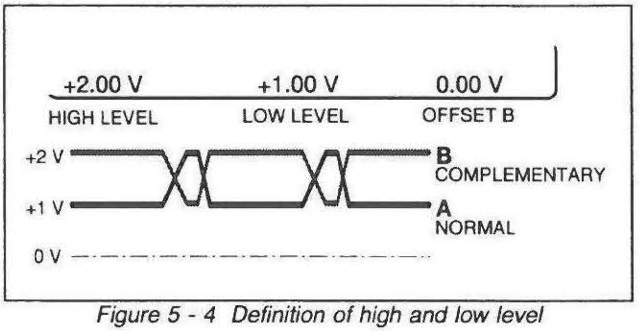


Figure 5 - 4 Definition of high and low level

The A and B outputs are complementary. (ON/OFF times are reversed.) Changing from normal to complementary pulse mode will not change the high and low levels of the outputs (see also Voltage Limits).

Offset B

Offset B means that channel B gets an offset voltage added to the high and low levels set for channel A. The added voltage (offset B) then indicates the voltage difference between the two channels.

To set OFFSET B:

- Position the cursor on OFFSET B.
- Set the offset value.

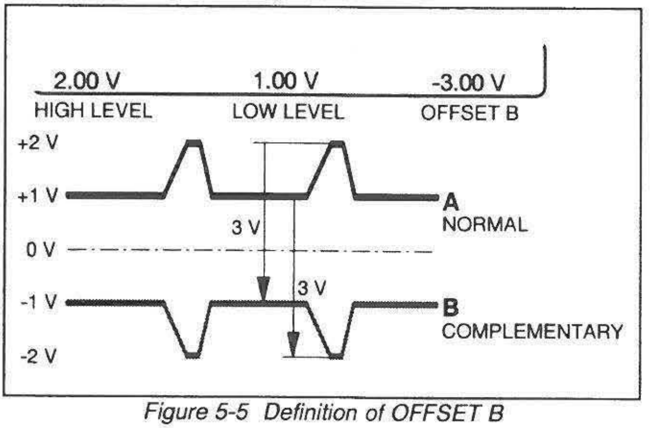


Figure 5-5 Definition of OFFSET B

Track ß

A pulse with fixed amplitude but variable offset from the zero-line can be achieved on both channels by use of the β-tracking facility.

To set tracking:

- Position the cursor on HIGH LEVEL.
- Press TRACK, and the value is marked with the " β" sign.
- Repeat the procedure for the low level.

If one of the levels is changed, the other will follow, keeping the difference (the amplitude) unchanged. Offset B can not be included.

To remove tracking:

- Position the cursor on the parameter.
- Press TRACK, (plus CLEAR for both signs).

Lock

The LOCK function freezes one or several of the parameter values to avoid unwanted changes. These changes can originate from a manual or bus setting of a new value or from the AUTO SET function.

To set locking:

- Position the cursor on the parameter.
- Press LOCK and the value is locked.

For example, the LOCK function may be used to lock the high or low output level to a certain value to protect the connected circuits. The locked parameters are marked with the sign ■ on the display.

NOTE: If a parameter with tracking on is locked, the other parameters with the same tracking sign will be locked as well.

To remove locking:

- Position the cursor on the parameter.
- Press LOCK, (plus CLEAR for all signs).

This page is intentionally left blank.

Error Correction and Auto Set

When a new parameter value has caused an error message, the setting may be corrected in several ways.

Auto Set

Auto Set

The AUTO SET function will attempt to set the pulse parameters to valid values, while holding the parameter at the cursor constant.

To initiate the AUTO SET function:
• Press AUTO SET.

If you want some parameters to remain unchanged, use the LOCK function to protect them before initiating AUTO SET. If the AUTO SET function cannot solve the problem due to too many restrictions, then it will say "Auto set failed". In this case the faulty setting will return and the error message will remain, urging you to correct the setting manually.

The AUTO SET function will be limited in its performance when a restriction due to tracking is present.

If the AUTO SET function is initiated when there is no error present. Then the display will give the message "AUTO SETTING", then "No error present"

Vernier and Reset

Vernier

The VERNIER can be used to correct an erroneous parameter.

To initiate the VERNIER correction:

- Position the cursor on the erroneous parameter.
- Press the correct "fast" VERNIER key.

The parameter value will **jump back** to within the range.

Reset

To return to the last correct front panel setting:

- Press RESET.

This page is intentionally left blank.

Trigger Mode

Trigger Mode

The TRIGGER MODE display shows the selected trigger source and trigger function. If the selected trigger mode can't be combined with the set pulse mode, then the pulse mode will change automatically when the trigger function is selected.

To select trigger mode:
- Press TRIG MODE.
- Select trigger source.
- Select trigger function.

Internal

- *Trig* — The pulse period oscillator provides the pulse period.
- *Gate* — The pulse period oscillator is in this mode gated by the burst period oscillator. The duty factor of the gate signal is 50%. The maximum burst period time is 4000 s.
- *Burst* — The pulse period oscillator is in this mode switched ON and OFF by the burst counter. The burst counter starts counting up to the set number of output pulses at each trigger pulse from the burst period oscillator. As the burst period oscillator has crystal accuracy it is possible to set (in 1 µs steps) an output pulse with high stability.

External

- *Trig*

  Each output pulse is triggered by the external input signal.

- *Gate*

  The pulse period oscillator gives the period time but it is gated by a signal on the external input. This external signal turns the pulse period oscillator ON and OFF. Any initiated output pulse, either SINGLE or DOUBLE, will always be completed.

- *Burst*

  The pulse period oscillator is gated by the burst counter allowing a preset number of pulses to pass. The burst of pulses is initiated by the external input pulse.

- *Duration*

  The external input pulse is shaped and used as the time reference instead of using the pulse period oscillator. This means that the output pulse will have the same period and duration as the external pulse. The DELAY and DURATION functions are disabled in this mode. All three outputs will give pulses with the same duration as the input pulse.

Trigger Mode

Manual

• *Trig*	The pulse period oscillator gives one pulse each time the MANUAL key is pressed. There will on the outputs be a single or a double pulse.
• *Gate*	The pulse period oscillator is running and producing pulses for the output as long as the MANUAL key is pressed (or released if negative slope polarity).
• *Burst*	The pulse period oscillator is gated by the burst counter allowing a preset number of pulses to pass. The burst of pulses is initiated by pressing the MANUAL key.
• *Duration*	When the MANUAL key is pressed, the output level changes from low to high or vice versa.

External Input

External Input

This is the external input for the three functions trigger, gate, and burst. The input signal can have all kinds of wave forms therefore it is necessary to shape the signal so the timing circuits can handle it. The input sensitivity (hysteresis) is 500 mVpp. For low amplitude signals with long transition times this means that the actual triggering might occur slightly later then what could be expected.

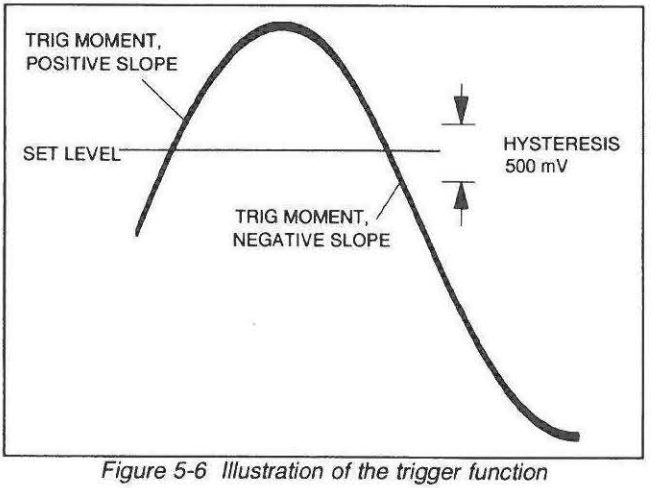


Figure 5-6 Illustration of the trigger function

Whether the input signal triggers the generator or not is indicated in the TRIGGER MODE display with the TRIG'D indicator. This indicator is blinking when triggering occurs, it is off when the trig level is set too high and on when the trig level is set too low. See fig. 5-7.

Impedance and Slope

50 Ω or 1 MΩ

Input impedance for this input is selectable as 50 Ω or 1MΩ, which calls for some attention when a signal source is connected. In the 50 Ω position, the input can withstand a maximum of 10 VRMS, but not more. For the 1 MΩ position, the limit is 50 V at frequencies up to 10 MHz.

Be careful to match the input impedance correctly to avoid reflections. Reflections occurring on the external input can cause incorrect triggering. Note also the capacitance is 45 pF in parallel in the 1 MΩ setting.

+ / -

The NEG SLOPE key selects between positive going and negative going edges of the signal applied to the external input. The slope selection also inverts the function of the MANUAL key.

Trig Level

Trig Level

The TRIG LEVEL key is used to view and change the trigger level setting. When you press the TRIG LEVEL key, the display shows the present trigger level setting. If you want to change it, you may set a new value.

To set the trigger level:

- Press the TRIG LEVEL key.
- Set the new trigger level value, press ENTER.
- Press ENTER to leave the trigger level menu.

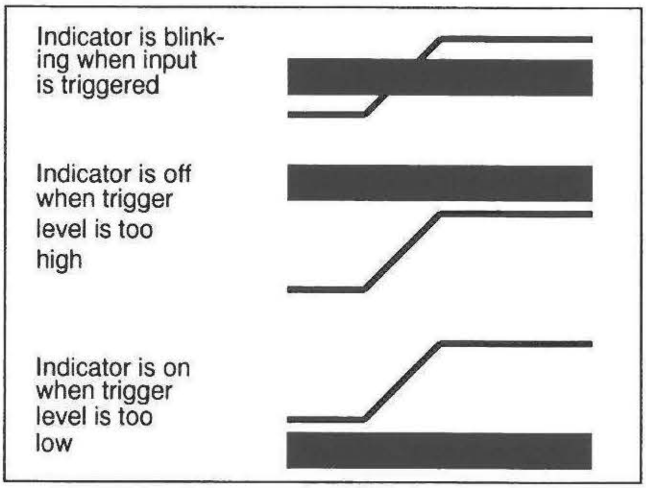


Figure 5 - 7 Tristate indicator for trig level setting

Pulse Mode

Pulse Mode

Pulse Mode

You can select square, single or double pulse with the pulse mode key. The PULSE MODE display shows the choice.

To select pulse mode:

- Press the PULSE MODE key.
- Select the pulse mode.

EXTERNAL DURATION is a trigger mode, selected with the TRIG MODE key. (see "Trigger mode")

- *Square*

Square wave mode gives a 50 % duty cycle. The DELAY and DURATION functions are not active in this mode.

- *Single*

One single pulse is produced for each trigger event. The output pulse is delayed from the clock pulse. The delay time is set with the DELAY parameter. The fixed internal delay of approximately 8 ns is added to the set DELAY value.

- *Double*

Two identical pulses, like the single pulse are produced for each trigger event. The second pulse is delayed from the first pulse with the set DELAY.

- *External Duration*

This indicator shows that the trigger mode "External Duration" is selected.

Complementary

When the complementary function is activated, with the COMPL key, the ON/OFF time for each channel is reversed.

Outputs

Outputs A and B

Outputs A and B

The two pulse outputs are complementary. The B output may be offset from the A output, this offset is indicated in the central display as OFFSET B. This means that the values for high and low level are always true only for the A output, but the B output high and low level may carry an offset.

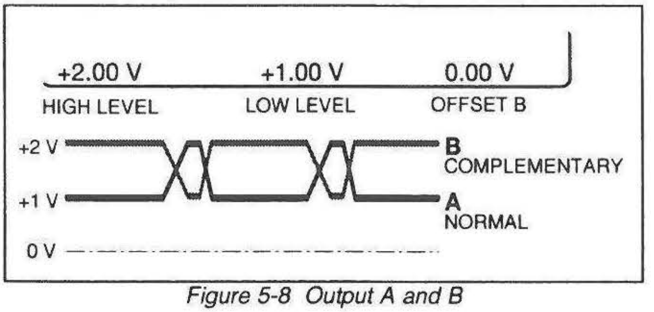


Figure 5-8 Output A and B

Each output may be disabled with a DISABLED key, located above the BNC connector. The outputs have an output impedance of 50 Ω which remain on 50 Ω even when the outputs are disabled. The outputs can withstand both short and open circuits. In all pulse and trigger modes, the output pulses are delayed from the CLOCK PULSE.

Clock Out

The CLOCK OUT outputs the internal clock pulse before the rise and fall time circuits. It gives a symmetrical pulse in internal trigger modes. In external and manual trigger modes it works as a pulse shaper with the same timing as the external or manual trigger pulses. The generator outputs the CLOCK OUT pulse about 8 ns before the first pulse in double pulse mode.

This makes the clock signal ideal as:

Trigger signal, for oscilloscopes.

Time reference. When the CLOCK OUT pulse is used as time reference (in fast applications), the external output cables must be matched to give pulses that coincide at the device under test, i.e. the cable on the clock output must delay the pulse 8 ns more than the cable on the A and B output.

Transition time

The CLOCK OUT pulse has a very fast transition time (typical 1.5 ns). This means that you must be extra careful with the load impedance to preserve a good waveform.

Amplitude

The Clock pulse levels may be changed, with the TTL/ECL key:
TTL levels, 0 V and +2.5 V into 50 Ω ,
(0 V and +5 V into high load impedance).
ECL levels, -0.8 V and -1.7 V into 50 Ω.

Source impedance

The source impedance is 50 Ω .

Disabled

The output signal is automatically disconnected from the output connector when the level is changed, or manually when the DISABLED key is pressed. The output impedance remains 50 Ω when the output signal is disabled.

Burst Sync Out

Burst Sync Out

A Burst Sync pulse with the levels –0.4 V and –0.8 V is available, on the rear panel. The level is high (–0.4 V) during the pulse burst duration.

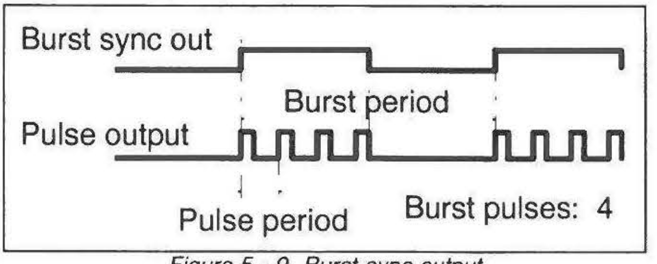
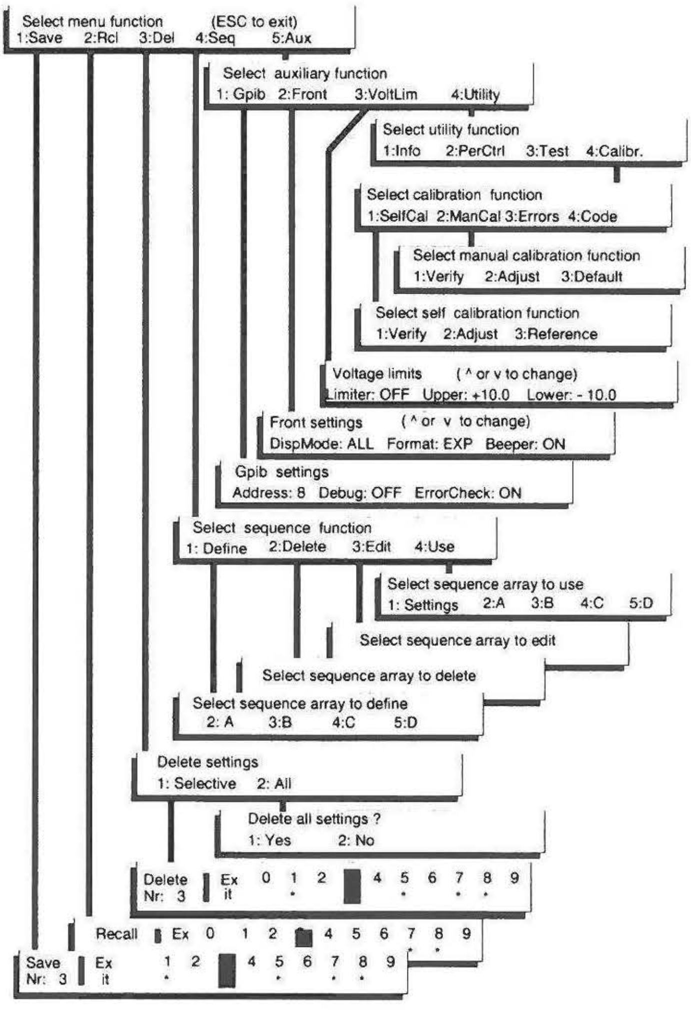


Figure 5 - 9 Burst sync output

Menu

Menu Tree

The MENU key gives access to many functions in PM 5781, that can't be seen on the front panel. To find out everything about these hidden functions, you have to look at each of the choices.



The first four choices in the first menu: SAVE, RECALL, DELETE and SEQUENCE are all described under "SAVED SETTINGS".

The fifth choice in the first menu: AUXILIARY is described under "OTHER FUNCTIONS".

Saved Settings

Setting

Setting

In PM 5781 it is possible to save up to 49 different settings. Each saved front panel setting is given a setting number when it is saved. When a saved front panel setting is recalled, its setting number is shown in the SETTING display.

As soon as any value or mode is changed, the front panel setting will no longer be equal to the saved setting and the setting display will be blanked.

Note that in the **VIEW** mode the whole front panel shows one saved front panel setting together with a blinking "VIEW No:" in the display, while another setting is active on the outputs. This may be confusing but the setting display will always indicate the setting on the outputs.

These saved front panel settings may also be combined to form **sequences** (see sequence on following pages). In these sequences, the saved front panel settings are arranged in any order desired. Settings may even be repeated on several locations in the sequence.

The SETTING display indicates the present (active) front panel setting. When handling saved front panel settings, it is good practice to keep an eye on this display because it shows which front panel setting is active on the outputs.

Save

When you have a front panel setting that you want to repeat in the future either alone or as a part of a test program (a SEQUENCE), save it.

To save a setting:

- Press MENU.
- Select 1 for SAVE.

You will now get a second menu starting with "SAVE Nr: __" and continuing with the numbers 1 through 49. Selection of a certain number is made with the keyboard or the PARAMETER keys, the central parameter key will move the cursor to the next decade.

SAVE	Ex-	1	2	3	4	5	6	8	9
Nr:3	it		*			*	*		

A number with a * underneath means that a front panel setting is already saved under this number. If you select an occupied number, the old setting will be over written.

The setting number (lower left corner of the display) may also be set with the keyboard (or the VERNIER keys).

Press ENTER, the display will shift back to normal again and the front panel setting is saved under the setting number you entered. Only front panel settings which are correct (without error messages) can be saved.

To leave this setting menu without saving any setting:

- Press the CLEAR key (the cursor moves to EXIT).
- Press ENTER (or CLEAR again).

Recall

To recall a saved front panel setting means that the present setting will be replaced with a saved one. Before you continue, decide whether or not the present front panel setting will be deleted or eventually saved before another one is recalled

NOTE: The old front panel setting is active until another setting number shows up in the setting display window.

Recall

To RECALL a setting:

- Press the MENU key.
- Select 2 for RECALL.
- Select any number from 0 to 49.
- Press ENTER.

Number 0 is the default setting (see also "default"), numbers with a saved setting are marked *.

CAUTION: A recalled setting will be active when recalled. Be sure your connections and the device under test will not be harmed by the new output levels. Before recalling see preview below.

To preview a setting before it is activated:

- Select the number as above.
- Press VIEW (this preview will not change the outputs).

The selected setting number (lower left corner of the display) may be changed with the VERNIER keys or set with the keyboard. Changing the decade is done with the parameter key in the middle.

To leave this menu:

- Press CLEAR (the cursor moves to EXIT).
- Press ENTER (or CLEAR again).

Delete

To delete a saved front panel setting:

- Press the MENU key.
- Select 3 for DELETE.
- Select 2 to delete all saved settings.
- Select 1 for selective delete of some saved settings.
- Move the cursor to a setting that you want to delete.
- Press ENTER, to "D" mark the setting number. (pressing ENTER again will unmark the number). Mark also the other setting numbers that you want to delete.
- Use the CLEAR key to move the cursor to the exit position.
- Press ENTER or ESCAPE now to delete all the settings marked "D".

Sequence

The SEQUENCE MODE is designed for applications where several front panel settings are frequently used. The handling of sequences is divided into four parts: **define** a sequence, **delete** a sequence, **edit** a sequence and **use** a sequence.

The SETTING display indicates the setting number of the present (active) front panel. In handling saved front panels, it is good practice to keep an eye on this display as it shows which front panel setting is active at the outputs.

Define

To define a sequence of front panel settings, simply make a list of setting numbers in the sequence memory.

To define a sequence:

- Press the MENU key.
- Select 4 for SEQUENCE.
- Select 1 to DEFINE.
- Select one of the letters A to D as the name for this sequence; press ENTER.
- Now, select the saved settings in the same order as you want to recall them, position the cursor and press ENTER (the settings will be marked A0, A1,...).
- To unmark a setting press ENTER again.
- When finished, leave this menu by moving the cursor to Exit, and then press ENTER.

Memory locations (1 through 49) with a saved setting are marked with a *. A maximum of 10 settings may be saved in each sequence. The same setting may be used several times.

Delete

DELETE, allows the deletion of any of the saved sequences. This means that the sequence memory will be cleared, but the saved front panel settings remain in the settings memory.

To delete a sequence:
- Press the MENU key.
- Select 4 for SEQUENCE.
- Select 2 for DELETE.
- Select sequence array to delete, A, B, C, or D.

Edit

EDIT is used to rearrange any sequence of settings.

To edit a sequence:
- Press the MENU key.
- Select 4 for SEQUENCE.
- Select 3 for EDIT.
- Select sequence array to edit A, B, C, or D.
- Position the cursor in the sequence.
- Select 1 to add a new setting number, or select 2 to delete a setting number from the sequence. Confirm by pressing the ENTER key.
- Press ENTER again to leave the EDIT menu.

Use

The USE mode, is designed for calling up (running through) stored sequences. You may choose either the full list of saved settings (SETTINGS) or one of the sequences you have defined (A, B, C or D).

To use a sequence:
- Press the MENU key.
- Select 4 for SEQUENCE.
- Select 4 for USE.
- Select all settings or a sequence (A, B, C, or D).

Sequence

Now you may change from one front panel setting to another in numerical order (or according to the selected sequence) simply by pressing the cursor keys for left or right. The parameter values for each setting are not displayed in this mode. See also VIEW.

To leave this menu:

- Press CLEAR three times.

View

Sometimes you want to see all the parameter values of a certain saved front panel setting. You can do this without interrupting the presently used front panel setting.

The VIEW mode has been created for this purpose. It works in the following setting menu modes: Save, Recall, Delete, Sequence Define, Sequence Edit, and Sequence Use. The VIEW mode works in combination with the menu in all the setting memory positions listed.

To view a saved setting:

- Position the cursor on the setting number that you want to study.
- Press the VIEW key. The display will show all the parameter values of the selected front panel setting, together with a flashing number of the selected setting.

NOTE: The outputs are not changed in this mode. The previous front panel setting remains. If a saved front panel setting was active, its setting number will remain in the setting display.

To return from the VIEW mode:

- Press any key.

If the Sequence USE mode is combined with VIEW, the USE function will remain; only a flashing sign "SEQUENCE No ..." will be added to the central display.

Default Setting

Recalling the default setting is an easy way to set the generator to an output a pulse with a known value on all parameters.

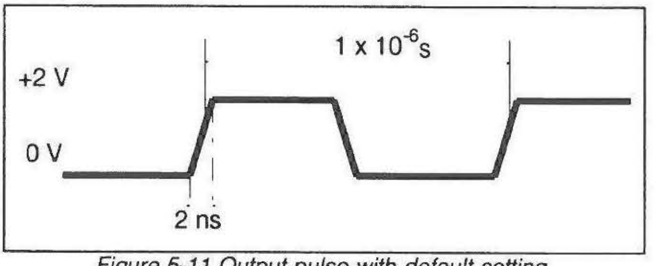


Figure 5-11 Output pulse with default setting

To get a default setting:

- Press the MENU key.
- Select 2 for RECALL.
- Select 0 + ENTER for the default front panel setting.
- Press any of the DISABLED keys to enable an output.

Outputs A and B will give positive pulses going from zero to plus 2.00 V.

CLOCK OUT will give a positive pulse with TTL levels.

Do not forget to use a 50 Ω load because the waveform will be distorted and the voltage levels will be inaccurate.

The complete list of all front-panel-operated parameters set by the DEFAULT function is given below.

Pulse parameters

Period	1 µs	Leading edge	2 ns
Delay	250 ns	Trailing edge	2 ns
Duration	100 ns	Burst	5
Burst period	100 µs	High level	2 V
Offset B	0 V	Low level	0 V
Tracking	OFF	Locking	OFF
Volt limit	-10 / +10V		

Output

Outputs (all)	OFF
Clock out level	TTL

Pulse mode

Pulse mode	square
Complement	OFF

Trigger

Trig source	internal
Trig function	trigger

Input

Trig level	1.00 V
Trig slope	positive
Impedance	high

Help

Debug	OFF
Error disp	ON

System

Disp mode	All
Limiter	OFF
Error check	ON
Period contr	ON
Disp format	Exponent
Beeper	ON

This page is intentionally left blank.

Other Functions

GPIB

Some GPIB parameters such as GPIB-address, debug and error check, can be set via the front panel.

To set the bus parameters:

- Press the MENU key.
- Select 5 for AUXILIARY.
- Select 1 for GPIB.

Address for the bus operations may be set to any number (0 through 30).

Debug ON means that the bus commands received are shown in the central display.

Error Check ON means that all set parameters are checked versus each other (see Error message). This check can be switched OFF to save some time when extra fast setting via the bus is required.

Front Panel Disabled

In some situations you may want to have the front panel disabled, i.e., to hide specific parameter values, to make execution of bus commands faster.

- Press MENU
- Press 5 for AUX
- Press 2 for FRONT
- Select:

  ALL to show all the values in the different displays.

  STATE to disable the central display. All other indicators and displays will work.

  OFF to disable the central display and all other indicators and displays, except bus status.

To enable the front panel:

- Check the bus status. If the REMOTE or LLO indicator is ON, then the generator is under control of the GPIB.

  Remote If you decide to interrupt the program that is running on the bus, press LOCAL/RESET. The generator is now set for front panel operation, but the front panel is still disabled. See below.

  LLO Check the bus program that is running before you interrupt it. Switch it OFF either with the controller (see controller manual) or with the generator by switching the generator to 0 (off) and then to 1 (on) again. The generator is now set to front panel operation, but the front panel is still disabled.
- Press the MENU key. The central display lights up again (but only in this mode).
- Select 5 for Auxiliary functions.
- Select 2 for Front.
- Select Display mode ALL

Front Panel Disabled

Another way to switch on the front panel again, is to recall a saved front panel setting that was specified with enabled front panel:

To recall a saved front panel setting:

- Press the MENU key.
- Select RECALL.
- Select a marked setting number, and press ENTER.

Front Panel Format

On the display, the value of the time parameters are expressed either in units: ns, μs, ms and seconds on the display or in engineering units. These formats may be changed.

To change the front panel format:

- Press the MENU key.
- Select 5 for AUXILIARY.
- Select 2 for FRONT.
- Select FORMAT.
- Select EXPONENT or UNIT.

Exponent will make all the units of the parameter values be indicated as exponents.

Unit turns the set exponents into units on the display.

Beeper

Beeper

The beeper will give a warning sound when an error is detected when changing parameter values. The BEEPER function may be set ON or OFF.

To set the beeper:

- Press the MENU key.
- Select 5 for AUXILIARY.
- Select 2 for FRONT.
- Select BEEPER.
- Set the beeper ON or OFF with the VERNIER keys.

Voltage Limits

To protect a connected circuit, you can restrict the output levels of PM 5781 to certain values set in the VOLTAGE LIMIT mode. These levels are set in volts and are consequently valid both for the A and the B output channels.

To set voltage limits:

- Press the MENU key.
- Select 5 for AUXILIARY.
- Select 3 for VOLTAGE LIMITS.
- Select limits ON.

- Position the cursor on Upper level.
- Set the value for upper level.
- Position the cursor on Lower level.
- Set the value for lower level.

CAUTION: The voltage limits are saved in memory and may be changed if you recall another front panel setting.

Period Control

Period Control

The control of the internally generated pulse period performed by the built in timer/counter may be switched ON or OFF.

To set period control:

- Press the MENU key.
- Select 5 for AUXILIARY.
- Select 4 for UTILITY.
- Select 2 for PERIOD CONTROL.
- Set the period control ON or OFF with the VERNIER keys.

Test

Internal test programs are available for the major parts of the generator.

To initiate the test program:

- Press the MENU key.
- Select 5 for AUXILIARY.
- Select 4 for UTILITY.
- Select 3 for TEST.
- Select 1 for ALL.

The program will now run a test of all the major parts of the logic. Selective tests of the individual parts can also be run: PROM, RAM 1, RAM 2, Front and Counter. Each test is confirmed as passed or failed.

For further information about the testing procedure, please refer to the service manual.

Information

This function gives some information about the temperature and calibration status of the generator.

To see the information menu:

- Press the MENU key
- Select 5 for AUXILIARY
- Select 4 for UTILITY
- Select 3 for SERVICE
- Select 1 for INFO

- *Temperature*

The display will now give the temperature in degrees centigrade measured inside the cabinet.

PM 5781 compares this temperature with the temperature when the generator was last calibrated (Cal: XX°C). If the temperatures deviates more than 5°C, the display shows HITEMP or LOTEMP. If the temperature is within the range of ±5°C, the display shows CALIBR.

- *Calibration temperature*

The display will also give the temperature inside the cabinet last time a calibration was made.

- *Elapsed time*

The elapsed time indicator gives the total operating time measured from when the generator was first turned on.

- *Warm-Up*

This indication will tell you what remains of the warm-up time. When the 30 minutes warm-up time has passed, the display indicates "Warm up completed", and the warm-up indication changes to "OK".

Manual Calibration

- *Performance check* This function gives a detailed guide to verify the specification of the generator. See chapter "Performance check" in this manual for further information about the procedure.
- *Adjustment* The Adjustment function is used for software adjustment of the generator. It is protected by an access code in order to avoid unauthorized changes of the calibration constants. See chapter "Calibration and Adjustment" in the service manual for further information.
- *Default* The Default position in this menu is used to set all the software calibration constants to default position.

DO NOT USE THIS FUNCTION! This function is intended for service. The generator will no longer conform to the specification when the default calibration constants is used. It requires at least a software adjustment procedure to reach the specification for all ranges.

Self Calibration

An Self Calibration function is included in generators with the type number index /02.

SELF CALIBRATION is used for three different purposes:
- to verify the specification of the generator (VERIFY)
- to optimize the accuracy (ADJUST)
- to list the accuracy of the measured parameter values, (CALIBRATION REPORT) available via the bus.

The time and voltage parameters of all ranges of PM 5781 are in the Self Calibration procedure compared with internal references - a crystal controlled oscillator and precision voltage sources. These references may in a simple way, described on following pages, be checked against external frequency and voltage standards.

The VERIFY mode of the Self Calibration will check some calibration points on every range and make a list of eventual errors.

The ADJUST mode of the Self Calibration will make an adjustment of the software calibration constants for the calibration points on every range. If the software is unable to correct a range to the desired optimized accuracy then it is given an error number which is indicated afterwards.

To initiate the "VERIFY" function:
- Press the MENU key.
- Select 5 for AUXILIARY.
- Select 4 for UTILITY.

- Select 4 for CALIBRATION.
- Select 1 for SELFCAL.
- Select 1 for VERIFY and the program starts after an extra check question.

It takes about 3 minutes to perform the verification test while the display shows how the testing proceeds through all the measuring points. During the Self Calibration all outputs are disabled.

If a parameter value is found not to be within the accuracy it is given an error number.

When finished the display will tell you that the verification is completed and the number of errors found. If an error has been detected, do the following :

- Press ESCAPE twice to get back to the menu where you selected calibration function.
- Select 3 for ERRORS to see which measuring point and what kind of error the Self Calibration function has detected.

If no errors are found press ESCAPE until you are out of the menu

To initiate the "ADJUST" function:

- Press the MENU key.
- Select 5 for AUXILIARY.
- Select 4 for UTILITY.
- Select 4 for CALIBRATION.
- Select 1 for SELFCAL.
- Select 2 for ADJUST.
- Type the requested access code and press ENTER. If the access code is not known to you, refer to

authorized personnel or the service manual for further information.

- Answer the check question with yes and the "Adjustment" program is started.

The "ADJUSTMENT" program will optimize the adjustment of each parameter range. If it, on any range, fails to reach the improved accuracy specification as specified in the chapter "Specifications" under the heading "After Self Calibration" it is regarded as an error and is given an error number.

To analyze detected errors:

- Press the menu key.
- Select 5 for AUXILIARY.
- Select 4 for UTILITY.
- Select 4 for CALIBRATION.
- Select 3 for ERRORS.

Eventual calibration errors will now be displayed, starting with the first detected error.

Make a note of the error and use the parameter keys to continue to the next error. When they are all passed through the display will say "No more errors".

Remember that if these errors are detected in the "Adjustment" mode the generator may still be inside the specified accuracy for the temperature range ±5°C from calibration temperature. To check that, run the "VERIFY" program when you have noted down the error information given.

To remedy the error, make a hardware adjustment according the service manual, chapter "Calibration and Adjustment", or contact your nearest Fluke Service centre.

To check the references:

- Press the menu key.
- Select 5 for AUXILIARY.
- Select 4 for UTILITY.
- Select 4 for CALIBRATION.
- Select 1 for SELFCAL.
- Select 3 for REFERENCE.
- Type the requested access code and press ENTER. If the access code is not known to you, refer to authorized personnel or the service manual for further information.
- Select 1 for FREQUENCY.
  The generator will now output a squarewave with a frequency of 1.000 MHz. To check this you need a frequency standard with an error of less than 1×10^{-6}. The internal frequency standard can be set by hardware adjustment.
- Select 2 for VOLTAGE.

To adjust the internal voltage reference you need to have a DC-voltmeter with an inaccuracy of less than 1 mV on the 10 V range.

- Use the parameter keys to select either plus or minus 10 output voltage from channel A.
- Measure the voltage and change the DAC value if it is needed to get the correct output voltage.
- Continue with the other output polarity on channel A and then both plus and minus 10 V on channel B.
- Press ENTER to continue in the program. You are now offered the choice to save the corrected values, to resume the measurements or to leave the menu without saving the changes made.

Access code

Access code

The adjustment of the output parameters is in PM 5781 set with both hardware and software constants. These software constants can be changed with manual calibration or the Self Calibration function.

To safeguard the generator against unintentional or unauthorized changes of the software constants they are protected by an access code. The defaultcode (1875) used at delivery may be changed by the user to his own code with maximum 12 positions. The method to change the code is described below.

To set a new access code:

- Press the MENU key.
- Select 5 for AUXILIARY.
- Select 4 for UTILITY.
- Select 4 for CALIBRATION.
- Select 4 for CODE.
- Enter the old access code as requested. If the old access code is not known to you, refer to authorized personnel or the service manual for further information.
- Make a note of your new access code.
- Type the new access code, maximum 12 positions and press ENTER.
- Type the new access code again and press ENTER.

If the access code is not known to you, refer to authorized personnel or the service manual for further information

Bus Status

This display shows General Purpose Instrument Bus (GPIB: IEEE - 488.2) status. The display can give four different messages: REMOTE, LLO, ADDR, and SRQ.

- *Remote*

In this mode the generator is fully controlled from the bus. The instrument settings can't be changed from the front panel. To get control of the generator from the front panel you have to press LOCAL/RESET.
CAUTION: You may disturb a current GPIB program. When the REMOTE sign is off, the generator may be controlled from the front panel.

- *LLO*

In the LOCAL LOCK OUT mode, the LOCAL/RESET key is disabled. This is to safeguard the bus operation from unintentional interruptions from the front panel. It is necessary to switch the power OFF/ON to shift the control of the generator to the front panel.

- *ADDR*

This message means that the generator is addressed as listener or talker on the GPIB-Bus. See also chapter "GPIB operation" for further information about ADDRESS.

- *SRQ*

The SRQ message means that the generator asks for attention from the bus controller. The message can be caused by many reasons such as: a command error, execution error, query error, operation complete or output queue. See also chapter "GPIB operation" for further information about SERVICE REQUEST.

Local/Reset

This key is used to bring the control of the generator from remote to local. Any half-finished command will be excluded and the last correct setting will be recalled.

This page is intentionally left blank.

General

Keyboard

The keyboard is used to enter numerical parameters and to select alternatives in TRIGGER MODE and MENU operation.

Clear / Escape

Use the CLEAR / ESCAPE key to:
- clear erroneous keyboard entries, when in keyboard operation
- go one step back in the menu or to leave the TRIGGER MODE and the TRIG LEVEL menus

CLEAR/ESCAPE is also used as a second level function key. When it is pressed, the following functions are available:

The **PARAMETER** keys functions as NEXT/PREVIOUS keys to recall saved settings without using the menu procedure.

The **LOCK** key functions as unlock for all locked parameters.

The **VERNIER** keys can be used to set the 4th digit at pulse period and burst period.

EE / View

EE is the key for entering an exponent. First press EE, then the exponent. If the front panel format is set to UNIT, the exponent will be shown as ns, µs, ms, or s.

Enter

The red LED in the ENTER key has three modes:

OFF indicates that you are not supposed to use the key.

FLASHING indicates that you must press ENTER to confirm a value set with the keys.

ON indicates that you must make a choice from the MENU or select a TRIG MODE.

Parameter

The three PARAMETER keys are used to select which pulse parameter is to be set. They are also used to move the CURSOR when a selection is to be made in the TRIGGER MODE or MENU mode.

- The center key moves the cursor (up or down) to the other line. The center key is also used to change selection in the MENU mode.
- The right hand key moves the cursor to the right.
- The left hand key moves the cursor to the left.

Vernier

These keys are used to increase or decrease the value of any selected parameter. When you are satisfied with the indicated value, release the vernier.

FAST: Gives an increasing speed of the alteration. The alteration speed is proportional to the value, which gives a continuous accrleration on the screen of an oscilloscope.

SLOW: Gives constant speed of alteration for period and burst period.

The fourth digit of PULSE PERIOD (INT. TRIG) and BURST PERIOD may be set by pressing CLEAR/ESCAPE and a VERNIER key at the same time. (Not values within brackets).

If the value set with the VERNIER key passes any of the limitations for the generator, an error message will appear on the display and the pulse configuration on the output will remain on the latest valid combination set. The error message indicates the limit, and when the other VERNIER key (FAST) is touched, the value jumps back to the limit.

Error Messages

 # + Beeper

When a parameter value or function mode is changed, the # sign may appear on the display together with a beeping sound and a message saying that one value should be lower or higher than a certain limit. The message indicates the limit of the allowed range for the parameter at the cursor.

The limitation may be due to the specification of that particular parameter or because some other parameter values make the combination impossible. The error message may contain # signs marking the parameter which are involved in the error.

If the cursor is moved to another # marked parameter, the limits of that parameter will show.

A faulty value may be entered on the display but the parameter will not change at the outputs as long as the value is outside the limit. The pulse on the outputs will remain on the last valid setting to avoid impossible settings and unstable pulses.

When you change the pulse parameters to a new combination, you may experience one or several error messages during the setting procedure. These messages may be neglected during the setting. What counts is that the final combination is accepted by the generator.

Corrections:

An erroneous setting may be corrected in several ways.

Error Messages

• *Auto Set*	The AUTO SET function will make a correction of the other parameters which are limiting the selected parameter.
• *Vernier*	Position the cursor on the erroneous parameter, press the correct (up or down) "FAST" VERNIER key to set the value. The parameter value will jump back to within the range.
• *RESET*	When several parameters are faulty, press the RESET key and the last correct front panel setting is restored.

Messages: All error messages are explained in chapter 16.

Preventive Maintenance

Start -Up Sequence

Each time PM 5781 is switched ON, a starting up sequence is initiated. This sequence starts with a test of the logic and the front panel, which causes all displays and LEDs to flash.

The next step is that the generator gives the revision date for the main firmware, the GPIB firmware, and the selfcal firmware if installed.

```
PM 5781:  MAIN  V.  X.XX  day  month  year
          GPIB  V.  X.XX  day  month  year
```

After that, the GPIB address selected for this generator is displayed. To change this address, see Other Functions in chapter "Reference".

```
GPIB  Address :  XX
```

If any errors were detected during startup, a message with a diagnostic error code may eventually follow the GPIB address. The error code must be confirmed before the program continues. In such a case, refer to the chapter "Error Messages" or contact the Fluke service center for advice or repair of the generator. The PM 5781 Service Manual gives detailed information about the troubleshooting procedure.

After a successful test sequence the display will switch over to the warm-up message for 30 seconds:

Warm up in progress

30 seconds later the output voltages have stabilized, and a new message is given telling that the Initial warm-up is completed.

Now the generator will repeat the front panel setting used when it was switched off. The only difference is that the outputs are disabled.

After another 30 minutes, the generator has reached the normal operating temperature and the specified performance. This is indicated in the display with the following message message:

Warm up completed press any key to continue

Fans

The pulse generator has two fans for cooling. The air is taken in through the side panels and blows out through the rear panel in two air streams. This enables an easy check of the fans. Simply use your hand to check that there are two air streams through the rear panel. Check this every second month.

The two fans have a variable speed, which minimizes noise disturbance in normal room temperature and allows a shorter warm-up time. A temperature sensor inside the generator monitors temperature fluctuations and controls the speed of the fans. when the temperature increases, the speed of the fans increases also.

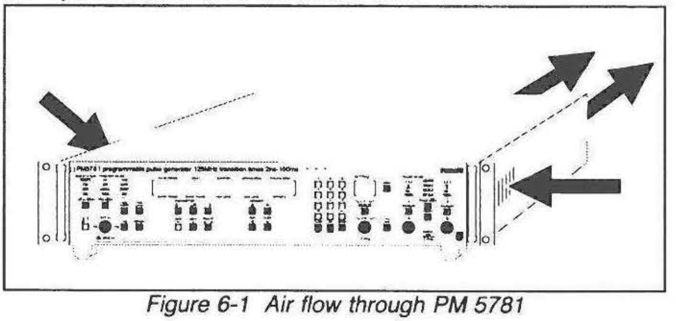


Figure 6-1 Air flow through PM 5781

To maintain a high reliability for the generator, the fans must be replaced every second year if the generator is used in a full-time application. For part-time applications and low ambient temperatures an extended service interval is acceptable. Replacing fans involves disassembly of the generator which must be done by a service technician. All information about this replacing the fans is included in the service manual. The date that the fans are replaced should be written on a label and attached to the rear panel of the generator.

Overheating

Overheating in the generator may originate from ambient temperatures that are too high or from obstacles in the cooling system. The generator contains temperature detection systems for protection against overheating.

If the logic senses a too high temperature it produces a warning in the display. This warning must be manually confirmed by the operator.

Warning over temperature Press any key to continue

When you see this warning on the display, **TURN THE GENERATOR OFF**. Prolonged overheating may result in a high and unnecessary repair cost.

When the generator has cooled down, power it up again and check the ventilation holes, the air streams, and the ambient temperature. To further examine the situation you may go into the SERVICE menu mode and read the temperature inside the generator via the INFORMATION menu. Refer to the chapter "Reference".

If the power supply of the generator is overheated, it will turn off and wait for the temperature to decrease. When the temperature has reached a safe level, the generator will restart and follow the normal start-up procedure.

Due to the different possible reasons for overheating, the detection systems may sense the temperature increase quickly or after some time has passed. Consequently, the power supply may turn off before the warning appears on the display. A generator that has been overheated should be checked with the "Performance Check" to verify that no calibration errors have occurred.

Battery

To preserve the variable data needed for the use of PM 5781 a lithium battery is included in the unit. The battery is estimated to have a lifetime of 5 to 10 years; however, we recommend changing the battery every 5 years to avoid disturbances to data.

Replacing the battery is takes less than one hour, but it should only be done by a service technician because it requires disassembly of the generator. The PM 5781 Service Manual describes this operation in detail.

If the battery is too weak before it is replaced the generator will lose the calibration data and will display the following message.

DIAGNOSTIC ERROR (S) Press any key to continue

Then you will get the message:

Calibration data lost

After replacing the battery the generator should be calibrated and adjusted according to the procedure in the PM 5781 Service Manual.

NOTE: The date of the battery replacement must always be noted on the label on the rear side of PM 5781 and in the bus operation program. See chapter Commands.

Calibration

To maintain the specified performance of PM 5781/011 and PM 5781/012 (without Internal Calibration function) it is recommended that a calibration and adjustment be done once a year. The procedure for calibration and adjustment is thoroughly described in the PM 5781 Service Manual.

A performance check of the generator can be made according to the information in chapter Performance Check.

Generators with the Internal Calibration function (PM 5781 /021 and PM 5781/022) are more easily calibrated and adjusted because the only procedure required is to activate the Internal Calibration function. Two minutes later the whole procedure is ready. For further details read "Other Functions" in chapter Reference.

This page is intentionally left blank.

Performance Check

Introduction

This chapter gives detailed information about how to verify the performance of PM 5781. This performance check may be used for incoming inspection of new units or for checking the performance of generators in use.

All limits and tolerances given in this section are calibration guide lines and should not be interpreted as instrument specifications unless they are also found in chapter "Specification". All tests are made from the front panel, and you do not have to remove the instrument covers.

Temperature

Allow the specified warm-up time of at least 30 minutes before starting the measurements. Then check the inside temperature of the generator.

To measure the temperature:

- Press the MENU key.
- Select 5 for AUXILIARY.
- Select 4 for UTILITY.
- Select 1 for INFO.

The display will now give the temperature in degrees centigrade, measured inside the cabinet. This temperature is compared with the temperature when the last calibration and adjustment of the generator was performed (Cal: XX°C).

If the actual temperature deviates more than about 5°C, the indication in the display will be HITEMP or LOTEMP. As long as the temperature is within the range of ±5°C the indication is CALIBR.

To run a performance check at a temperature that is too high or too low means that the values that are found to be outside the limits stated in the list must be compared with the tolerances given for the wider temperature range (0 to+50°C) in the specification. The maximum and minimum values stated in the lists refers to the range ±5°C from the calibration temperature.

Self Test

Run the self test program to check the logic and the front panel functions at the correct temperature (after the warm- up time). During the self test, observe the display for possible error messages. Error messages related to the front panel will not interrupt the sequence.

To initiate the test program:

- Press the MENU key.
- Select 5 for AUXILIARY.
- Select 4 for UTILITY.
- Select 3 for TEST.
- Select 1 for ALL.

If the self test is run without error messages, continue with the following performance check. If there is an error message, check with the list of error messages in chapter 16 for further information.

If the following front panel test or the measurements fail contact your nearest Fluke service center for advice about calibration and adjustment or possibly repair of your generator.

Performance Check

The performance check procedure is based on a menu program in the generator that allows you to go step by step through all the ranges for each parameter. This menu program is also needed for adjusting software calibration constants used for linearization of parameter ranges. The access code requested in this program for adjustments is not needed for the verify procedure.

If an adjustment of the generator is needed, please refer to the PM 5781 Service Manual for instructions and for the access code.

Test Equipment

Only a few test instruments are needed for the performance check. (Use the recommended type or an equivalent.) The test equipment must be calibrated before the performance check starts.

Instrument Type	Recommended model
Timer/counter	Philips PM 6681
Oscilloscope, 2GHz	Philips PM 3340
Oscilloscope, LF	Philips PM 3320
DVM	Fluke 8920A
Signal source	Philips PM 5786

Before you start the performance check, make a copy of the six pages in this chapter with measuring lists.

Display Test

- Press the MENU key.
- Select 5 for AUXILIARY.
- Select 2 for FRONT.
- Check that the front panel settings are:
  Display mode: ALL,
  Format: EXPonent
  Beeper: ON.

If any of the settings are wrong, use the three parameter keys to change them.

- Press ESC to return to the auxiliary menu.
- Select 4 for UTILITY.
- Select 3 for TEST.
- Select 4 for FRONT; check that all LEDs except TRIG'D are lit (23 pcs).
- Check to make sure the LED keys are lit: Neg SLOPE, 50 OHM, CLOCK OUT DISABLED, CLOCK OUT TTL/ECL, OUTPUT A, and OUTPUT B.
- Check all parts of the alphanumeric display.
- Check that all segments in the setting display works (the display shall indicate "88").
- The BEEPER should make a short beep during this test.
- If you are unable to see all the parts at the first test, repeat it by pressing number 4 for FRONT to test again.
- After the test, the display will indicate "Front panel test passed".
- Press ESCAPE until you return to the parameter menu.

Key Test

- *Local/Reset*

All activated display functions will go out and light up again after about 1 second.

- *Trig Mode*

Press the TRIG MODE key to activate the "Select trig source" menu. Press TRIG MODE again to move the cursor one step, Press ENTER and check to make sure you get the "Select trig function" menu. Press the TRIG MODE key again and then press the ENTER key. Check to make sure the new choice is updated in the display.

- *Neg Slope* — Press the NEG SLOPE key and the green LED in the key should go on/off.
- *50 Ohm* — The LED in the key should go on/off
- *Trig Level* — The central display should indicate "Trig level". Press the ESCAPE key to turn it off.
- *Manual* — Select "Manual Trig" by pressing the TRIG MODE key plus the ENTER key. Press the MANUAL key and check that the yellow LED indicator "TRIG'D" is lit when the MANUAL key is pressed.

  Check "Manual Gate" in the same way.
- *Parameter* — Move the cursor left and right. Keep the cursor key pressed to check scrolling; also check that it is possible to change from the upper to the lower line.
- *Lock* — Press the LOCK key and check that the sign ■ appears at the cursor position (not values within brackets). Press again and the sign should go out.
- *Track* — The TRACK key works in the same way as LOCK, but TRACK gives a sign α in the upper line and β in the lower line. (High and low level only)
- *Vernier* — Check that all four keys make the parameter value scroll. Check: up/down and fast/slow.
- *Keyboard* — Check that all keys 0 to 9 work. Check the decimal point "." key, the polarity sign +/– key, and the exponent EE key. Use the CLEAR key to empty the parameter field. Also check that the ENTER LED flashes.
- *Clock Out Disabled* — Check that the LED in the key goes on/off
- *Clock Out TTL/ECL* — Check that the LED in the key goes on/off, and that the CLOCK OUT DISABLED key lights up when the TTL/ECL key is pressed.

- *Pulse Mode*

Check that the green LED's in the keys above are changing and the yellow indicators are changed a second later. Press the TRIG MODE key to activate EXTERNAL DURATION.

- *Compl*

Press the COMPL key and check that the yellow COMPL indicators go on and off.

- *Output Disabled A/B*

Check that the LED in each key goes on and off.

Pulse Period

To check the divider chain for Pulse Period:

- Press the MENU key.
- Select 2 for RECALL.
- Position the cursor on Setting number 0.
- Press the ENTER key.
- Enable output A.
- Set the Pulse Period to 2.0 μs.
- Press the MENU key.
- Select 5 for AUXILIARY.
- Select 4 for UTILITY.
- Select 2 for PERIOD CONTROL.
- Set the period control to OFF.
- Press the ESCAPE key until you are out of the menu.

Connect a PM 6680 counter to output A. Set up the counter for "Period A" measurement with triggering on +1.0 V. Input impedance must be 50 Ω. Read the pulse period and make a note of the value on **a copy of page 7-9.**

Change the PERIOD setting of PM 5781 to **20.0 μs**

Measure and make a note of the pulse period time again. The measured value must be ten times higher than in the previous measurement. A difference of only a few units in the fourth digit is allowed.
Change the PERIOD setting of PM 5781 to **200 μs.**
Measure and make a note of the pulse period time again.

The measured value must be ten times higher than in the previous measurement. A difference of only a few units in the fourth digit is allowed.
In the same way as above set also the values: **2 ms**, **20 ms**, **200 ms**, and **2 s**. For each value, measure the result and write it down. Now compare the results. All the measurements must give the same numerical value in the first three decades.

Checking the Pulse Period:

- Press the MENU key.
- Select 5 for AUXILIARY.
- Select 4 for UTILITY.
- Select 4 for CALIBRATION.
- Select 2 for MANUAL CALIBRATION.
- Select 1 for Verify. The generator is now set up for pulse period measurement on range 1.
- Position the cursor on Lo. A parameter value is now visible in the upper right corner of the display. This is the period time that must be measured on the output in this first step.
- Make a note of the value on the line: Range 1, Lo.
- Compare with the min. and max values stated, and make a check mark in either the pass or fail columns.

NOTE: The measuring values must be within the stated minimum and maximum limits if the internal temperature of the generator is within the limits for calibration. If the temperature is outside the limits, a calculation of new limits must be done according to the specification.

- Move the cursor to the next position and repeat the procedure.
- Continue with the other three pulse period ranges and each DAC position.

Pulse Period Measurements

	Measured Value	Pass	Fail
2.0 µs			
20.0 µs			
200 µs			
2.0 ms			
20.0 ms			
200 ms			
2.0 s			

Pulse Period	Min.	Actual	Max.	Pass	Fail
Range 1					
Lo 8.0 E-9	7.8 E-9		8.3 E-9		
ML 9.8 E-9	9.0 E-9		10.6 E-9		
Mid 12.6 E-9	11.7 E-9		13.5 E-9		
Hi 20.0 E-9	18.9 E-9		21.1 E-9		
Range 2					
LoDac 20.0 E-9	18.9 E-9		21.1 E-9		
MidDac 45.0 E-9	43.1 E-9		46.8 E-9		
HiDac 100 E-9	96.5 E-9		103.5 E-9		
Range 3					
LoDac 100 E-9	97.0 E-9		103.5 E-9		
MidDac 320 E-9	310 E-9		330 E-9		
HiDac 1.00 E-6	0.97 E-9		1.03 E-6		

Waveform

Disconnect the counter and connect the PM 3340 oscilloscope to both A and B channels of the generator outputs; use the clock out as trigger source. Press the AUTO SET key on the oscilloscope, and check the waveform on the A and B output pulses.

- Position the cursor of PM 5781 on LoDAC (100 ns).

Check the waveform: Preshoot, overshoot, and ringing must be less than ± 5 % of pulse amplitude ± 10 mV.

Delay

The delay function is checked in the same way as the pulse period. After the last range in the pulse period, position the cursor in the upper left corner and use the center parameter key to change to delay. You must also change to range number 1.

The oscilloscope is used to measure the delay time on the A output, from the leading edge of the first pulse to the leading edge of the second pulse. Measure the delay time at 50 % of the pulse amplitude.

Measure the PM 5781 with the oscilloscope on the A output and use the clock output of the PM 5781 to trigger the oscilloscope. Use the AUTO SET function in PM 3340 to set the front panel controls.

From range 5, starting with 10 µs, use the PM 6680 counter instead of the oscilloscope.

Set the PM 5781 to pulse mode: SINGLE.

Set the counter for time interval A-B measurements.

Set trig level A to 1.25 V (from clock out), and set the trig level B to 0 V (from Output A), both 50 Ω, DC.

Set the MATH on L=(clock out - pulse out A) + X ns = 8 ns. This will compensate for the internal fixed delay of 8 ns in the generator.

Delay	Min.	Actual	Max.	Pass	Fail
Range 1					
Lo 8.0 E-9	7.5 E-9		9.0 E-9		
ML 11.0 E-9	9.7 E-9		12.3 E-9		
Mid 14.0 E-9	12.6 E-9		15.4 E-9		
Hi 20.0 E-9	18.4 E-9		21.6 E-9		
Range 2					
LoDac 20.0 E-9	18.4 E-9		21.6 E-9		
MidDac 47.0 E-9	44.6 E-9		49.4 E-9		
HiDac 110 E-9	106 E-9		114 E-9		
Range 3					
LoDac 110 E-9	106 E-9		114 E-9		
MidDac 340 E-9	329 E-9		351 E-9		
HiDac 1.00 E-6	0.97 E-6		1.03 E-6		
Range 4					
LoDac 1.00 E-6	0.97 E-6		1.03 E-6		
MidDac 3.20 E-6	3.10 E-6		3.30 E-6		
HiDac 10.0 E-6	9.70 E-6		10.3 E-6		
Range 5					
LoDac 10.0 E-6	9.70 E-6		10.3 E-6		
MidDac 32.0 E-6	31.0 E-6		33.0 E-6		
HiDac 100 E-6	97.0 E-6		103 E-6		
Range 6					
LoDac 100 E-6	97.0 E-6		103 E-6		
MidDac 320 E-6	310 E-6		330 E-6		
HiDac 1.00 E-3	0.97 E-3		1.03 E-3		
Range 7					
LoDac 1.00 E-3	0.97 E-3		1.03 E-3		
MidDac 3.20 E-3	3.10 E-3		3.30 E-3		
HiDac 10.0 E-3	9.70 E-3		10.3 E-3		
Range 8					
LoDac 10.0 E-3	9.70 E-3		10.3 E-3		
MidDac 32.0 E-3	31.0 E-3		33.0 E-3		
HiDac 100 E-3	97.0 E-3		103 E-3		

Duration

The duration measurements follow the delay measurements in the menu program. Change the parameter and range in the same way as discussed earlier under "Delay" (2 pages back). You should start in the first range of duration.

The measuring instrument to be used here is the PM 3340. (Connect it as explained earlier under "Delay" and use the AUTO SET function.

Now measure the pulse duration at 50 % of the pulse amplitude.

From the third range you have to use the PM 6680 counter again. Set the counter for "Pulse width A averaged". Check the remaining duration ranges.

Duration	Min.	Actual	Max.	Pass	Fail
Range 1					
Lo 4.0E-9	3.6 E-9		4.4 E-9		
ML 6.5 E-9	5.3 E-9		7.7 E-9		
Mid 9.00 E-9	7.7 E-9		10.3 E-9		
Hi 15.0 E-9	13.5 E-9		16.4 E-9		
Range 2					
LoDac 15.0 E-9	13.5 E-9		16.4 E-9		
MidDac 41.0 E-9	38.8 E-9		43.2 E-9		
HiDac 110 E-9	106 E-9		114 E-9		
Range 3					
LoDac 110 E-9	106 E-9		114 E-9		
MidDac 340 E-9	329 E-9		351 E-9		
HiDac 1.00 E-6	0.97 E-6		1.03 E-6		
Range 4					
LoDac 1.00 E-6	0.97 E-6		1.03 E-6		
MidDac 3.20 E-6	3.10 E-6		3.30 E-6		
HiDac 10.0 E-6	9.70 E-6		10.3 E-6		
Range 5					
LoDac 10.0 E-6	9.70 E-6		10.3 E-6		
MidDac 32.0 E-6	31.0 E-6		33.0 E-6		
HiDac 100 E-6	97.0 E-6		103 E-6		
Range 6					
LoDac 100 E-6	97.0 E-6		103 E-6		
MidDac 320 E-6	310 E-6		330 E-6		
HiDac 1.00 E-3	0.97 E-3		1.03 E-3		
Range 7					
LoDac 1.00 E-3	0.97 E-3		1.03 E-3		
MidDac 3.20 E-3	3.10 E-3		3.30 E-3		
HiDac 10.0 E-3	9.70 E-3		10.3 E-3		
Range 8					
LoDac 10.0 E-3	9.70 E-3		10.3 E-3		
MidDac 32.0 E-3	31.0 E-3		33.0 E-3		
HiDac 100 E-3	97.0 E-3		103 E-3		

Leading Edge

Disconnect the counter and connect the PM 3340 oscilloscope. Use the clock out as the trigger source, and press the AUTO SET key on the oscilloscope.

The transition time of the leading edge will be measured, and you must carefully define the top and baseline to be able to calculate the 10 and 90 % levels of the pulse.

From the fourth range on (1.1 µs) you must change to the PM 6680 counter again. Set the counter for "Rise/Fall Time A averaged" measurements, and set the trigger slope for **positive edges.** For the last ranges you must change to the PM 3340 oscilloscope again.

Continue through the remaining ranges of the leading edge.

Leading Edge	**Min.**	**Actual**	**Max.**	**Pass**	**Fail**
Range 1					
Lo 2.0 E-9	1.8 E-9		2.2 E-9		
ML 5.00 E-9	3.7 E-9		6.2 E-9		
Mid 8.00 E-9	6.6 E-9		9.4 E-9		
Hi 18.0 E-9	16.1 E-9		19.9 E-9		
Range 2					
LoDac 11.0 E-9	9.5 E-9		12.5 E-9		
MidDac 45.0 E-9	41.7 E-9		48.2 E-9		
HiDac 180 E-9	170 E-9		190 E-9		
Range 3					
LoDac 110 E-9	104 E-9		116 E-9		
MidDac 450 E-9	427 E-9		473 E-9		
HiDac 1.80 E-6	1.71 E-6		1.89 E-6		
Range 4					
LoDac 1.10 E-6	1.04 E-6		1.16 E-6		
MidDac 4.50 E-6	4.27 E-6		4.73 E-6		
HiDac 18.0 E-6	17.1 E-6		18.9 E-6		
Range 5					
LoDac 11.0 E-6	10.4 E-6		11.6 E-6		
MidDac 45.0 E-6	42.7 E-6		47.3 E-6		
HiDac 180 E-6	171 E-6		189 E-6		
Range 6					
LoDac 110 E-6	104 E-6		116 E-6		
MidDac 450 E-6	427 E-6		473 E-6		
HiDac 1.80 E-3	1.71 E-3		1.89 E-3		
Range 7					
LoDac 1.10 E-3	1.04 E-3		1.16 E-3		
MidDac 4.50 E-3	4.27 E-3		4.73 E-3		
HiDac 18.0 E-3	17.1 E-3		18.9 E-3		
Range 8					
LoDac 11.0 E-3	10.4 E-3		11.6 E-3		
MidDac 45.0 E-3	42.7 E-3		47.3 E-3		
HiDac 90.0 E-3	85.5 E-3		94.5 E-3		

Trailing Edge

When you are finished with all the eight ranges of leading edge, the menu program continues with the measurements of trailing edge.

Here the PM 3340 oscilloscope must be used again. Connect it as earlier and use the AUTO SET function to get a picture on the screen.

The transition time of the trailing edge will be measured, and you must carefully define the top and baseline to be able to calculate the 10 and 90 % levels of the pulse.

From the fourth range on (1.1 µs) you have to change to the PM 6680 counter again. Set the counter for "Rise/Fall Time A averaged" measurements, and set the trigger slope for **negative edges.** For the last ranges you have to change back to the PM 3340 oscilloscope again.

Continue through all the remaining ranges of the trailing edge.

Trailing Edge	Min.	Actual	Max.	Pass	Fail
Range 1					
Lo 2.0 E-9	1.8 E-9		2.2 E-9		
ML 5.00 E-9	3.7 E-9		6.2 E-9		
Mid 8.00 E-9	6.6 E-9		9.4 E-9		
Hi 18.0 E-9	16.1 E-9		19.9 E-9		
Range 2					
LoDac 11.0 E-9	9.5 E-9		12.5 E-9		
MidDac 45.0 E-9	41.7 E-9		48.2 E-9		
HiDac 180 E-9	170 E-9		190 E-9		
Range 3					
LoDac 110 E-9	104 E-9		116 E-9		
MidDac 450 E-9	427 E-9		473 E-9		
HiDac 1.80 E-6	1.71 E-6		1.89 E-6		
Range 4					
LoDac 1.10 E-6	1.04 E-6		1.16 E-6		
MidDac 4.50 E-6	4.27 E-6		4.73 E-6		
HiDac 18.0 E-6	17.1 E-6		18.9 E-6		
Range 5					
LoDac 11.0 E-6	10.4 E-6		11.6 E-6		
MidDac 45.0 E-6	42.7 E-6		47.3 E-6		
HiDac 180 E-6	171 E-6		189 E-6		
Range 6					
LoDac 110 E-6	104 E-6		116 E-6		
MidDac 450 E-6	427 E-6		473 E-6		
HiDac 1.80 E-3	1.71 E-3		1.89 E-3		
Range 7					
LoDac 1.10 E-3	1.04 E-3		1.16 E-3		
MidDac 4.50 E-3	4.27 E-3		4.73 E-3		
HiDac 18.0 E-3	17.1 E-3		18.9 E-3		
Range 8					
LoDac 11.0 E-3	10.4 E-3		11.6 E-3		
MidDac 45.0 E-3	42.7 E-3		47.3 E-3		
HiDac 90.0 E-3	85.5 E-3		94.5 E-3		

Amplitude Parameters

The next group of ranges in the menu program is the amplitude parameters. Change to the PM 3320 oscilloscope. Connect it to output A and use the clock out for triggering. The output pulse is a square wave with a period time of 10 ms.

Check the pulse amplitude in normal (NRM) and complementary (CMP) mode according to the program.

The next function is the offset possibilities for the A channel, which is tested with minimum amplitude.

Now you must move the oscilloscope connection from the A output to the B output channel to check the offset and the attenuation on this B output.

You are now finished with the menu-guided part of the performance test.

Press CLEAR/ESCAPE until you are out of the menu program.

Change the connection of the oscilloscope. You will now measure the levels of the clock pulse. The generator will now be set to give a symmetrical pulse with 10 ms period time.

Change the TTL/ECL switch setting and also measure the additional two levels.

Amplitude Parameters	Min.	Actual	Max.	Pass	Fail
Amplitude Mode:NRM					
HiDac +3.170	3.08		3.26		
MidDac +6.580	6.38		6.78		
LoDac +10.00	9.70		10.3		
Amplitude Mode:CMP					
HiDac +3.170	3.08		3.26		
MidDac +6.580	6.38		6.78		
LoDac +10.00	9.70		10.3		
Offset A					
HiDac –9.900	–9.76		–10.04		
MidDac 0.000	–0.04		+ 0.04		
LoDac +9.900	+9.76		+10.04		
Offset B					
HiDac –9.900	–9.76		–10.04		
MidDac 0.000	–0.04		+0.04		
LoDac + 9.900	+9.76		+10.04		
Attenuation					
-10dB +3.16	3.03		3.29		
-20dB +1.000	0.93		1.07		
-30dB +0.316	0.27		0.365		
Clock out					
ECL-high –0.8V	–0.7 V		–0.9 V		
ECL-low –1.7 V	–1.8 V		–1.9 V		
TTL-high +2.5 V	+2.3 V		+2.7 V		
TTL-low 0 V	–0.2 V		+0.2 V		

Burst Sync Out

Set the generator for internal burst.

- Press the TRIGGER MODE key.
- Select INTERNAL.
- Select BURST.
- Press the PULSE MODE key and select SQUARE WAVE.
- Set pulse period to 100 ns.
- Set burst period to 1 μs.
- Set burst pulses to 4.

Connect the PM 3320 oscilloscope to the A output and to the BURST SYNC OUT on the rear panel. Check that the BURST SYNC OUT pulse goes positive at the beginning of the A output pulse burst and returns at the end of the burst. See fig 7-1

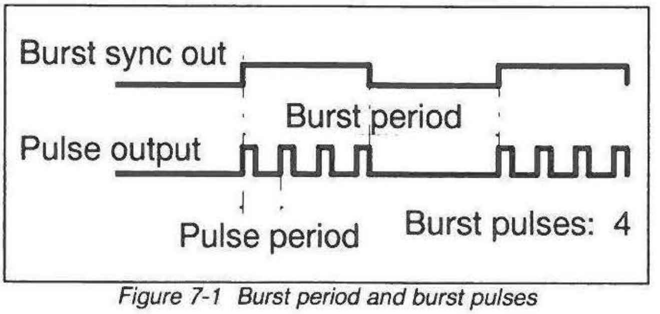


Figure 7-1 Burst period and burst pulses

Also check the levels of the Burst Sync Out pulse. High level will be –0.4 V and low level –0.8 V.

Burst Pulses

- Set the PULSE PERIOD to 8 ns.
- Set the DELAY to 8 ns.
- Set the DURATION to 4 ns.

- Set the BURST PULSES to 0000.
- Set CLOCK OUT to TTL level, and enable the output.
- Set the pulse mode to square.

Change the number of pulses from 0 to 10, and use the PM 3340 to check that the number of output pulses is correct. Check that the Burst sync out pulse remains "high" slightly longer than the burst pulses.

- Press the TRIG MODE key.
- Select MANUAL.
- Select BURST.
- Set PULSE PERIOD to 100 ns.
- Connect PM 6680 to the clock output of PM 5781 and set it for "TOT A/MAN" measurement
- Set the number of pulses to 1111, 2222, and so on, up to 9999.
- Check that a correct number of pulses are produced.

- Press the TRIG MODE key.
- Select INTERNAL BURST.
- Set BURST PULSES to 1.

Check that 1 pulse is produced every 8 μs.

Burst Period

- Set the Burst Period to 10 s.

Measure the period time with the PM 6680 counter connected to output A. A difference of max. ± 20 ppm is allowed; this means a minimum of 9.99980 s and a maximum of 10.00020 s.

External Input

Connect a signal, with a high level of +0.5 V, and a low level of –0.5 V, with a duty factor of 10 %, and at 100 kHz to the External Input.

- Press TRIG MODE.
- Select EXTERNAL TRIG.
- Set the trig level to –5.0 V.
- Check that the TRIG'D indicator is ON.

Use the "FAST VERNIER" key to increase the value up to –0.6 V, and check with the oscilloscope that there is no signal on CLOCK OUT, the indicator should remain ON.

Use the VERNIER key to increase the value up to –0.4 V and check that there is a signal now. The indicator should be flashing.

Use the VERNIER key to increase the value up to +0.4 V and check that there is output signal. The indicator should still be flashing.

Continue with the VERNIER key up to +0.6 V. There should be no output signal, and the indicator should be OFF.

Continue with the VERNIER key to +5 V and check that there is no output signal. The indicator should remain OFF.

- *125 MHz*
  - Set the TRIG LEVEL to 0 V.
  - Change the signal source to 125 MHz, symmetrical signal of 500 mV_{pp}.
  - Check with the PM 6680 counter that there is an output signal of 125 MHz.

- *Gate and Slope*
  - Press TRIG MODE.
  - Select EXTERNAL GATE

Use the LF oscilloscope to check that there are output pulses during the positive part of the external input signal.

- Select NEGATIVE SLOPE.

Check that there are output pulses during the negative part of the input signal.

- *Burst*
  - Press the TRIG MODE key.
  - Select EXTERNAL BURST.

Check that the burst starts after the negative edge of the external input pulse.

- *Duration*
  - Press the TRIG MODE key.
  - Select EXTERNAL DURATION.
  - Select POSITIVE SLOPE .

Check that there is the same duty factor on the output of PM 5781 as on the external input signal.

You are now finished with the performance check, and there must be no check marks in the Fail column.

This page is intentionally left blank.

Specifications

Specifications apply with all outputs terminated with 50 Ω load resistance.

Inaccuracy and repeatability error specifications

Temperature Range	20 - 30°C 1)	0 - 50°C	After internal calibration 2) Typical Values	Repeatability Error Typical Values
Period 3):	± 3% ± 0.5ns	± 5% ± 0.5ns	± 1% ± 0.5ns	± 0.5ns
Period 4):	± 0.1%	± 0.1%	± 0.1%	± 0.1%
Delay:	± 3% ± 1ns	± 5% ± 1ns	± 1% ± 1ns	± 0.5%
Duration:	± 3% ± 1ns	± 5% ± 1ns	± 1% ± 1ns	± 0.5%
Transition Times:	± 5% ± 1ns	± 8% ± 1ns	± 3% ± 1ns	± 1%
High and Low Levels:	± 1% of level ± 2.5% of amplitude ± 40 mV	± 1% of level ± 3% of amplitude ± 60 mV	± 1% of level ± 2% of amplitude ± 20 mV	± 0.5% of level ± 0.5% of amplitude ± 20 mV
Burst Repetition Period	± 20 ppm	± 20 ppm	± 20 ppm	± 20 ppm

1)
Manual calibration is possible for another operating temperature range through a menu-and-cursor driven procedure, accessible via the front panel. The 20°C to 30°C specifications apply within ± 5°C from the actual calibration temperature.

2)
The enhanced accuracy version of PM 5781 features built-in capabilities for internal calibration. The specified inaccuracies apply within ± 5°C from the actual calibration temperature for 24 hours.

3)
After the 5th pulse in each burst in Gate and Burst Modes.

4)
In Internal Trigger Mode with Period Control on.

Time Parameters

Specified at 50 % of pulse amplitude at minimum transition times.

Pulse Repetition Period

Range: 8 ns...10 s *)

Programming Resolution:
Internal trigger mode; 4 digits
Gate and burst modes; 3 digits

Jitter$_{RMS}$:
0.1 % + 50 ps for period times <1 µs.
0.03 % for period times >1 µs
*) In internal burst mode, with only 1 pulse per burst, the period range is extended up to 4000 s

Pulse Delay

Range: 8 ns...100 ms

Programming Resolution: 3 digits

Jitter$_{RMS}$: 0.1 % + 50 ps

Pulse Duration

Range: 4 ns...100 ms.

Programming Resolution: 3 digits

Jitter$_{RMS}$: 0.1 % + 50 ps

Duty Factor: Up to 50 % in normal mode. Approaching 100 % in complementary mode

Constant Duty Factor

Repetition period and pulse duration trakked.

Range: 0 %...50 % in normal mode. 50 %...100 % in complementary mode

Preset Burst

Range: 1...9999 pulses in each burst

Internal Burst Repetition Period:
1 µs...4000 s

Programming Resolution:
4 digits, min 1 µs

External Burst Repetition Period:
> 250 ns

Transition Times

(Between 10% and 90% of amplitude)

Range: 2 ns...100 ms

Minimum transition time, between 20 % and 80 % of amplitude, is 1.4 ns.

Leading and trailing edge transition times are independently programmable within each of the following ranges:

2 ns...20 ns   10 ns...200 ns   100 ns... 2 µs
1 µs...20 µs   10 µs...200 µs   100 µs ... 2 ms

1 ms...20 ms   10 ms...100 ms

Programming Resolution: 3 digits

System Delays

Main Pulse Outputs
Fixed trigger input to main output delays.
External Trigger Mode: Approx. 35 ns
External Gate Mode: Approx. 45 ns
External Burst Mode: Approx. 60 ns
External Duration Mode: Approx. 35 ns
Applies in double pulse and square wave modes. Add programmed pulse delay in single pulse mode.

Clock Output
Pulse Advance:
Approx. 8 ns ahead of main outputs in double pulse and square wave modes.
Approx. 8 ns plus programmed pulse delay ahead of main outputs in single pulse mode.

Outputs

Main Pulse Outputs

Outputs:
2 channels, A (normal) and B (complementary)

Pulse Modes:
Single pulse, double pulse and square wave

Output Mode:
Normal/complementary; to make A complementary and B normal

High Level Range: – 9.80 V...+ 10.0 V

Low Level Range: – 10.0 V...+ 9.80 V

Programming Resolutions: 10 mV

Pulse Amplitude:
0.2 V...10 V, being the difference between the high and low levels

Offset Channel B: within ±10 V

Preshoot, Overshoot, Ringing:
± 5 % of pulse amplitude ± 10 mV

Source Impedance:
50 Ω, reflection coefficient < 0.1

Output Disable:
Output signal connect / disconnect

Level Limit:
High and low levels can be limited to preprogrammed values

Output Protection:
Against short and open circuit

Linearity:
≤ ± 10% for transitions < 20 ns
≤ ± 5% for transitions ≥ 20 ns

Clock Output

Pulse Levels into 50 Ω:
TTL;
low 0 V, high 2.5 V (doubles when driving into open circuit)
ECL; high – 0.8 V, low – 1.7 V

Transition Time: 1.5 ns typically

Source Impedance: 50 Ω

Output Protection:
Against short and open circuit

Output Disable:
Clock signal connect/disconnect

Burst Sync Output

Approx. 0.4 V pulse amplitude into 50 Ω from 50 Ω source during the pulse burst duration

Levels: high level - 0.4 V, low level- 0.8 V

Input

Frequency Range: DC...125 MHz

Minimum Pulse Duration: 4 ns

Sensitivity: 500 mV_{pp}

Dynamic Input Voltage Range:
10 V_{pp} within ± 10 V

Trigger Level Range: - 5.00 V...+ 5.00 V

Programming Resolution: 10 mV

Trigger Slope: positive or negative

Input Impedance:
1 MΩ//45 pF or 50 Ω

Max Input Voltage Without Damage:
10 V_{RMS}, 50 V_{RMS} at 1 MΩ and at < 10 MHz

Trigger Modes

Internal Triggering:
Continuously repetitive waveform is generated

Internal Gate:
Internal burst generator enables the period generator synchronously, burst duty factor 50 %

Internal Burst:
Internal burst generator enables the period generator to provide a preprogrammed number of pulses (1...9999)

External Triggering:
Each input pulse generates a single output pulse

External Gate:
External signal enables internal period generator synchronously

External Burst:
Each input pulse generates a pre-programmed number of pulses (1...9999)

External Duration:
External signal generates an output signal with same duration and repetition rate and selectable transition times and output levels

Manual:
Simulates an external input signal

Features

Front Panel Displays

Alphanumeric Display:
A 2 lines x 40 characters alphanumeric LCD display with backlighting. The display shows all 10 pulse parameters simultaneously. The display is also used for various function selection and error messages

Bus Status Display:
An LED display shows bus states; REMOTE, LLO, ADDR (listen / talk) and SRQ

Trigger Mode Display:
An LED display shows trigger mode

Setting Number Display:
An LCD display shows setting number

Pulse Mode Display:
An LED display shows main pulse output mode

Nonvolatile Memory

Internal battery-backed RAM is used to save settings. Current setting is saved on power down

Power On:
Last setting is activated when the power is switched on (output disabled)

Settings:
49 complete instrument settings can be saved and recalled

Macros:
Up to 20 macros can be defined and saved

Error Handling

Error Detection:
Erroneous settings are detected and error messages are displayed

Autoset:
Erroneous settings are easily corrected with the AUTOSET key

Track

A change of one pulse parameter value can automatically cause other parameters to follow proportionally. E.g. tracking of pulse period and duration results in a pulse with constant duty factor

Lock

Unintentional changes of numerical parameter values can be prevented by the lokking facility

Enhanced Accuracy Version PM 5781/02

The enhaced accuracy version includes circuitry that can measure pulse parameters. The internal frequency- and voltage references, that are part of this measuring system, can be made accessible via the front panel to enable external traceability of the internal calibration function.

Parameters:
Period, delay, duration, leading and trailing edges, high and low levels, and offset B.

Time for Self-Calibration:
Approx. 3 min.

Inaccuracy:
See table on page 8-2.

IEEE-488 Capabilities

All functions and parameters are programmable. The interface complies with the requirements of the IEEE-488.1 - 1987 and IEEE-488.2 - 1987 Standards

Interface Functions:
SH1, AH1, T6, L4, SR1, RL1, PP0, DC1, DT1, C0, E2

Address Range:
0...30, set via front panel entry

Address Time: 17μs

Times for bus transfer of settings:
Complete Parameter Set:
3 ms (binary format)
Single Command: 600 μs

Times to implement settings (ErrorCheck and DispMode Off):
Complete Parameter Set:
50 ms (binary format)
Single Command: 30 ms

***RCL:** 15 ms

Acknowledgement of complete execution is possible with *OPC? command or service request signal.

GET Time:
360 ns (DAV to H.W. line trigger)

GET with Aaddressing: 19 μs

General Specifications

Environmental Conditions

Meets environmental requirements of MIL-T-28800D Type III, Class 5 as follows.

Calibration Interval: 1 year recommended, 2 years recommended for version with built-in calibration facility

Warm-Up Time:
30 min to reach specified performance

Ambient Temperature:
Operating; 0°C...+50 °C.
Nonoperating; -20°C...+70°C

Altitude: <4600 m (15000 ft)

Humidity: <95% RH, 0°C...30°C

Vibration: Operating; 15 min along each of the 3 major axis, 10 Hz...150 Hz, 0.33 mm_{pp} amplitude and 2 g max. acceleration

Shock: Operating; 3 shocks along each of the 3 major axis, half-sinewave, 6 ms duration and 40 g acceleration pulses

Bench Handling: MIL-STD-810D method 516.3, procedure VI

Safety: CSA Bulletin 556B certified and Philips certification to comply with IEC 348, Class 1 requirements

Power Requirements

100, 120, 220, and 240 V ± 10 %, 48 Hz...420 Hz, 200 VA

Dimensions and Weight

Width: 440 mm, (17.3 in), excluding 19" rack mount brackets

Height: 89 mm, (3.5 in)

Depth: 500 mm, (19.7 in)

Weight: Net 12 kg (26.5 lbs), Shipping 15.5 kg (33 lbs)

Ordering Information

- *Models*

PM 5781/011 Pulse Generator with Front Panel Connectors

PM 5781/012 Pulse Generator with Rear Panel Connectors

PM 5781/021 Pulse Generator with Front Panel Connectors and built in calibration facility

PM 5781/022 Pulse Generator with Rear Panel Connectors and built in calibration facility

- *Included Accessories*

1 Power Cord
1 Bench-Top Conversion Kit
1 Operators' Manual
3 IEEE 488 Pocket Guides

- *Optional Accessories*

PM 9613: Rack-mount Slide Kit

PM 9581:
50 Ω Feed-through Termination; 3 W

PM 9584:
50 Ω T-Piece (matched power splitter)

PM 9585:
50 Ω Feed-through Termination; 1 W

PM 9588: 50 Ω Coaxial Cable Set (5x 1 ns, 4x2 ns, 3x3 ns, 3x10 ns)

- *Manuals*

4822 872 20002 Operators Manual (English)
4822 872 25014 Service Manual (English)
4822 872 20003 Pocket guide (English)

GPIB section

Introduction

This section is divided into six chapters aimed at different levels of reader knowledge. *Interface Functions* in chapter 10 and *System Functions* in chapter 11 can be disregarded by the experienced user.

Programming the PM 5781 in chapter 12 contains hints on how to get going with this particular instrument, how to make programs execute fast etc.

Chapter 13, *Command reference* is the 'Programmers Reference' section of this manual with complete information on all commands.

Chapter 14 contains an alphabetical subset of chapter 12. Chapter 14 contains only: Shortform description of function and Command syntax.

Programming examples, in chapter 15, contains examples of typical programs for a wide variety of applications. These programs are written in GW-basic. They can be used as they are on an IBM PC compatible controller with a PM 2201 interface board.

Conventions

Syntax Specification Form

This manual uses the EBNF* notation for describing syntax. This notation uses the following types of symbols:

- *Printable characters:*

  Printable characters like Command headers etc. are printed just as they are, e.g. Period means that you should type PERIOD.

- *Non printable characters:*

  Two non-printable characters are used: ␣ indicates the space character (ASCII code 32) and ↵ indicates the new line character (ASCII code 10).

- *Specified expressions: < >*

  Symbols and expressions that are further specified elsewhere in this manual are placed between the <> signs. E.g. <Dec. data.>. The following explanation is found on the same page: "Where <Dec. data> is a four digit number between 0.1 and $8 \bullet 10^{-9}$."

- *Alternative expressions giving the same result: /*

  Alternative expressions giving the same result are separated by a forward slash /, e.g. Duration/Dur/Width means that you can use either Duration or Dur or Width to select the same function.

---

* Extended Backus-Naur Form

- *Alternative expressions giving different result:* | — Alternative expressions giving different results are separated by |. E.g. On|Off means that the function may be switched on or off.

- *Grouping: ( )* — Example: TRIGSLOPE (POS|NEG) specifies the command header TRIGSLOPE followed by a space character and either POS or NEG.

- *Optionality: [ ]* — An expression placed within [ ] is optional.

  Example:SETINCREMENT [Parameter]

  means that command SETINCREMENT may or may not be followed by a space character and a parameter.

- *Repetition: { }* — An expression placed within { } can be repeated zero or more times.
  Example: DELMEM <No.> {,<No.>} specifies a sequence of one or more numbers after the DELMEM command.

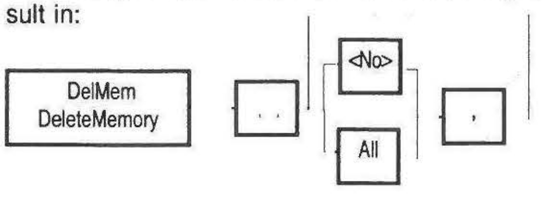
- *Equality: =* — Equality is specified with = Example: <Separator> = ,

Syntax example: The procedure of deleting one, several or all front panel settings from the memory is described as:

DeleteMemory/DelMem { (<No.> |All) , }

Where <No.> is a number between 1 and 49.

Describing the same function as a **Flow Diagram** will result in:



Interface Functions

What Can I Do with the Bus?

All the capabilities of the interface for the PM 5781 are explained below. For a complete description of all GPIB-interface functions, read the "Philips Instrumentation-Systems Reference-Manual", ordering No. 9499 997 00411.

Summary

Description	Code
Source handshake	SH1
Acceptor handshake	AH1
Control function	C0
Talker Function	T6
Listener function	L4
Service request	SR1
Remote/local function	RL1
Parallel poll	PP0
Device clear function	DC1
Device trigger function	DT1
Bus drivers	E2
Source and Acceptor Handshake	SH1, AH1

SH1 and AH1

Means that the pulse generator can exchange data with other instruments or a controller, using the bus handshake lines; DAV, NRFD, NADC.

Control Function, C0

The pulse generator does not function as a controller.

Talker Function, T6

The pulse generator can send responses and the results of its measurements to other devices or to the controller. T6 means that the pulse generator has the following functions:

- Basic talker.
- No talk only.
- It can send out a status byte as response to a serial poll from the controller.
- Automatic un-addressing as a talker when it is addressed as a listener.

Listener Function, L4

The pulse generator can receive programming instructions from the controller. L4 means that the generator has the following functions:

- Basic listener.
- No listen only.
- Automatic un-addressing as listener when addressed as a talker.

Service Request, SR1

The pulse generator can call for attention from the controller, e.g. when a measurement is completed and a result is available.

Remote/Local, RL1

You can control the pulse generator manually (locally) from the front panel, or remotely from the controller. The LLO, local-lock-out function, can disable the LOCAL button on the front panel.

Parallel Poll, PP0 The pulse generator does not have any parallel poll facility.

Device Clear, DC1 The controller can reset the pulse generator via interface message DCL (Device clear) or SDC (Selective Device Clear).

Device Trigger, DT1 You can trigger the generator, as specified by the *DDT command, from the controller via interface message GET (Group Execute Trigger).

Bus Drivers, E2 The GPIB interface has tri-state bus drivers.

System Functions

Introduction

About this Chapter

This chapter describes the operation of instruments that follow the IEEE–488.2 standard. This chapter also describes the operation of Fluke system–compatible instruments. This chapter is NOT a complete description of IEEE–488.2, but it is an introduction for the user of the IEEE–488.2–compatible instrument PM 5781 in GPIB–controlled environments.

Relationship IEEE–488.1 to IEEE–488.2

The IEEE–488.2 standard does not replace IEEE–488.1 but is an addition to it. Instruments compatible with IEEE–488.2 must also be compatible with IEEE–488.1, but IEEE–488.1 compatible instruments may not conform to the IEEE–488.2 standard.

- *Identification of the Main Subjects*

The IEEE–488.2 standard specifies the higher interface layers within instrumentation systems using GPIB interfaces.

IEEE–488.1 Functions

IEEE–488.2 specifies more precise rules in the application of an IEEE–488.1 interface. IEEE–488.2 restricts the number of alternatives allowed when designing IEEE–488.1 interfaces. For instance, IEEE–488.2 instruments must have a Remote Local function with a "return-to-local" key (complete capability), where IEEE–488.1 must not.

IEEE 488.2 adds new requirements to several IEEE–488.1 interface functions, e.g., the IEEE–488.1 source handshake protocol does not prevent the talker from sending data when there are no listeners on the bus whereas 488.2 Source Handshaking prevents data bytes from being lost.

Message Exchange Control Protocol

This protocol specifies the message exchange between the controller and device. The protocol specifies, for instance, exactly how a device shall handle program and response messages it receives from and sends to a controller. The protocol uses queries and commands. Queries are program messages which order the device to send a response. Commands will not cause the device to generate a response.

Unlike older instruments it is, therefore, always necessary to send a query to get an answer from the instrument. The previous practice of just doing a "read" when a result was wanted does not work with the new protocol.

The message exchange control protocol specifies the order of execution of program messages. It defines several categories of errors exactly, such as command errors, query errors, execution errors, and device specific errors. The protocol demands that the device report any violation of the IEEE–488.2 rules to the controller, even when it is the controller that breaks these rules.

Syntax Definitions

IEEE–488.2 also specifies the syntax of program and response messages. The new standard does not change the basic ideas from the previous standardized syntaxes for instrumentation systems, but it changes the terminology and coding rules somewhat. IEEE–488.2 introduces compound headers, which makes the coding of complex message structures easier. New types of messages such as common program messages, query and command messages, etc., live up to the demands of modern instrumentation systems. A broad range of data types can be used for parameter notation to support many applications.

Allowing the syntax to forgive variations in program data makes it user friendly. For instance you can send program messages in any mixture of upper- and lower- case alpha characters. You can send decimal data to the device in any notation form (with or without a decimal point, exponent, etc.). The data may be preceded by leading spaces and followed by a suffix for additional unit specification. A similar approach applies to the ending of messages; You may use several different terminator formats as long as the device is able to determine the end of a message unambiguously.

Status Reporting

IEEE–488.2 greatly improves the status reporting structure. The IEEE–488.1 standard requires the device to send a status byte only as the response to a serial poll. This limited specification resulted in a lot of different implementations, e.g., some instruments reported conditions and other instruments reported events in the status byte. The status byte bits where also reset at different times.

The IEEE–488.2 status byte has an overlying status reporting structure, containing events in the layer above the status byte, and optionally containing conditions in the layer above the events. This structure has several commonly reported events standardized, e.g., several error events, power-on event, etc.. These events are summarized in the status byte. The status reporting structure also makes the reporting of device–specific events easier. Several common program messages support this status reporting structure, e.g., commands for clearing the status reporting structure, selecting which events are to be summarized in the status byte by programming Enable registers, permitting certain status byte bits to generate service request by programming the service request enable registers, reading the standard event registers, etc.

Common Program Messages

IEEE–488.2 defines an extensive range of common program messages supporting commonly used functions. A subset of the common program messages are mandatory in all IEEE–488.2 compatible instruments.

With the common program messages it is possible to identify the device (manufacturer, model, and type number, etc.) and to determine the available options. It is also possible to let the device do a calibration cycle or an internal self test. Other features are common queries for the current instrument settings, or asking if an operation is completed. Other standardized program messages reset the device functions and trigger device actions.

Common program messages that support macros are also available. A macro may contain complex message structures stored by the user in the instrument. Sending a simple command (macro label) executes this complex message structure. Other common program messages control storing of the current instrument settings in the controller. You can use these instrument settings again to set up the instrument for the same measurements.

Message Exchange

Description of the Functions

The protocols of IEEE–488.2 define rules for the communication between the controller and the instruments. This includes defining when instruments should send response messages, when the controller can send program messages, and also what happens when the units do not follow the protocol.

The controller sends programming messages to the device. A program message may consist of commands and/or queries. Queries require the device to send a response message. Commands will not cause the device to send any response message. Queries differ from commands by the appended "?".

- *Message Exchange Model*

The following functional elements are the base of the protocol:

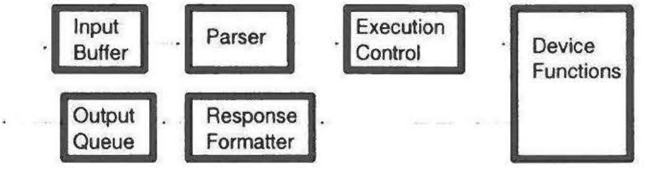


Fig 11-1 Simplified Message Exchange Model

Input Buffer. When you address the instrument as a listener, the bus interface receives message bytes from the controller and places them in the input buffer before parsing and executing them. The input buffer is large enough to hold a typical program message from the controller. This allows the controller to proceed with sending program messages to another instrument, while the first instrument is parsing and executing the commands from its input buffer. The input buffer of the PM 5781 can hold 128 characters.

Parser. The parser removes the data bytes from the input buffer. It interprets the data and translates it into internal device codes, which it passes on to execution control. The parser checks if the received data has the correct syntax and reports command and query errors. It detects whether a program message is a command or a query. If it is a query, the message exchange control function will allow the device to send a response message.

Execution Control receives successfully parsed messages and determines if a command can be executed in the current state of the instrument. If the execution control finds a parameter that is out of range, or if the current settings do not allow execution of a requested command/query, it will report an execution error.

Device Functions. This block contains all the device-specific functions and features of the PM 5781. It receives executable messages from the execution control block and makes the requested operations. It also supplies the response formatter with response data as a result to valid queries received. The device function block detects and reports internal operating faults.

Response Formatter. The PM 5781 will only give a response after receiving a valid query. The device functions block will do the operation requested by the query and then send the internal response data to the response formatter. From the internal response data the response formatter builds response message elements which follow the IEEE–488.2 syntax. Finally the response formatter places the formatted data as a response message into the output queue.

Output Queue. This is the memory area where the instrument stores all response messages until the controller reads them. The instrument removes the response messages from the output queue and sends them to the controller when the controller addresses the instrument as a talker. The output queue can hold 128 characters.

Protocol Requirements

In addition to the above functional elements that process the data, the following characteristics of the message exchange protocol can be mentioned:

- The controller must end a program message containing a query with a message terminator before reading the response from the device (address the device as talker). If the controller breaks this rule, the device will report a query error.
- The controller must read the response to a query in a previously (terminated) program message before sending a new program message. When the controller violates this rule, the device will report a query error.
- The instrument sends only one response message for each query message. If the query message resulted in more than one answer unit, all answers will be sent in one response message.

• *Order of Execution* **Deferred and Coupled Commands**

Execution control collects commands until the end of the message, or until it finds a query or other special command* that forces execution. It then checks that the setting resulting from the commands is a valid one: no range

limits are exceeded, no coupled parameters are in conflict, etc. If this is the case, the commands are executed, otherwise an execution error is generated and the commands are discarded.
This partially deferred execution, due to the coupling between most of the pulse parameters, guarantees that:

- All valid commands received before a query are executed before the query is executed.
- All queries are executed in the order they are received.
- The order of command execution is never reversed with respect to the order the commands were sent. At most, the execution times coincide. If it is vital to enforce the execution of one part of a program message before a later part, the 488.2 command "✱WAI" is used.

Executing Commands

Forces execution of all deferred commands received earlier in the same program message. All queries are executing.

Immediate Commands

These commands are executed immediately without executing other commands first. After the immediate command has been executed, the pulse generator resumes execution of other commands from the input queue, except if the immediate command is a command that resets the input queue (✱RST).

Device Clear

The Device Clear (DCL) or Selective Device Clear (SDC) command clears the input buffer and the output queue, resets the parser, and clears any pending commands.

The device clear commands will not:

- Change the instrument settings or stored data in the instrument.

- Interrupt or affect any device operation in progress.
- Change the status byte register other than clearing the Message Available (MAV) bit as a result of clearing the output queue.

NOTE: Many older IEEE instruments, that where not IEEE–488.2 compatible, returned to the power-on default settings when they received a device clear command. IEEE–488.2 does not allow this.

- *When to Use a Device Clear Command*

The Device Clear command is also useful to escape from erroneous conditions without having to alter the current settings of the instrument. The instrument will then discard pending commands and will clear responses from the output queue. For example, suppose you are using the PM 5781 in an automated test equipment system, where the controller program returns to its main loop on any error condition in the system or the tested unit. In order to ensure that no unread query response remains in the output queue, and no unparsed message remains in the input buffer, it is wise to use the Device Clear command. (Such remaining responses and commands could influence later commands and queries.)

Remote /Local protocol

- *Definitions*

Remote Operation

When an instrument is in Remote Operation, all local controls, except the local key, are disabled.

Local Operation

An instrument is in Local Operation when it is not in remote mode as given above.

Local Lock Out

I addition to the remote operation state, an instrument can be set to remote with 'Local Lock Out'. This disables the return-to-local button. In theory the state local with local lock out is also possible; then all local controls except the return-to-local key is active.

The PM 5781 in Remote Operation

When the PM 5781 is in Remote Operation, it disables all its programmable local controls.

The PM 5781 in Local Operation

When the PM 5781 is in Local Operation, the instrument is fully programmable both from the front panel and from the bus. If a bus message arrives while a change is being entered from the front panel, the front panel entry is interrupted and the bus message is executed.

Syntax and Coding of Messages

Basic Message Structure

The basic message structure is shown in the following two syntax diagrams. This structure is valid for the messages from the controller to a device (program messages) and the messages from the device to the controller (response messages)

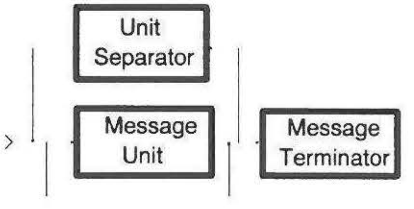


Fig. 11-2 Basic Message Structure

Message Unit Structure

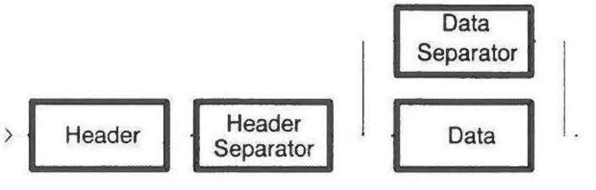


Fig. 11-3 Message Unit Structure

The syntax diagrams for program messages shows many more variations than those for response messages since instruments listen in a more flexible manner than they send. To keep the syntax diagrams readable this section will not show all the variations, but a few examples are given to show some differences. To learn what the protocol allows, please refer to the official IEEE Standard 488.2-1987.

Program Messages Features:

— Forgiving syntax.

The following syntax diagram shows the structure of a program message. The syntax matches the basic structure of a message given above. However, in this diagram the (program) message unit is integrated into the syntax of the complete (terminated) program message.

- *Program message syntax*

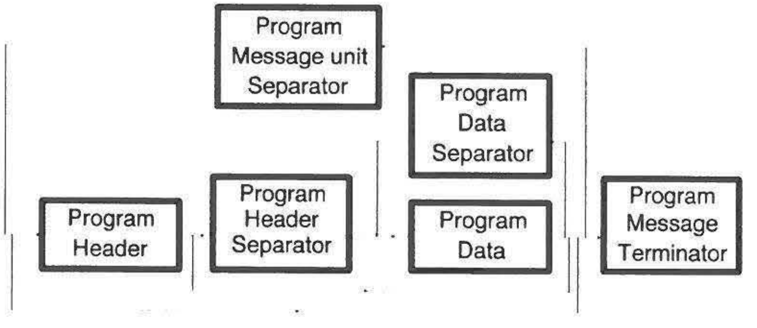


Fig 11-4 Syntax, Terminated Program Message

Note that you may omit data fields, and use only a header. The program messages can be split up into two groups:
Commands: Do not allow the device to generate a response.

Queries: Require the device to generate a response. A question mark "?" ends the program header. For example "✱IDN?".

There are three types of program headers:
- Simple Header. Device specific program message which contains a single header (mnemonic).
- Common header. Common (IEEE 488.2 defined) program message which starts with ✱.
- Compound header. (Not used in PM 5781.) Device specific program message containing one or more program headers separated with a colon (":").

Examples of Program Headers		
	Commands	*Queries*
Simple	PERIOD	PERIOD?
Common	✱TRG	✱LRN?
Compound (not used in PM 5781	OUTP: IMP 50	OUTP: IMP?

Note: When using compound headers, (not used in PM 5781) the 'command tree' determines the order of the headers. For instance, a root level command can always be parsed if it is the first command of a program message or if it is preceded by a colon. A leading colon or a program message terminator will reset the parser to the root level of the command tree.

Response Messages

Features:

- Precise syntax

- Header is optional, i.e., it allows data only.

The following syntax diagram shows the structure of a response message. The syntax matches the basic structure of the message, given above. However, in this diagram the (response) message unit is integrated into the syntax of the complete (terminated) response message.

- *Response Message Syntax*

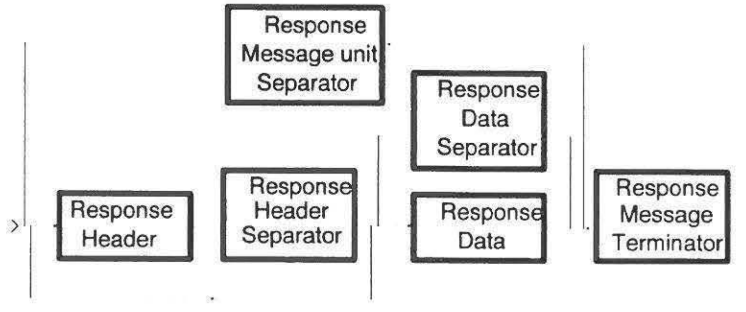


Fig 11-5 Syntax, Terminated Response Message

Note that the header may be omitted, so that only data is output.

- *Message Data Coding*

Data transfer normally uses 7-bit ASCII coding. IEEE–488.2 defines special data types to make it possible to transfer any 8-bit device specific code.

Separators and Terminators

Separator/ terminator	Code	Example
Header Separator	space character (= ASCII-code 32 decimal)	PER 0.1
Unit separator	;	PER 0.1 ; DUR 0.01
Data separator	,	OUTPUT ON , A , B
Message terminator	↵ ↵ ^End dab ^End	OUTPUT OFF ↵ OUTPUT OFF ↵ ^End OUTPUT OFF ^End

^End means that the data byte, dab, which is written with the ^End appended, is sent with the GPIB EOI line set true, which many controllers and instruments are able to use to detect the end of a message.

Data Formats

The data formats can be split up into two groups: the data format for program data (sent to the instrument), and that for response data (sent from the instrument).

The following four examples show the added tolerance in syntax for program data versus the precise syntax of response data.*

Message Element	Response Message	Program Message
Decimal Numeric Data	NRf	NRf f = flexible. NR1, NR2 or NR3 can be used.

Examples of the NR1, NR2 and NR3 are given under "Numeric Data".*>

Message Element	Response Message	Program Message
String Data	enclosed in double quote (") characters	enclosed in double (") or single (') quote characters.
Headers	only upper-case alpha	upper-case and lower-case alpha can be mixed.
Header Separators	one single space character (˽ = ASCII-code 32 decimal)	multiple white-space characters can be used. The white-space includes all ASCII-codes in the range 0-9 and 11-32 decimal.

- *Character data*

For parameters that can best be expressed mnemonically as short alphanumericals and/or numerals.

- Alpha characters, digits, underscore
  example: PERIOD   OUTPUT   AUTO

String data

Useful to transmit text, for example, for printers, plotters, CRT's.

- Any ASCII character(s) enclosed in single or double quote characters. If you want a quote inside the string of the same type as used to enclose the string, this quote must be doubled inside the string.
  Examples:
  "THIS IS ""STRING"" DATA"
  "THIS IS 'STRING' DATA"
  'THIS IS "STRING" DATA'

Expression data

- Any printable ASCII character except # ' " ( ) ; enclosed in parentheses ( ). Expressions may be nested.
  Example: (A+B/C✱3)
- Specific program data elements except indefinite arbitrary block data.
  Example: ("RESULT =";A✱B(+3))

• Numeric data

Decimal data

Program Data: NRf with possibility for suffix

Example:0.032E+3 (NR3)	032.0 (NR2)	32 (NR1)

Response Data:

PM 5781 responds with one digit, decimal point, decimals, and an exponent
Example: 8.25E-6

Non-decimal data

IEEE 488.2 allows the manufacturer of instruments to use hexadecimal, octal, and binary data. These are not used in PM 5781.

• Arbitrary block data

For transmission of any 8–bit codes

Definite type #212cccccccccccc

Where the first digit (2) shows how many of the following characters determines the length of the block data. The following digits shows that this block consists of 12 characters.

Indefinite type #0ccccccc lf ^End

Only allowed at the end of a terminated response message.

• Arbitrary ASCII response data

Any ASCII byte except nl

Status Reporting

Status reporting model

This section handles the status reporting features of the device. Figure 11-6 shows a block diagram for the IEEE–488.2 Standard Status data structure. The status model is based on reporting events or queue status. An event is a transition of a device condition. The events and the device conditions may be represented in registers (event and condition registers), but may also be logical states of the hardware. An example of an event is when an error occurs in the device.

By programming the Event Enable Register, you can select which events in the event registers the instrument should summarize, (logical OR) into the IEEE–488.1 status byte register.

The instrument summarizes the events from the standard event status register in the Event Status Bit (ESB–bit 5) and the events from a device defined event status register in one of the bits 0, 1, 2, 3 or 7 (see fig. 6).

When you read the event registers, by using the related query, you will also clear the registers you read. You can also clear them with the *CLS (Clear Status) command which clears all event registers.

You can clear an event enable register by programming it with a value of zero. Power-on clears all event enable registers and the service request enable register, if the power-on status clear flag is set TRUE. (See the common command *PSC.)

The not-empty-status of the output queue is reflected in the message available (MAV) bit (bit 4).

You can select which status byte bits will cause the device to generate a service request (setting the RQS bit TRUE) by setting bits in the service request enable register. See *SRE in the 'Command reference' chapter.

You can read the status byte using a serial poll or the *STB? query. This will not affect the contents of the status byte (except the RQS bit when read by serial poll).

The instrument clears only the summary message bit (standard or device defined) in the status byte when you clear or read the related event register, or if a queue model status data structure is tied to the summary bit, when the queue is empty.

Example: The instrument clears the MAV-bit in the status byte register when the output queue is empty, i.e., when it has sent the complete response message.

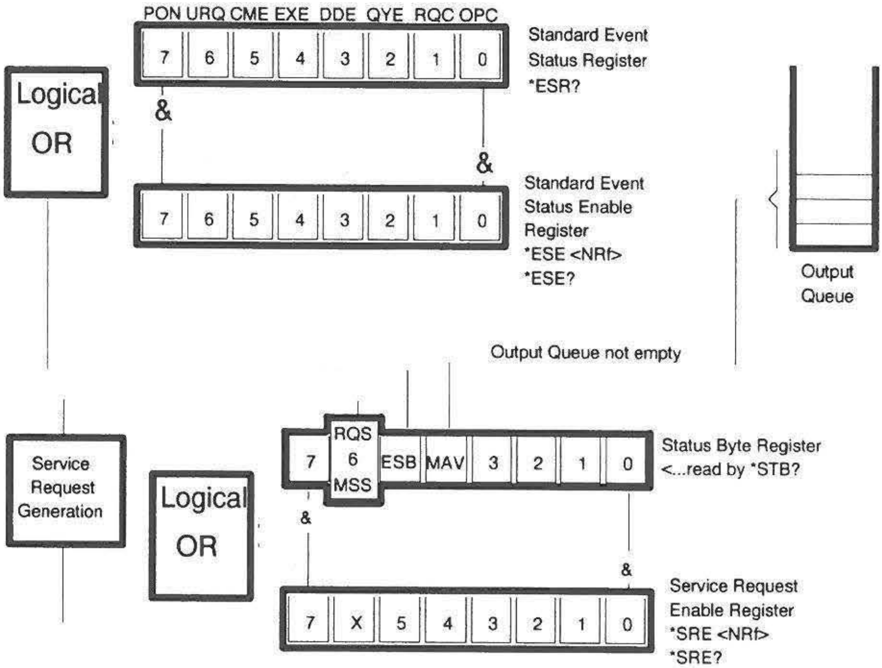


Fig 11-6 Overwiew of Standard Status Data Structures

All bits (except bit 6, RQS) in the status byte register are summary messages. Each summary bit summarizes an overlying status data structure. An example of a register model overlying status data structure is given in figure 11-7.

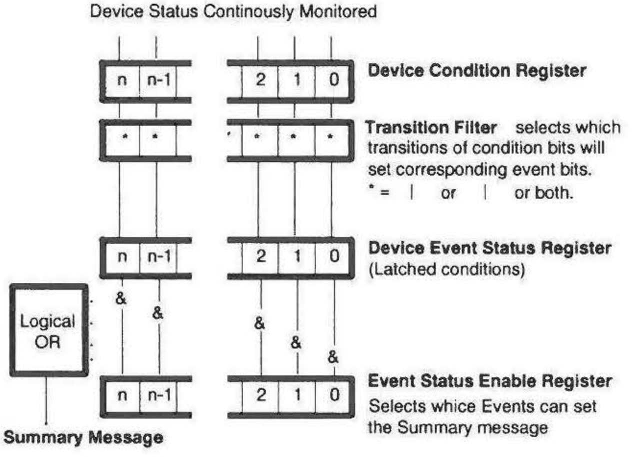


Fig. 11-7 Model of Overlying Status Data Structure

Summary messages always track the status of the associated status data structure. Reading the status byte register by serial poll or by *STB? does not clear its summary bits. The instrument clears a summary bit only when it clears the associated event register or queue (see figure above).

Service Request and Reading of the Status Byte

- *Selecting a Summary Message to Generate SRQ*

You can program the service request enable register (*SRE <number>) to select which summary message(s) in the status byte register will cause the device to generate a service request and as a result, set the RQS (requested service) bit in the same register. When a service request occurs, the SRQ line of the GPIB will be activated. Whether or not the controller will react on the service request depends on the controller program. The controller may be interrupted when a service request occurs or it may regularly test the SRQ line, or the controller may not react at all.

Example: *SRE 16

This sets bit 4 ($16=2^4$) in the service request enable register (see figure 11-6). The instrument ANDs this bit with the corresponding bit (bit 4 = MAV) in the status byte register. It directs the result to the input of a logical OR, the output of which sets the MSS (Master Summary Status) bit. Now when a response message is available in the output queue, it sets the MAV bit TRUE. As a consequence, it sets the MSS bit TRUE and generates a service request by setting the RQS (requested service) bit TRUE. When the GPIB controller reacts to the service request with a serial poll, the instrument resets the RQS bit FALSE. When a new service request occurs the instrument sets the RQS bit TRUE again. The MSS bit remains TRUE after the serial poll. When the GPIB controller reads the response message from the device, the instrument empties the output queue and resets the MAV bit. This also resets the MSS bit if MAV was the only enabled summary message.

You can read the status byte register in two ways:

- Using the Serial Poll (IEEE–488.1 defined).
  Response:
  Bit 6: RQS message, shows a service request.
  Other bits: Summary messages (same as using *STB?)
  A serial poll sets the RQS-bit FALSE, but does not change other bits.
- Using the common query *STB?
  Response:
  Bit 6: MSS message, shows that there is a reason for service request.
  Other bits: same as using serial poll.
  Reading the response will not alter the status byte.

The above does not completely explain the difference between the MSS and RQS bits. An IEEE–488.2 defined synchronization mechanism takes care of not losing a new reason for service request, while a serial poll is in progress. That is, during a serial poll, the instrument may not yet have removed the current service request when a new reason for service request occurs. It sets the MSS bit TRUE, but it holds off the new service request until it has finished the current serial poll. The instrument now carries out the service request and sets the RQS bit TRUE. Now the controller can react to the new service request.

Standard Event Status Register

• *Bit Definitions*

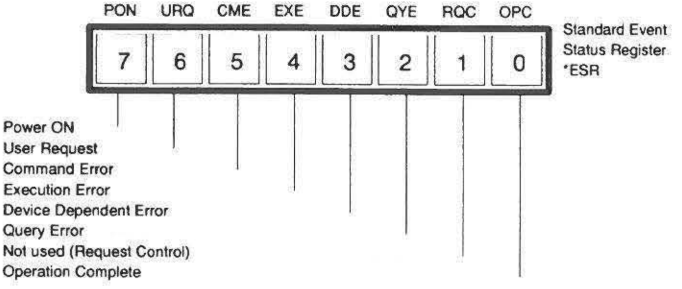


Fig. 11-8

Bit 7 - Power On (PON)

Shows that the device's power supply has been turned off and on (since the last time the controller read or cleared this register).

Bit 6 - User Request (URQ)

Shows that the user has activated the URQ signal. This bit will be set regardless of the remote local state of the device. The purpose of this signal is, for example, to call for the attention of the controller by generating service request. (How the URQ signal will be activated is device dependent. In PM 5781 all active keys will set the URQ signal, except reset/local).

Bit 5 - Command Error (CME)

Shows that the instrument has detected a command error. This means that the device has received data that violates the syntax rules for program messages.

Bit 4 - Execution Error (EXE)

Shows that the device detected an error while trying to execute a command (see 'Error handling' on page 11-45). The command is syntactically correct, but the instrument cannot execute it, for example because a parameter is out of range.

Bit 3 - Device-Dependent Error (DDE)

A device-dependent error is any device operation that did not execute properly because of some internal condition, for instance overload. This bit shows that the error was not a command, query, or an execution error.

Bit 2 - Query Error (QYE)

The output queue control detects query errors. The QYE bit shows for example the unterminated, interrupted and dead-lock conditions. For more details see 'Error handling' on page 11-45.)

Bit 1 - Request Control (RQC)

Shows the controller that the device wants to become the active controller-in-charge. Not used in the PM 5781.

Bit 0 - Operation Complete (OPC)

The instrument sets this bit TRUE only in response to the operation complete command (*OPC). It shows that the device has completed all previously started actions.

- *Using the Standard Event Status Register*

Reading: *ESR? This also clears the register.

Example of response: "136"

Explanation: $136 = 2^7 + 2^3$

bit 7 = PON = Power-on

bit 3 = DDE = Device Dependent Error

Writing: Not possible

Clearing:

1. ✱CLS
   Use the ✱CLS command with caution: The command will not only clear the Event Status Register, but will clear all status data structures including the output queue.

2. ✱ESR?

Device Defined Status Registers

The data structures can be registers or queues. IEEE–488.2 defines the standard status data structures and the manufacturer of the instrument defines the Device Defined status data structures..

The standard status data structure #4 is a queue model structure (output queue), and the standard status data structure #5 is a register model structure (event status). They reflect their summary message in the MAV and ESB bit of the status byte register respectively.

The PM 5781 has two device–defined status registers:

- Data structure #7 The "operation status group" summarized in bit 7, OPR.
- Data structure #3 The "questionable data/signal group" summarized in bit 3, QUE.

These groups are 16-bits wide while the status byte and the standard status groups are 8-bits wide.

• *Device–Defined Status Data Structure*

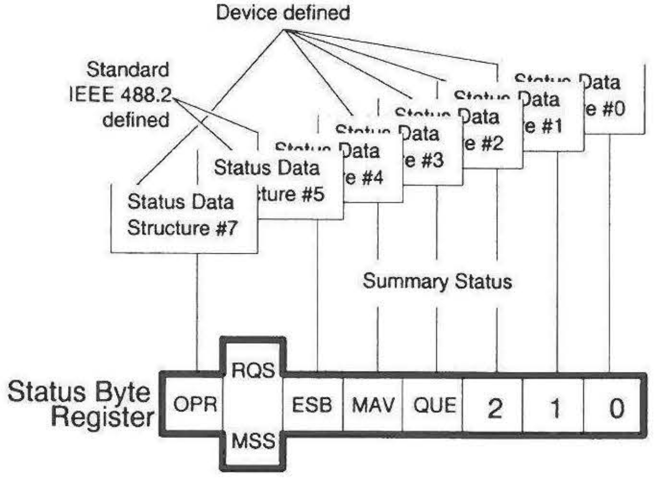
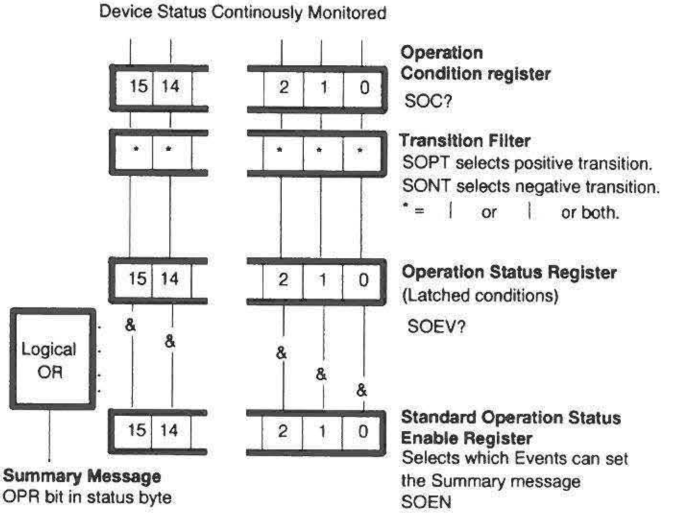


Fig. 11-9 Device-Defined Status Data Structures (model)

This model is the same as the model used for the standard event status, figure 11-6. The register model may include transition filters (see figure 11-7) for the creation of events derived from the device–defined conditions.

PM 5781 uses status data structure #7, operation status group, and status data structure #3, questionable data/signal status group.



11-10 Status data structure 7

PM 5781 status condition registers

The condition register monitors the hardware and software status of the device continuously. The register is updated in real time and it can only be read. The operation status condition register is read with SOC?, and the questionable data/signal condition register is read with SQC? but they cannot be read during an on-going condition. So, use SRQ to monitor conditions.

PM 5781 transition filters

The transition filter specifies if the beginning or the ending of a condition should set the corresponding bit in the Event Status register. Both transitions can be selected for one and the same condition. They can be set with:

SOPT (Status) Operation Positive Transition Filter.

SONT (Status) Operation Negative Transition Filter.

SQPT Status Questionable Positive Transition Filter.

SQNT Status Questionable Negative Transition Filter.

PM5781 Status Event Registers

This register latches the events from the condition registers if allowed by the transition filter. The bits in this register stay set until the register is read or cleared with the *CLS command (clear status). *CLS clears all event status registers, both the device defined and the standard defined. It is neither meaningful nor permitted to write to an event status register. The registers are read by:

SOEV? (Status) operation event.

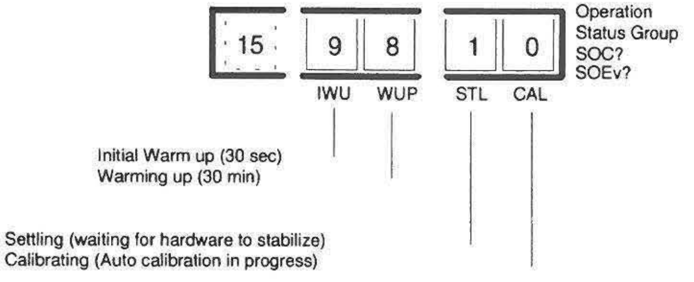
SQEV? Status Questionable event.

PM 5781 Status Event Enable Registers

The enable register selects which bits of the event status registers should be summarized in the status byte. You can clear an event status enable register by programming it with a value of zero or by sending *CLS.

NOTE: If a condition should be summarized in the status byte, the corresponding bit must be selected in both the transition filter and the enable register.

• *Operation status group*



11-11 Bits in the Operation Status register

Bit 9 - Initial Warm-Up (IWU)

This bit indicates that the voltages from the power supply in the pulse generator are stabilizing. The stabilizing process continues 30 seconds from power on.

Bit 8 - Warming Up (WUP)

This bit indicates that the pulse generator is warming up. The warming up process continues 30 minutes after power on.

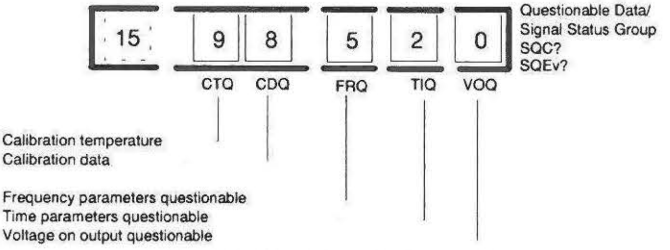
Bit 1 - Settling (STL)

This bit indicates the time it takes for the pulse generator hardware to stabilize after a change of pulse parameters. (For instance, can be used check that the period is properly regulated before measuring.)

Bit 0 - Calibration (CAL)

This bit indicates that the pulse generator is calibrating itself, i.e., the self-calibration routine has been activated.

- *Questionable data/signal status group*



11-12 Bits in Questionable data register

Bit 9 - Calibration Temperature Questionable (CTQ)

The temperature inside the pulse generator is outside the range where the calibration is accurate.

Bit 8 - Calibration Data Questionable (CDQ)

The calibration data stored in the instrument is lost or faulty. The reason can be that the battery is discharged or that the default calibration data has been recalled.

Bit 5 - Frequency Questionable (FRQ)

The output frequency (period time) is of questionable quality. If period control is on, the period control loop may not be completed. If period control is off the warm up may not be completed, the temperature may be out of calibration range, or the calibration data may be bad.

Bit 2 - Time Parameters Questionable (TIQ)

The time parameters are of questionable quality. The warm up may not be completed, the temperature may be out of calibration range, or the calibration data may be bad.

Bit 0 - Voltage questionable (VOQ)

The output voltage is of questionable quality. The warm up may not be completed, the temperature may be out of calibration range, or the calibration data may be bad.

- *Queue status Data Structure*

The queue model status data structure allows reporting sequential status. It is not used in PM 5781. The output queue with its associated MAV bit is an example of a queue model status data structure, although this is a standard status data structure. (See figure 1111-88.)

Status of the Output Queue

The MAV (Message Available) summary message is a standard IEEE–488.2 defined message, which appears in bit 4 of the status byte register. A '1' in bit 4 (MAV summary message TRUE), shows that the instrument has data in the output queue. When the controller reads the data, the output queue will be emptied. For as long as the output queue holds one or more data bytes, the MAV bit in the status byte register remains TRUE. when all the data have been sent, the output queue is empty and MAV is set FALSE.

Setting Up the Instrument

Power-On

When the PM 5781 is turned off the battery back up of the internal memory stores the setting. When the PM 5781 is turned on, the previous setting is displayed.

The outputs are always disabled when the generator is switched on to avoid damage to connected equipment.

If the power-on status clear flag in the instrument is set FALSE, power-on does not clear the *SRE and the *ESE registers (see 'Command reference' chapter, *PSC).

If the power-on status clear flag in the instrument is set TRUE, all condition registers and all enable registers will be cleared, but not the transition filters.

Identification

Use *IDN? to identify the instrument and *OPT? to identify which options are installed. (See 'Command reference' chapter, *IDN? and *OPT?)

Initialization and Resetting

- *Reset Strategy*

There are three levels of initialization.

1. Bus initialization
2. Message exchange initialization
3. Device initialization

Bus initialization

This is the first level of initialization. The controller program should start with this which initializes the IEEE interfaces of all connected instruments. It puts the complete system into remote enable (REN-line active), and the controller sends the interface clear (IFC) command. The command or the command sequence for this initialization is controller and language dependent. Refer to the user manual of the system controller in use.

Message exchange initialization

Device clear is the second level of initialization. It initializes the bus message exchange, but does not affect the device functions. Device clear can be signalled either with DCL (to all instruments) or SDC (Selective device clear) only to the addressed instruments. The instrument action on receiving DCL and SDC is identical. See 'Device Clear' on page 11-10.

When you use DCL, remember that it is always sent to all connected instruments and that old instruments (pre 488.2) may e.g. set default settings when receiving this command..

Device Initialization

The third level of initialization is on device level. This means that it concerns only the addressed instruments.

<u>The *RST command.</u>

Use this command to make a device reset. It initializes the device-specific functions in the PM 5781.

The following happens when you use the *RST command:

- The pulse-generator-specific functions are set to a known default state. This state is the same as when you recall setting 0. (See *RCL in the 'Command reference' chapter).
- The macro, defined by *DDT is set to a device-defined state. (See *DDT in the 'Command reference' chapter)

- Macros are disabled.
- The pulse generator is set in an idle state (outputs are disabled), so that it can start new operations.

The *CLS command.

Use this command to clear the status data structures. (See "Status Reporting" in this chapter.)

The following happens when you use the *CLS command:
- The instrument clears all event registers summarized in the status byte register.
- The command will clear all status data structures including the output queue. It empties all queues.

Storing Instrument Settings

The common query *LRN? and the queries BINPROG? and MEMPROG? generates a response containing all instrument settings (including enabled outputs!). You can store the response and resend it to set the device in the same state as it was when you issued the query.

The difference between the queries is in the response: The BINPROG? and MEMPROG? queries gives an arbitrary block response, which is a compact format. The *LRN? query results in a response using normal program messages you can read and edit. This response contains more data bytes to represent the same information.

Programming-Codes query

The CODES? query gives a response with a readable list of all possible program messages. The list can be helpful when writing a program for the instrument, informing you of its programming possibilities. SYNTAX? explains the syntax of a specified command.

Calibration and Self -test

- *Calibration Query* The ✱CAL? command tells the instrument to make an internal calibration. The response shows a value of 0 if the instrument completed the internal calibration without errors. Any other value means an error. (The response is an <NR1> integer in the range -32767 to +32767, see the Command Reference chapter)

- *Self-test Query* The ✱TST? command makes the instrument perform an internal self-test and reports a '0' if the test passed without failures. When the test failed, the PM 5781 will output an error number, showing what is wrong with the instrument. (See ✱TST in the 'Command reference' chapter.) You can use the DEFTEST command to define which tests the instrument should make, when receiving the ✱TST command. (See the 'Command reference' chapter.)

- *Define Self-test Command* Use the DEFTEST command to define the internal device self-test the instrument should execute when receiving ✱TST?. See also under "✱TST?".

- *Define Self-test Query* The DEFTEST? query gives a response containing the internal device self-tests executed when using ✱TST?. See also the 'Command reference' chapter.

Programmable Functions

Memory Storage for Instrument Settings

- **SAV and *RCL*

Save and recall commands

You can store up to 49 full instrument settings in the nonvolatile back-up memory of the PM 5781.
Use the *SAV <number> command to save the current settings of the PM 5781 in the register number <number>. Use the *RCL <number> command to recall the settings from the register number <number>. Register 0 always contains predefined default settings. Switching the power off and on does not change the settings stored in the registers.
The contents of memory register 0 cannot be changed by a *SAV 0 command. If you try to do so anyway, the PM 5781 generates an execution error.

The Scope of *SAV and *RCL

The following functions are saved/recalled when using the *SAV and *RCL:

Scope of *SAV, *RCL, BINPROG, MEMPROG, *RST, *LRN? in PM 5781			
Pulse Parameters	Period	Delay	Duration
	Leading Edge	Trailing Edge	Burst Period
	Burst	High Level	Low Level
	Offset_B	Tracking	Set increment
	Locking	Limiter	Voltlimits
	Period Control		
Trigger	Trigger Source	Trigger Function	Define Device trigger

Scope of *SAV, *RCL, BINPROG, MEMPROG, *RST, *LRN? in PM 5781			
Input	Trig Level	Trig Slope	Impedance
Pulse Mode	Pulse Mode	Complementary	
Output	Clock level	Disable A, B and clock	
Help	Debug	Error Display	
System	Display Mode	Display Format	Beeper
	Response Header		Error Check

NOTE: The contents of macros are not stored or retrieved!

Macros

A macro is a single command which represents one or several other commands, depending on the definition. You can define 24 macros with a total length of 256 characters in the PM 5781.

Use macros to:

— Provide a shorthand for complex commands.

— Cut down bus traffic.

• *Defining Macro Command*

*DMC assigns a sequence of commands to a macro-label. When you later use the macro-label as a command the pulse generator will execute the sequence of commands.

Use the following syntax:

*DMC <macro-label>, <Arbitrary block program data>

Example:

*DMC "MYPER",#224 PER $1 ; LEE 1E-6 ; TRE 1E-6 nl

This defines a macro named MYPER which takes one argument (the period) and sets the leading and trailing edges to fixed values.

One macro can address other macros, but the macro definition does not allow calling a macro within itself (recursion).

You can pass arguments (variable parameters) with the macro. Insert a dollar sign ($) followed by a single digit in the range 1 to 9 where you want to insert the parameter. See the preceding example.

When a macro with defined arguments is used, the first argument sent will replace any occurrence of $1 in the definition; the second argument will replace $2 etc.

Example:

*DMC "EDGE" , #213LEE $1 ; TRE $1
defines the macro EDGE.

(#213 defines the length of the arbitrary block data element, see data formats in this chapter. LEE $1 ; TRE $1 is the actual macro)

Now sending the command "EDGE 1.0E-9"
sets both TRE and LEE to 1.0E-9.

You can use both commands and queries as macro-labels. Remember to put a question mark after the macro if you want a response from the instrument. The label cannot be the same as common commands or queries. If a macro label is the same as a PM 5781 command, the PM 5781 will execute the macro when macros are enabled and it will execute the PM 5781 command when macros are disabled, see 'Enabling and Disabling Macros' below.

- *Deleting Macros*

Use the ***PMC (purge macro) command to delete all macros defined with the ***DMC command. This removes all macro labels and sequences from the memory. It is not possible to delete only one macro from memory.

• *Enabling and Disabling Macros*

✱EMC Enable macro command

When you want to execute a PM 5781 command or query with the same name as a defined macro, you must disable macro execution. Disabling macros does not delete stored macros, it just prevents them from executing.

Disabling: ✱EMC 0 disables all macros.
Enabling: ✱EMC 1

✱EMC? Enable Macro Query

Use this query to determine if macros are enabled.

Response:

1	macros are enabled
0	macros are disabled

• *Executing Macros*

Macros are disabled after ✱RST so, to be sure, start by enabling macros with ✱EMC 1. Now macros can be executed by using the macro labels as commands.
Example:

✱DMC "ONE_MS";#225 PER 1E-3 ; LEE 1E-6 ; TRE 1E-6 ↵
✱EMC 1

Now sending the command ONE_MS will cause the PM 5781 to program the period to 1 ms and the leading and trailing edges to 1 µs.

• *Retrieving Macros*

✱GMC? Get macro contents query

This query gives a response containing the definition of the macro you specified when sending the query.
Example using the macro defined above;

command: ✱GMC? "ONE_MS"
response: #225 PER 1E-3 ; LEE 1E-6 ; TRE 1E-6↵

✱LMC? Learn macro query

This query gives a response containing the labels of all the macros stored in the pulse generator.

Example:
  Command: *LMC?
  Response: "MY_PER" , "ONE_MS"

There are at the moment two macros in memory and they have the following labels: "MY_PER" and "ONE_MS"

Protected data

- *Protected User Data Command*

Use the *PUD command to store data such as the calibration date; inventory number etc., in the nonvolatile memory of the instrument. The data is protected against unintentional alteration. You must use the UNPROTECT command in the same programming message to be able to write in this memory. The new message is automatically protected.

Syntax:

*PUD <ARBITRARY BLOCK PROGRAM DATA>

Example:

UNPROTECT <password>; *PUD # 241 CALIBRATED 1990-05-01, INVENTORY NO.02027 nl

See also the 'Command reference' chapter.

- *Protected User Data Query*

Use the *PUD? query to read the protected data, previously stored with the *PUD command.

Example: *PUD?
  Response: # 241 CALIBRATED 1990-05-01, INVENTORY NO.02027 nl

Resource Description

This command makes it possible to store a description of the performance of the instrument. PM 5781 does not support this function.

Triggering

- *✱TRG Trigger Command*

The trigger command has the same function as the Group Execute Trigger command GET, defined by IEEE–488.1.

When to use ✱TRG and GET

The ✱TRG and the GET commands have the same effect on the instrument. If the pulse generator is idle, i.e., not parsing or executing any commands, and the ✱DDT has not been used to redefine the trigger command, GET will execute much (≈ 20 µs) faster than ✱TRG (≈ 4 ms), since the instrument must always parse ✱TRG.

✱DDT Define Device Trigger Command

If you want the pulse generator to execute specific commands when triggered by ✱TRG or GET, define the trigger actions with ✱DDT.

Example:✱DDT #217 TRGSRC MAN ; MANTRG

If you send ✱TRG or GET, the PM 5781 will now execute the command string "TRIGSOURCE MANUAL ; MANUALTRIG". This string will have the same effect as selecting manual triggering and pushing the MANUAL key on the front panel.

✱DDT? Define Device Trigger Query

This query reads the command sequence that the PM 5781 will execute when it receives the ✱TRG or GET command.

Example:✱DDT?

Response: #217 TRGSRC MAN ; MANTRG When using the above definition.

Automatic Instrument Setting

The command AUTOSET is the same as the AUTO SET key on the front panel. It corrects erroneous pulse generator settings using the selected parameter as a reference. AUTOSET does not change locked parameters or track-relations. This command has to be sent in the same program message as the erroneous settings it is supposed to correct. The instrument will otherwise discard the erroneous settings.

Syntax: AUTOSET <parameter>

Where <parameter> is the parameter you want automatically set.

Example: PER 15E-9 ; AUTOSET PER

This example gives 15 ns period regardless of the other settings.

Configuration of the Displays

DISPMODE, Display mode command

This command defines which parts of the display will be active.

Syntax: DISPMODE ALL | STATE | OFF

Where:

ALL = All will show all the values in the different displays (default).

STATE = Means that the central display will show "FRONT PANEL DISABLED" but all the other indicators and displays will work.

OFF = Means that the central display will show "FRONT PANEL DISABLED" and all other indicators and displays except the bus status are blank.

DISPMODE?, Display mode query

The response to this query tells you which display indicators are active.
Example: DISPMODE?
Response: DISPMODE STATE

Error Handling

Survey of Error Events

Four types of errors can occur in the instrument:
- Command Errors
- Query Errors
- Execution Errors
- Device Dependent Errors

• *Command Errors*

The parser will detect Command Errors and report them by setting the Command Error bit in the Standard Event Status Register.

The Command Errors are classified as follows:

Syntax Error. Occurs when a Program Message is in violation of the IEEE–488.2 standard. This is the case if the byte sequence of the Program Message does not follow the proper encoding syntax rules. This error category also includes wrong parameter type for the associated header.

Semantic Error. Occurs when the instrument receives an unrecognized header. This can be an incorrect PM 5781–specific program header or an incorrect or un-implemented IEEE–488.2 common command.

Group Execute Trigger (GET) received inside a Program Message. Occurs when the controller sends a GET in the middle of a program message, i.e., between the first byte of the Program message and the program message terminator.

Device response to command errors

When the parser detects a command error, the PM 5781 will respond as follows:
- It sets the CME (command error) bit in the standard event status register.

- It discards all the following data bytes until it receives a program message unit separator, the program message terminator, or the ^END message.

I.e. any valid command parsed before the unit containing the command error will have been sent for execution, and new valid units after the invalid one will also be sent for execution.

- *Query errors*

The pulse generator will report a query error if the proper protocol for reading a query is not followed. Some reasons for query errors are as follows:

Unterminated action is if the controller attempts to read a response message from the PM 5781 without first having sent a complete query message (including the program message terminator) to the instrument. The PM 5781 will set the query error bit in the standard event status register. It will reset parser and clear the output queue. This means that the pm 5781 will not execute the unterminated, partially parsed program message unit.

Interrupted action is if the PM 5781 has not finished sending a response message before being interrupted by a new program message from the controller. The PM 5781 will set the query error bit in the standard event status register. It will discard the unread portion of the response message but will execute the program message that interrupted the response.

Buffer deadlock is if both the input buffer and the output queue become full. This can happen when sending multiple queries, in the same message, with a length longer than the input buffer (128 characters). The total length of the program message becomes longer than the input buffer can hold and the queries produce more response data than the output queue can hold. The controller cannot read the response data from the output queue of PM 5781 until it has completed sending the entire program message, up to and including the PMT (program message terminator). Fortunately, the PM 5781 will break such a deadlock. It will set the query error bit in the standard event status register, clear the output queue, and resume reading, parsing, and executing the remaining portion of the program message. It will not generate any response data from the remaining program message units, so the output queue will remain empty.

Query after indefinite response. Occurs when the controller sends a query that causes the PM 5781 to generate an indefinite response (<INDEFINITE LENGTH ARBITRARY BLOCK RESPONSE DATA> or <ARBITRARY ASCII RESPONSE DATA>). This query message unit must be the last in the program message. The PM 5781 will not respond to any query received afterward in the same message.

- *No Error Reporting* **Addressed to talk with nothing to say.** If the controller attempts to read the response from a query before the PM 5781 has finished executing the query (for example a measurement), it will not provide any response. The PM 5781 will not set the query error bit in the standard event status register in this case. The controller must wait until the response becomes available.

Serial poll during a program message or a response message. If the controller makes a serial poll while the PM 5781 is receiving a program message or sending response data, the PM 5781 will respond by sending its status byte and then continue receiving the rest of the program message or finishing the response.

Unaddressing during a program message or a response message. If the controller un-addresses the PM 5781 during the input of a program message or during the output of response data, the pulse generator suspends the input or output until it is re-addressed.

Device Response to Query Errors

The PM 5781 will report a query error by setting the query error bit in the standard event status register. This register can be read by the *ESR? command.

Additionally the PM 5781 will;
- Clear the output queue:
- Discard partially parsed program message units when an uncompleted command or query causes the query error.
- Execute completely parsed program message units.
- When a deadlock causes the query error, the PM 5781 will continue to parse and execute program message units. It will, however, discard any query responses until it finds the program message terminator in the input buffer.
- When an interrupted response causes the query error, the PM 5781 will clear the output queue.

• *Execution errors*

When the execution control block detects an execution error, the PM 5781 will set the EXE (Execution Error) bit in the standard event status register.

The PM 5781 will report an execution error under the following conditions:
- When the program data following a program message header is outside its legal input range or is inconsistent with the capabilities of the PM 5781. For example a coupled parameter violates the capabilities of the PM 5781;
- When the current functional state of the PM 5781 does not allow execution of a valid program message, because of inconsistency.

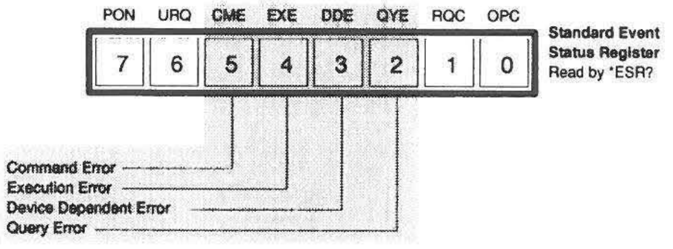
After an execution error, the PM 5781 will normally discard all deferred commands and then continue to process the input stream.

- *Device–Dependent Errors*

The PM 5781 reports its device–specific errors in the DDE bit of the standard event status register. Examples of such an error is over-temperature and calibration data lost. The device–specific errors will not influence the processing of the input stream nor lead to a command, query, or execution error.

Error Reporting

The PM 5781 reports an error in the standard event status register. It summarizes each group of errors in one bit.



11 - 13 Error bits in the Standard Event Status Register

When an error occurs, the PM 5781 places the related error message in the error queue.

Reading the Error Queue

ERROR? Error query

Use the ERROR? query to read errors from the error queue. The error queue is a FIFO buffer (First In First Out). When you read the queue, you will always read the oldest error.

This page is intentionally left blank.

Programming the PM 5781

Introduction

The PM 5781 can be controlled by a computer (controller) via the built in GPIB interface. All functions that can be controlled from the front panel can also be controlled via the bus in a similar way, except for the power switch.

To select functions, you send program message units to the pulse generator. These program message units are similar to the text on the front panel, which makes them easy to remember. For example, the program message unit to select a period of 100 ns is PERIOD 100E-9, and the program message unit to set the high level to 5 V is

HIGHLEVEL 5.0.

The same program message units followed by a question mark (?) cause the pulse generator to output the current setting of that parameter, e.g., HIGHLEVEL? will result in the response HIGHLEVEL 5.0.

NOTE: The characters in a program message unit can be in both upper and lower case, but all responses from the pulse generator will be in UPPER CASE only.

Getting Started

Connecting the Controller

The bus interface connector is on the rear panel of the pulse generator.

Connect the controller via an IEEE-488 cable to the bus connector. If you use IEC-625 cables, an adapter is available, see ordering information in chapter 8, "Specifications".

Giving the Pulse Generator an Address

The pulse generator must have a unique address so that the controller can communicate with it. The default address is 17. You can easily change the address from the front panel using the Auxiliary menu:

- Press the Menu button.
- Select: 5 AUX.
- Select: 1 GPIB.
- Using the keypad, enter an address between 0 and 30 as a decimal value.

Response Message Terminator

The response terminator ends messages from the pulse generator to the controller. Since this instrument is IEEE–488.2 compatible, the response message terminator is always $10_{decimal}$ (new line ↵) and ^EOI. Set your controller to accept this terminator.

The pulse generator is now ready for bus control.

Checking the Communication

To check if the pulse generator and the controller can communicate, run program example 1 in the 'Program Examples' chapter. (The programming example is for a pulse generator with the address 17.)

Programming

Accurate

A normal method of programming is to set the pulse generator to a known status by sending ✱RST, then specify each step of the setup that deviates from this default setting.

Simple

A simpler way of programming the pulse generator is to manually set up the measurement you want from the front panel of the pulse generator. Then let the controller ask the pulse generator how it is set up with the ✱LRN? query. The data the controller gets from the pulse generator can be used to set up the same pulse over and over again.

Fast

You can program the pulse generator so that it can change the output pulse very rapidly by using the internal storage in the generator. Store the complete instrument setting using ✱SAV, or store sequences of commands in a macro. Program the controller to switch between these stored settings.

Other means of speeding up execution include switching off the display with DISPMODE OFF, turning off Error Check, or using TRACK and INCREMENT to change several parameters at once.

Programming Check List

Check that the following steps have been taken to ensure correct programming of the instrument.

1. Do you know the current setting of the pulse generator? If not, send device clear ✱RST to get the default settings.
2. Select triggering.
3. Select pulse mode
4. Select pulse parameters.
5. Enable the outputs.

All functions and commands in the checklist will be explained later.

REMEMBER: You only have to program the changes from the previous setup.

Status Reporting

The PM 5781 defined status reporting is described in chapter 11, "System Functions".

Over- programming

To compensate for the setting inaccuracy of the instrument, you must be able to program parameters to values outside the specified limits.

Example: The specifications of the instrument states that the maximum amplitude is +10 V. The setting inaccuracy of the parameter is ±2%. The result of the setting inaccuracy is that if you program HighLevel as +10.00 V and Lowlevel to 0 V the result may be an amplitude of 9.8 V. To reach the specified 10.00 V you may need to program HiLevel to +10.10 V and Lolevel to –0.1 V.

NOTE: This is not a method to get higher than specified performance; it is just a method to get out to the corners of the specification.

Command Reference

Introduction

This chapter is the reference part of this manual with a complete list of all PM 5781 commands and queries, including all IEEE–488.2 common commands and queries that can be used with the PM 5781 Programmable Pulse Generator.

A short form alphabetical command list with all commands is available in chapter 14.

Pulse Parameters

Period

Period **command/query**

Sets the pulse period to any value between 8 ns and 10 s.

Command Syntax: **Period/ Per** <decimal data>

Where:
<decimal data> = a number between 7.8* • 10^{-9} and 10.

- *Example:* Period 100e-6 ↵

Query Syntax: Period? / Per?

- *Response Format:* [(PERIOD / PER)]<decimal data> ↵

Type of Command: Coupled command**. Coupled to: Delay, Duration, LeadingEdge, TrailingEdge, BurstPeriod, Burst, TriggerSource, TriggerFunction and PulseMode.

---

* Note that this value includes overprogramming range.

** The rules for coupling is available in the 'Appendix' chapter.

Delay **command/query**

Sets the delay between the clock pulse and the output pulse. If double pulse is selected, the first pulse is synchronous with the clock pulse, and the second pulse is generated after the delay set with this command.

Range: 8 ns to 100 ms

Command Syntax: **Delay / Dly** ␣ <decimal data>

Where:

<decimal data> = a number between $7.8^{*} \bullet 10^{-9}$ and 0.1.

• *Example:* Delay ␣ 10E-4 ↵

Query Syntax: Delay? / Dly?

• *Response Format:* [(DELAY / DLY) ␣]<decimal data> ↵

Type of Command: Coupled command**. Coupled to: Period, Duration, LeadingEdge, TrailingEdge, Trigger Source, Trigger Function and Pulse Mode.

* Note that this value includes overprogramming range.

** The rules for coupling is available in the 'Appendix' chapter.

Duration

Duration **command/query**

Sets the duration of the output pulse within the range 4 ns to 100 ms.

Command Syntax: **Duration / Dur / Width** <decimal data>

Where:

<decimal data> = a number between $3.9^{*} \bullet 10^{-9}$ and 0.1.

• *Example:* Duration 100E-3 ↵

Query Syntax: Duration? / Dur? / Width?

• *Response Format:* [(DURATION / DUR)]<decimal data> ↵

Type of Command: Coupled command**. Coupled to: Period, Delay, LeadingEdge, TrailingEdge, Trigger Source, Trigger Function, and Pulse Mode.

* Note that this value includes overprogramming range.

** The rules for coupling is available in the 'Appendix' chapter.

Leading Edge **command/query**

Sets the transition time, from 10% to 90% of the pulse amplitude, to a value in the range 2 ns to 100 ms.

Command Syntax: **LeadingEdge / LeE** <decimal data>

Where:

<decimal data> = a number between $1.9^{*} \bullet 10^{-9}$ and 0.1.
The <decimal data> must be within the same range as the <decimal data> for the Trailing Edge.

- *Example:* Leadingedge 10e-7

Query Syntax: LeadingEdge? / LeE?

- *Response Format:* [(LEADINGEDGE / LEE)]<decimal data> ↵
- *Example:* LEADINGEDGE 1.0E-6 ↵

Type of Command: Coupled command. Coupled to: Period, Delay, Duration, TrailingEdge, TriggerSource, TriggerFunction, and PulseMode.

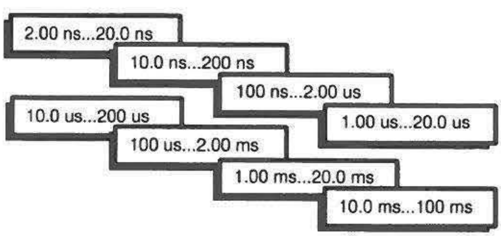


Fig. 13-1 Transition time ranges

---

* Note that this value includes overprogramming range.

Trailing Edge

Trailing Edge **command/query**

Sets the transition time from 90% to 10% of the pulse amplitude.

Command Syntax: **TrailingEdge / TrE** <decimal data>
Where:
<decimal data> = a number between $1.9^{*} \bullet 10^{-9}$ and 0.1. The <decimal data> must be within the same range as the <decimal data> for the Leading Edge.

- *Example:* TrailingEdge 10e-7 ↵

Query Syntax: TrailingEdge? / TrE?

- *Response Format:* [(TRAILINGEDGE / TRE)]<decimal data> ↵
- *Example:* TRAILINGEDGE 1.0E-8 ↵

Type of Command: Coupled command. Coupled to: Period, Delay, Duration, LeadingEdge, TriggerSource, TriggerFunction, and PulseMode.

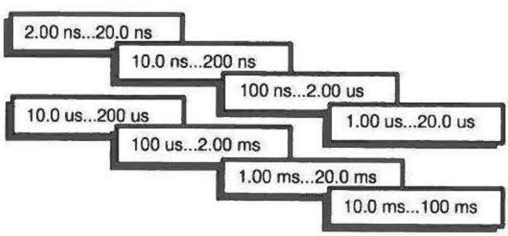


Fig. 13-2 Transition time ranges

---

* Note that this value includes overprogramming range.

Burst Period **command/query**

Sets the time between the starting point of two consecutive bursts. Range 1µs to 4000 s.

Command Syntax: **BurstPeriod / BrP** <decimal data>

Where:
<decimal data> = a number between $1 \bullet 10^{-6}$ and 4000.

- *Example:* Burstperiod 100e-2 ↵

Query Syntax: BurstPeriod? / BrP?

- *Response Format:* [(BURSTPERIOD / BRP)]<decimal data> ↵

Type of Command: Coupled command*. Coupled to: Period, Burst, TriggerSource, and TriggerFunction.

---

* The rules for coupling is available in the 'Appendix' chapter.

Burst Pulses

Burst Pulses **command/query**

Sets the number of pulses in each burst to between 0 and 9999.

Command Syntax: **Burst / Bur** <decimal data>

Where:

<decimal data.> = a number between 0 and 9999.

- *Example:* Burst 5 ↵

Query Syntax: Burst? / Bur?

Response Format: [(BURST / BUR)]<decimal data> ↵

Where:
<decimal data> = an integer.

- *Type of Command:* Coupled command*. Coupled to: Period, BurstPeriod TriggerSource, and TriggerFunction..

---

* The rules for coupling is available in the 'Appendix' chapter.

High Level **command/query**

Sets the high level of the output pulse to any value between –9.8 V and +10 V.

Command Syntax: **HighLevel / HiLvl / HiL** <decimal data>

Where:
<decimal data> = a number between –10.0* and +10.1*.

- *Example:* Highlevel 2.00 ↵

Query Syntax: HighLevel? / HiLvl? / HiL?

- *Response Format:* [(HIGHLEVEL / HIL)]<decimal data> ↵

Type of Command: Coupled command**. Coupled to: LowLevel and Offset_B.

---

* Note that these values includes overprogramming range.

** The rules for coupling is available in the 'Appendix' chapter.

Low Level

Low Level **command/query**

Sets the low level of the output pulse to any value between -10 V and +9.8 V .

Command Syntax: **LowLevel / LoLvl / LoL** <decimal data>

Where:

<decimal data> is a number between –10.1* and +10.0*.

• *Example:* Lowlevel 2.00 ↵

Query Syntax: LowLevel? / LoLvl? / LoL?

• *Response Format:* [(LOWLEVEL / LOL)]<decimal data> ↵

Type of Command: Coupled command**. Coupled to: HighLevel and Offset_B.

* Note that these values includes overprogramming range.

** The rules for coupling is available in the 'Appendix' chapter.

Offset_B **command/query**

Adds an offset voltage between –19.8 V and +19.8 V to the HighLevel and LowLevel of output B. The offset is referenced to the HighLevel and LowLevel of output A.

Command Syntax: **Offset_B / OfB** <decimal data>

Where:

<decimal data> = a number between –19.9* and +19.9*.

Note that the underline character is part of the longform header.

- *Example:* Offset_B 0.00 ↵

Query Syntax: Offset_B? / OfB?

- *Response Format:* [(OFFSET_B / OFB)]<decimal data> ↵
- *Example:* OFFSET_B 5.1 ↵

Type of Command: Coupled command**. Coupled to: HighLevel and LowLevel.

---

* Note that these values includes overprogramming range.

** The rules for coupling is available in the 'Appendix' chapter.

Tracking

Tracking **command/query**

Selects which parameters shall be tracked to each other. Time parameters are tracked to time parameters; they are kept in the ratio that they have when the command is received. High and low level are tracked with a constant difference. Offset_B is not trackable.

Command Syntax: **Tracking / Track** Off I(On,<parameter> {,<parameter>})

Where:

<parameter> = (Period / Per I Delay / Dly I Duration / Dur I LeadingEdge / LeE I TrailingEdge / TrE I HighLevel / HiLvl / HiL I LowLevel / LoLvl / LoL)

NOTE: You can switch off all tracking with the command TRACKING OFF ↵, but you cannot switch on tracking unless you specify the parameters to track.

- *Example:* Tracking on , leadingedge , trailingedge , highlevel , lowlevel ↵

Query Syntax: Tracking? / Track?

- *Response Format:* [(TRACKING / TRACK)]OFF I (ON , <parameter> { , <parameter> } ↵

Where:

<parameter> = as specified above.

Type of Command: Executing command. Forces execution of all previously deferred commands.

Set Increment **command/query**

Sets parameter to be changed, whether the change shall be absolute or relative, and the absolute/relative change that shall be made when INCREMENT command is received. Relative change means as a part (e.g., 0.01) of the current value of the selected parameter. If the selected parameter is tracked, the other tracked parameters are changed too. A negative value of <dec. data> means that the value will be decremented.

To avoid the delay between receiving the INCREMENT command, and the setting of incremented parameters, SETINCREMENT can be sent without data. Then the pulse generator will precalculate the next incremented value (with the previously set increment) without actually setting any parameters.

Command Syntax: **SetIncrement / SetInc** [<parameter> , (Absolute / Abs I Relative / Rel) , <dec. data>]
Where:
<parameter> = (Period / Per I Delay / Dly I Duration / Dur I LeadingEdge / LeE I TrailingEdge / TrE I BurstPeriod / BrP I Burst / Bur I HighLevel / HiLvl / HiL I LowLevel / LoLvl / LoL I Offset_B / OfB)
<dec. data> = The size of the increment.

- *Example:* Setincrement highlevel , absolute , 0.5 ↵

Query Syntax: SetIncrement? / SetInc?

- *Response Format:* [(SETINCREMENT / SETINC)] <parameter> , (ABS I REL) , <dec. data>

Type of Command: Executing command. Forces execution of all previously deferred commands.

Increment **command**

Increments the parameter that is selected with the SET-INCREMENT command. To obtain minimum delay, the next value is calculated after the hardware is updated. If other commands are received between increment commands, the calculation is done before the execution.

Command Syntax: **Increment / Inc**

- *Example:* Increment ↵

NOTE:

The <program header> SETINCREMENT sent without any <program data> forces the generator to precalculate the settings it should use if a new INCREMENT is received.

Type of Command: Executing command. Forces execution of all previously deferred commands.

Autoset **command**

Corrects the pulse generator setting, using the selected parameter as reference. Locked parameters are not changed; tracking is not changed.

This command functions in the same way as the AUTO SET key on the front panel. It can be used for instance if you want a specific period time and the other parameters are of no interest. Then send PER 10E-3 ; AUTOSET PER↵. Make it a rule always to make a query about the reference parameter and compare it with the desired value to verify that autoset has not changed it.

Command Syntax: **Autoset** ␣ <parameter>

Where:

<parameter> = (Period / Per | Delay / Dly | Duration / Dur | LeadingEdge / LeE | TrailingEdge / TrEl BurstPeriod / BrP | Burst / Bur | HighLevel / HiLvl / HiLl LowLevel / LoLvl / LoL | Offset_B / OfB)

- *Example:* Autoset ␣ delay ↵

Type of Command: Executing command. Forces execution of all previously deferred commands.

Locking

Locking **command/query**

Prevents unintentional change of pulse parameters.

Command Syntax: **Locking / Lock** Off|(On,<parameter> {,<parameter>})

Where:
<parameter> = (Period / Per | Delay / Dly | Duration / Dur | LeadingEdge / LeE | TrailingEdge / TrE | BurstPeriod / BrP | Burst / Bur | HighLevel / HiLvl / HiL | LowLevel / LoLvl / LoL | Offset_B / OfB)

- *Example:* Locking On , Leadingedge , Trailingedge ↵

Query Syntax: Locking? / Lock?

- *Response Format:* [(LOCKING / LOCK)] OFF | ON(,<parameter>{ , <parameter>})↵

Type of Command: Executing command. Forces execution of all previously deferred commands.

Limiter **command/query**

Enables/disables the limiter function.

Command Syntax: **Limiter / Lim** ˽ (On | Off)

- *Example:* Limiter ˽ Off ↵

Query Syntax: Limiter? / Lim?

- *Response Format:* [LIMITER/LIM ˽] (ON | OFF)

Type of Command: Executing command. Forces execution of all previously deferred commands.

See Also: Voltlimits.

Volt Limits

Volt Limits **command/query**

Sets min. and max. limits for the output voltages of channel A and B to any value between −10 V and +10 V.

Command Syntax: **VoltLimits / VLim** <dec. data> , <dec. data>

Where:
<dec. data>,<dec. data> = <lower limit> , <upper limit> in the range −10 to +10.

- *Example:* Voltlimits −1.2 , 5.2 ↵

Query Syntax: VoltLimits? / VLim?

- *Response Format:* [VOLTLIMITS/VLIM] <dec. data> , <dec.data> ↵

Type of Command: Executing command. Forces execution of all previously deferred commands.

See Also: Limiter.

Period Control **command/query**

Disables/enables the period control loop.

Command Syntax: **PeriodCtrl / PerCtrl** ˽ (On | Off)

- *Example:* Periodctrl ˽ on ↵

Query Syntax: PeriodCtrl? / PerCtrl?

- *Response Format:* [PERIODCTRL/PERCTRL ˽] (ON | OFF)

Type of Command: Deferred command.

This page is intentionally left blank.

Trigger Commands

Trigger Source

Trigger Source **command/query**

Selects internal, external, or manual trigger source.
NOTE: Use Manual both when the generator should be triggered from the front panel key and via the Mantrig or ManGate command on the bus.

Command Syntax: **TrigSource / TrgSrc** (Internal / Int | External / Ext | Manual / Man)

- *Example:* Trigsource Internal ↵

Query Syntax: TrigSource? / TrgSrc?

- *Response Format:* [TRIGSOURCE / TRGSRC](INT | EXT | MAN) ↵

Type of Command: Coupled command*. Coupled to: TrigFunction, PulseMode, Period, Delay, Duration, LeadingEdge, TrailingEdge, BurstPeriod, and Burst.

---

* The rules for coupling is available in the 'Appendix' chapter.

Trigger Function **command/query**

Selects:

- Triggered by the trigger source.
- Gated by the trigger source.
- Triggers the burst generator with the trigger source.
- Duration controlled manually or with the external input

Command Syntax: **TrigFunction / TrgFnc** (Trig / Trg | Gate | Burst / Bur | Duration / Dur)

- *Example:* TrigFunction Trig ↵

Query Syntax: TrigFunction? / TrgFnc?

- *Response Format:* [TRIGFUNCTION/TRGFNC](TRG|GATE|BUR|DUR)↵

Type of Command: Coupled command*. Coupled to: TrigSource, PulseMode, Period, Delay, Duration, LeadingEdge, TrailingEdge, BurstPeriod, and Burst.

* The rules for coupling is available in the 'Appendix' chapter.

Manual Trigger

Manual Trigger **command**

Triggers the pulse generator when Trigger Source is set to Manual, and Trigger Function to Trig or Burst. When the Trigger Function is set to Gate or Duration, ManTrig toggles the gate on the first time and off the next time you send it. This command has the same function as pressing the MANUAL key on the front panel of the generator.

This command is default command for *DDT.

Command Syntax: **ManualTrig/ ManTrg**

• *Example:* ManualTrig ↵

Type of Command: Executing command. Forces execution of all previously deferred commands.

Manual Gate **command/query**

Opens and closes the gate when Trigger Source is set to Manual and Trigger Function to Gate or Duration. This command has the same function as pressing and holding the MANUAL key on the front panel of the generator.

Command Syntax: **ManualGate / ManGate** (Close|Open)

• *Example:* ManGate Open ↵

Query Syntax: ManualGate? / ManGate?

Response Format:

[MANUALGATE/MANGATE] (CLOSE|OPEN) ↵

Type of Command: Executing command. Forces execution of all previously deferred commands.

Define Device Trigger

Define Device Trigger **command/query**

Stores a command sequence which is executed when a group execute trigger (GET) message or a ✱TRG command is received.

Default command: MANTRG

Command Syntax: **✱DDT** <blockdata>

Where:

<blockdata> = an arbitrary block data containing the sequence of <program message units> which are executed when a GET or ✱TRG is received.

- *Example:* ✱DDT #14 NEXT ↵

Query Syntax: ✱DDT?

- *Response Format:* <blockdata> ↵

- *Example:* #14 NEXT ↵

Type of Command: Deferred command. Will not be executed until a ↵ or an executing command.

Trigger **command**

The Trigger command is the device-specific equivalent of the IEEE–488.1 defined Group Execute Trigger, and has exactly the same effect as a GET when received, parsed, and executed by the device.

Command Syntax: ***TRG**

Type of Command: Executing command. Forces execution of all previously deferred commands.

This page is intentionally left blank.

Input Commands

Trigger Level

Trigger Level **command/query**

Sets the trigger level of the external input to any value between –5 V and +5 V.

Command Syntax: **TrigLevel / TrgLvl** (<decimal data> I TTL I ECL I CMOS)

Where:

<decimal data> = a number between –5 and +5.

- *Example:* Triglevel 1.00 ↵

Query Syntax: TrigLevel? / TrgLvl?

- *Response Format:* [TRIGLEVEL / TRGLVL]<decimal data> ↵

Note that the pulse generator always responds with the decimal trigger level even if TTL or ECL has been used to set up the trigger level.

- *Example:* TRIGLEVEL 0.75 ↵

Type of Command: Deferred command. Will not be executed until a ↵ or an executing command is received.

Trigger Slope — command/query

Selects between triggering on the positive or negative going slope of the input signal.

Command Syntax: **TrigSlope / TrgSlp** (Pos | Neg)

- *Example:* Trigslope pos ↵

Query Syntax: TrigSlope? / TrgSlp?

- *Response Format:* [TRIGSLOPE / TRGSLP](NEG | POS) ↵

Type of Command: Deferred command. Will not be executed until a ↵ or an executing command is received.

Input Impedance

Input Impedance **command/query**

Selects the input impedance of the external input.

Command Syntax: **Impedance / Imp** ␣ (High / Hi I Low / Lo)

- *Example:* Impedance ␣ hi ↵

Query Syntax: Impedance? / Imp?

- *Response Format:* [IMPEDANCE / IMP ␣](HIGH I LOW)

Type of Command: Deferred command. Will not be executed until a ↵ or an executing command is received.

Pulse Mode

Pulse Mode

Pulse Mode **command/query**

Command Syntax: **PulseMode / PlsMod** (Square / Sqr | Single / Sgl | Double / Dbl)

• *Example:* Pulsemode sqr ↵

Query Syntax: PulseMode? / PlsMod?

• *Response Format:* [PULSEMODE / PLSMOD](SQR | SGL | DBL) ↵

Type of Command: Coupled command*. Coupled to: TrigSource, TrigFunction, Period, Delay, Duration, LeadingEdge, and TrailingEdge.

---

* The rules for coupling is available in the 'Appendix' chapter.

Complementary

command/query

Switches between normal and complementary waveform on both A and B outputs.

Command Syntax: **Complement / Compl** ˽ (Off I On)

• *Example:* Complement ˽ off ↵

Query Syntax: Complement? / Compl?

• *Response Format:* [COMPLEMENT / COMPL ˽] (ON I OFF)↵

Type of Command: Deferred command. Will not be executed until a ↵ or an executing command is received.

This page is intentionally left blank.

Output Commands

Output Disable

Output Disable **command/query**

Enables or disables output A, B, and Clock.

Command Syntax: **Output / Outp** (Disable / Off { , (A | B | Clock / Clk) }) | (Enable / On , (A | B | Clock / Clk) { , (A | B | Clock / Clk) })

NOTE: You can disable all outputs with the command OUTPUT OFF ↵, but you cannot enable all outputs unless you specify each of them as in the example below. This ensures that only the intended outputs are switched on.

- *Example:* Output enable , A , B , clock ↵

This example enables all outputs.

Query Syntax: Output? / Outp?

- *Response Format:* [OUTPUT / OUTP] OFF| (ON , (A | B | CLK) { , (A | B | CLK) }) ↵

The <response message> contains only the enabled outputs or OFF if no output is enabled.

- *Example:* OUTPUT ON , A , B ↵

Type of Command: Deferred command. Will not be executed until a ↵ or an executing command is received.

Clock Output Level

command/query

Sets the output level of the clock output to TTL levels (0 and +2.5 V into 50 Ω) or ECL levels (–1.8 V and –0.8 V into 50 Ω)

Command Syntax: **ClockLevel / ClkLvl** ˽(TTL | ECL)

• *Example:* Clocklevel˽ttl ↵

Query Syntax: ClockLevel? / ClkLvl?

• *Response Format:* [CLOCKLEVEL / CLKLVL˽](TTL | ECL) ↵

Type of Command: Deferred command. Will not be executed until a ↵ or an executing command is received.

This page is intentionally left blank.

Saved Settings

Save

Save **command**

Saves the current settings of the instrument in an internal nonvolatile memory. There are 49 memory locations available.

Command Syntax: ***SAV** <decimal data>

Where:

<decimal data> = a number between 1 and 49.

• *Example:* *SAV 10 ↵

Type of Command: Executing command. Forces execution of all previously deferred commands.

Recall **command**

Recalls one of the up to 49 previously stored complete instrument settings from the internal nonvolatile memory of the instrument. Number 0 is a memory location containing the default settings.

- *Command Syntax:* ***RCL** <decimal data>

Where:

<decimal data> = a number between 0 and 49.

- *Example:* *RCL 10 ↵

Type of Command: Immediate command. All previously deferred commands are discarded.

Reset

Reset **command**

The Reset command performs a device reset. It is the third level of reset in a 3-level reset strategy, and it primarily affects the device functions, not the IEEE–488 bus.

The instrument settings will return to the default settings, as when recalling memory 0 by executing *RCL 0, macros are disabled and the pulse-generator is prepared to start new operations.

Command Syntax: ***RST**

- *Example:* *RST ↵

Type of Command: Immediate command. All previously deferred commands are discarded.

See Also: Default settings on page 5-49.

Learn Device Setup — query

Learn Device Setup Query. Causes a response message that is directly resendable and will place the instrument in the state it was when the ✱LRN? query was made.

Query Syntax: **✱LRN?**

- *Response Format:* <Command> {, <Command>}↵

  Where:

  <Command> = the 'PM 5781 Commands' normally used for setting up the instrument.

- *Example:* *RST;PER 2.01E-04;DLY 2.5E-07;DUR 1.0E-07;LEE 2.0E-09 ;TRE 2.0E-09;BRP 1.0E-04;BUR 5;HIL 4.0;LOL -4.0 ; OFB 0.0; TRGLVL 1.0; TRGSRC INT ; TRGFNC TRIG ; PLSMOD SQR ; TRGSLP POS ; IMP HIGH ; CLKLVL TTL;COMPL OFF; OUTP OFF ; TRACK ON , LEE ; LOCK OFF ; DISPMODE ALL;DISPFMT EXP ; BEEPER ON ; PERCTRL OFF ; VLIM -5.0,5.0 ; LIM OFF ; ERRCHK ON ; DEBUG OFF ; RESHDR SHORT ; SETINC PER,REL,1.0E-02 ; *DDT #16MANTRG

Type of Command: Executing command. Forces execution of all previously deferred commands.

Next

Next | **command**

Recalls the next setting in the used sequence.

Command Syntax: **Next**

- *Example:* NEXT ↵

Type of Command: Immediate command. All previously deferred commands are discarded.

See Also: Previous, UseSequence, and DefSequence.

Previous command

Recalls the previous setting in the used sequence.

Command Syntax: **Previous / Prev**

- *Example:* Previous ↵

Type of Command: Immediate command. All previously deferred commands are discarded.

See Also: Next, UseSequence, and DefSequence.

Use Sequence

Use Sequence **command/query**

Selects the sequence that will be recalled when 'next' and 'previous' are received. 'UseSequence Memory' means that all defined settings, except the default setting, are recalled in sequence. Empty memory locations are not recalled. If an empty sequence (A, B, C or D) is selected by 'usesequence', execution error will occur when that sequence is recalled with 'next' or 'previous'.

Command Syntax: **UseSequence / UseSeq**0 Memory / Mem I A I B I C I D

• *Example:* Usesequence C ↵

Query Syntax: UseSequence? / UseSeq?

• *Response Format:* [USESEQUENCE / USESEQ](MEM I A I B I C I D)

Type of Command: Immediate command. Executed immediately without executing previous commands.

See also: Previous, Next, and DefSequence.

Define Sequence **command**

Defines the settings to be recalled in sequence (using Next and Previous). There are four user–definable sequences, each containing up to 10 settings. The string "Empty" clears the selected sequence.

Command Syntax: **DefSequence / DefSeq** (A | B | C | D) , Empty | (<dec. data> , <dec. data> { , <dec. data>})

Where:

<dec.data> = the number of a stored setting.

• *Example:* Defsequence A , 1 , 23 , 12 , 5 ↵

Query Syntax: DefSequence? / DefSeq? (A | B | C | D) { , (A | B | C | D) }

• *Response Format:* [DEFSEQUENCE / DEFSEQ](A | B | C | D)(, EMPTY | { ,<dec. data> })

{ ; [DEFSEQUENCE / DEFSEQ](A | B | C | D) ,

(EMPTY , { <dec. data> }) }↵

Type of Command: Immediate command. Executed immediately without executing previous commands.

See also: Previous, Next and UseSequence.

Binary Program

Binary Program

Binary program query reads the current setting of the pulse generator and sends it to the controller in binary form*. This data can be stored and sent back to the pulse generator to get the same setting again later. This command has the same function as the *LRN common command with the exception that the block data cannot be understood and edited in the controller. The Binprog data has the same data format as the Memprog data so, the header can be changed and a memory location number can be added to the data (in the controller) in order to convert the data to MemProg data, see next page.

BinProg is much faster than *LRN.

Command Syntax: **BinProg** <block data>

Where: <block data> is the instrument setting information previously retrieved via the BinProg? Query.

Query Syntax: BinProg?

- *Response Format:* [BinProg] <block data>

Type of Command: Executing command. Forces execution of all previously deferred commands.

See Also: Block data in the 'System functions' chapter.

* Binary data can be difficult to store in the controller because it may contain data that the controller interprets as 'end of file', etc. Consult your controller manual for the correct procedure for storing binary data.

Memory Program **command/query**

MemProg? query reads one of the settings (saved with ✱SAV) from the internal memory of the pulse generator and sends it to the controller in binary form*. This data can be stored in the controller and sent back to the pulse generator to get the same setting again later. The Binary data cannot be understood or edited in the controller.

The Memprog data has the same data format as the Binprog data so, the header can be changed and the memory location number deleted from the data (in the controller) in order to convert the data to BinProg data, see previous page.

Command Syntax: **MemProg** <dec. data> , <block data>

Where: <dec. data> is the memory location you want to write to, and <block data> is the instrument setting information previously retrieved with the MemProg? query.

Query Syntax: MemProg? <dec. data>

- *Response Format:* [MEMPROG] <dec. data> , <block data>

Type of Command: Immediate command. Executed immediately without executing previous commands.

See also: Block data in the 'System functions' chapter.

* Binary data can be difficult to store in the controller as it may well contain data that the controller interprets as 'end of file' etc. Consult your controller manual for correct procedure for storing binary data.

Delete Memory

Delete Memory **command**

Deletes the selected settings from the nonvolatile memory in the instrument. Up to 10 selected memory locations, or all memory locations, can be deleted in the same command.

Command Syntax: **DeleteMemory / DelMem** (<decimal data>|All) { , <decimal data> }

- *Example:* DeleteMemory 10 , 14 , 40 ↵

Type of Command: Immediate command. Executed immediately without executing previous commands.

Macros

Define Macro

Define Macro **command**

Allows the programmer to assign a sequence of one or more program message units to a macro label. The sequence is executed when the macro label is received as a command or query. (If macros are disabled, the execution of this command is fast, but if macros are enabled, the execution time for this command is longer.)

Command Syntax: ***DMC** <string> , <blockdata> I <string>

Where:

<String> = the macro label. (String data must be surrounded by " " or ' ' as in the example below.)

<blockdata> I <string> = the commands to be executed when the macro label is received, both blockdata and string data formats can be used.

- *Example:* *DMC "MyPer" , # 224 Per $1 ; LeE 1E-6 ; TrE 1E-6 ↵

This example defines a macro MyPer which takes one argument, i.e. the period. It also sets the edges to fixed values.

Type of Command: Immediate command. Executed immediately without executing previous commands.

Enable Macros **command/query**

This command enables and disables expansion and execution of macros. If macros are disabled, the instrument will not recognize a macro although it is defined in the instrument. (The Enable Macro command takes a long time to execute.)

Command Syntax: **✱EMC** <dec. data>

Where:

<dec. data> = in a number in the range -32767 to +32767. A value which rounds to 0 turns off macro execution. Any other value turns macro execution on.

- *Example:* ✱EMC 1 ↵

Enable macro expansion and execution.

Query Syntax: ✱EMC?

- *Response Format:* 0 | 1 ↵

1 indicates that macro expansion is enabled.

0 indicates that macro expansion is disabled.

Type of Command: Immediate command. Executed immediately without executing previous commands.

Get Macro

Get Macro **query**

This command results in a response that is the current definition for the given macro label.

Query Syntax: **✱GMC?** <string>

Where:

<string> = the label of the macro for which you want to see the definition. (String data must be surrounded by " " or ' ' as in the example below.)

- *Response Format:* <block data>
- *Example:* ✱GMC? "MyPer"↵

gives the following response:

"MYPER", #224 PER $1 ; LEE 1E-6 ; TRE 1E-6 ↵

Type of Command: Executing command. Forces execution of all previously deferred commands.

Learn Macro **query**

Makes the instrument output a list of string data elements, containing all macro labels defined in the instrument.

Query Syntax: ***LMC?**

- *Response Format:* <string> { ,<string> }

Where:

<string> = a Macro label. (String data must be surrounded by " " or ' ' as in the example below.)

- *Example:* *LMC? ↵

may give the following response:

"MYPER" , "STARTPER" , "TESTPER" ↵

Type of Command: Executing command. Forces execution of all previously deferred commands.

Purge Macros

Purge Macros

command

Removes all macro definitions. (The Purge Macro command takes a long time to execute.)

Command Syntax: **✱PMC**

- *Example:* ✱PMC ↵

Type of Command: Immediate command. Executed immediately without executing previous commands.

Help Commands

Help query

This command gives a brief help text, mentioning the other available queries in the help category.

Query Syntax: **Help?**

- *Response Format:* [HELP]<string>
- *Example:* Use 'CODES?' to get a list of all commands.

  Use 'SYNTAX? "and name"' to get the syntax of one command.

  Use 'ERROR?' to read out the number and explanation of the last error.

  Use 'DEBUG ON' to turn on PM5781's display of received messages.

Type of Command: Executing command. Forces execution of all previously deferred commands.

Codes **query**

This query gives a response containing all the user commands understood by the instrument. If the long response header format has been chosen, the returned list will contain long form; otherwise, short form will be used.

Query Syntax: **Codes?**

- *Response Format:* [CODES] <string>
- *Example:*

"*CAL?	*CLS	*DDT	*DMC
*EMC	*ESE	*ESR?	*GMC?
*IDN?	*LMC?	*LRN?	*OPC
*OPT?	*PMC	*PSC	*PUD
*RCL	*RST	*SAV	*SRE
*STB?	*TRG	*TST?	*WAI
HELP?	CODES?	SYNTAX?	BINPROG
MEMPROG	MEMPROG?	DEBUG	DEFTEST
DISPMODE	ERROR?	ERRORDISP	RESHDR
TEXT	UNPROTECT	SETCODE	SOEV?
SOC?	SOEN	SOPT	SONT
SQC?	SQEN	SQPT	SQNT
ONTIME	CALDEFAULT	CALTEMP	CALDATA
CALSTATUS?	CALADJUST?	CALREPORT?	CALERROR?
CALTIME?	PER	DLY	DUR
LEE	TRE	BRP	BUR
HIL	LOL	OFB	TRGLVL
TRGSRC	TRGFNC	PLSMOD	TRGSLP
IMP	CLKLVL	COMPL	OUTP
MANGATE	LOCK	TRACK	BEEPER

Codes

LIM	VLIM	DISPFMT	DEFSEQ
USESEQ	ERRCHK	SETINC	INC
PERCTRL	CUR	MANTRG	DELMEM
NEXT	PREV	AUTOSET	BEEP
DEFSEQ?	PARLIM?	TEMP?	WHO?"

Type of Command: Executing command. Forces execution of all previously deferred commands.

Syntax **query**

This query gives a response containing the syntax of the command specified in the query. The response is in simplified EBNF format.

Query Syntax: **Syntax?** <string>

Where:

<string> = the command for which you want to know the syntax. (String data must be surrounded by " " or ' ' as in the example below.)

Response Format: [SYNTAX]<string>

• *Example:* Syntax? 'dispmode' ↵

gives the response:

```
SYNTAX "DISPMODE OFF
                 | STATE
                 | ALL"
```

Type of Command: Executing command. Forces execution of all previously deferred commands.

Error

Error **query**

This query reads out an error message from a FIFO queue (First In First Out) containing up to 64 messages. The error message contains an error number and an explanation.

Query Syntax: **Error?**

- *Response Format:* [ERROR]<decimal data> , <string> ↵

  Where:

  <decimal data> = the error number.

  <string> = a string containing the explanation of the error number.

- *Example:* ERROR 20021 , "INVALID MESSAGE HEADER FOUND" ↵

Type of Command: Executing command. Forces execution of all previously deferred commands.

See also: Chapter 16 "Error Messages".

Error Display **command/query**

Selects if GPIB error messages should be shown on the display of the instrument or not. If Errordisp is set to ON, the instrument shows the error message and beeps each time an error is detected. The error code and explanation will disappear as soon as the instrument wants to display other data, a button is pressed (if not in remote mode) or a new message is received.

Command Syntax: **ErrorDisp** ON/OFF

Query Syntax: ErrorDisp?

- *Response Format:* [ERRORDISP] ON/OFF

Type of Command: Deferred command. Will not be executed until a ↵ or an executing command is received.

Debug

Debug **command/query**

The characters that are received on the bus are echoed to the display. Used to check that communication is OK.

Command Syntax: **Debug** (On | Off)

- *Example:* Debug On

Query Syntax: Debug?

- *Response Format:* [DEBUG](ON | OFF)
- *Type of Command:* Deferred command. Will not be executed until a ↵ or an executing command is received.

Parameter Limits **query**

Returns the present min. and max. limits for the selected parameter.

Query Syntax: **ParLimits? / ParLim?** <parameter>

Where:

<parameter> = (Period / Per I Delay / Dly I Duration / Dur I LeadingEdge / LeE I TrailingEdge / TrE I BurstPeriod / BrP I Burst / Bur I HighLevel / HiLvl / HiLl LowLevel / LoLvl / LoL I Offset_B / OfB

• *Response Format:* [PARLIMITS / PARLIM]<dec. data lower limit>, <dec. data upper limit> ↵

Type of Command: Executing command. Forces execution of all previously deferred commands.

This page is intentionally left blank.

System Commands

Display Mode

Display Mode **command/query**

Selects what the front panel should indicate:

ALL will show all the values in the different displays.

STATE means that the central display will show FRONT PANEL DISABLED, but all other indicators and displays will work.

OFF means that the central display will show FRONT PANEL DISABLED, and all other indicators and displays except bus status are blank.

Command Syntax: **DispMode** ␣ (All | State | Off)

• *Example:* Dispmode ␣ all ↵

Query Syntax: DispMode?

• *Response Format:* [DISPMODE ␣](ALL | STATE | OFF)↵

Type of Command: Deferred command. Will not be executed until a ↵ or an executing command is received.

Display Format — command/query

Selects exponent or prefix+unit representation of parameter values in the display.

Command Syntax: **DispFormat / DispFmt** (Exp / Exponent I Unit)

• *Example:* Dispformat exponent ↲

Query Syntax: DispFormat? / DispFmt?

• *Response Format:* [DISPFORMAT / DISPFMT](EXP I UNIT) ↲

Type of Command: Deferred command. Will not be executed until a ↲ or an executing command is received.

Text

Text **command**

Writes a text on the pulse generator display (max 80 characters). The text is removed either when a key is pressed, or when a new bus command is received, or when TEXT OFF is received.

Command Syntax: **Text** (<string> | Off)

Where:

<string> = string data that must be surrounded by " " or ' ' as in the example below.

• *Example:* Text "PM 5781 is a fantastic pulse Generator!" ↵

Type of Command: Immediate command. Executed immediately without executing previous commands.

Cursor **command/query**

Moves the cursor to the selected parameter.

Command Syntax: **Cursor / Cur** <parameter>

Where:

<parameter> = (Period / PerI Delay / Dly I Duration / Dur I LeadingEdge / LeE I TrailingEdge / TrE I BurstPeriod / BrP I Burst / Bur I HighLevel / HiLvl / HiL I LowLevel / LoLvl / LoL I Offset_B / OfB)

• *Example:* Cursor Period ↵

Query Syntax: Cursor? / Cur?

• *Response Format:* [CURSOR / CUR] <parameter> ↵

Where:

<parameter> = as above.

Type of Command: Immediate command. Executed immediately without executing previous commands.

Beeper **command/query**

Enables/Disables the beeper.

Command Syntax: **Beeper** (On | Off)

- *Example:* Beeper On

Query Syntax: Beeper?

- *Response Format:* [BEEPER](ON | OFF) ↵

Type of Command: Deferred command. Will not be executed until a ↵ or an executing command is received.

Beep **command**

Makes the pulse generator beep, if beeper is enabled.

Command Syntax: **Beep / Bell**

Type of Command: Immediate command. Executed immediately without executing previous commands.

Error Check

Error Check **command/query**

Used for maximum speed of already debugged programs. Preferably used together with DISPMODE OFF or DISPMODE STATE. The setting is checked when a save command is received. If an error is found when the error check is turned on (power on, front panel reset or ERRORCHECK ON), the pulse generator is set to the default setting.

Command Syntax: **ErrorCheck / ErrChk** (On| Off)

• *Example:* Errorcheck on ↵

Query Syntax: ErrorCheck? / ErrChk?

• *Response Format:* [ERRORCHECK](ON | OFF)

Type of Command: Executing command. Forces execution of all previously deferred commands.

Response Header **command/query**

Selects long, short, or no header for the responses on the bus. Note that commands can always be sent as either short or long headers, regardless of the response header selection.

Command Syntax: **ResHdr** (None / Nol Short I Long)

- *Example:* ResHdr Short ↵

Query Syntax: ResHdr?

- *Response Format:* [RESHDR] (NONE I SHORT I LONG)

- *Example:*

Long Headers	Short Headers
Period	Per
Duration	Dur
LeadingEdge	LeE
HighLevel	HiL

NOTE: Mnemonics, like HiLvl for HighLevel, can be used as headers in commands, but they are not used in response headers.

Type of Command: Deferred command. Will not be executed until a ↵ or an executing command is received.

Identify

Identify **query**

Identification query. Answered with three fields containing manufacturer, model, and firmware revision.

Query Syntax: ***IDN?**

• *Response Format:* PHILIPS,PM5781,<S>,<F>

Where:

<S> = Serial No. This function is not supported by PM 5781, it will always output a zero.

<F> = Firmware revision levels. <F> consists of two or three revision levels separated by a stroke. Main program/GPIB program/Internal calibration program. The internal calibration revision level is only available if an Internal Calibration unit, PM 9579 is installed.

• *Example:* PHILIPS, PM5781, 0, MAIN V1.05 28 March 1990 / GPIB V1.02 14 March 1990 / INTCAL V1.00 5 April 1990

Type of Command: Executing command. Forces execution of all previously deferred commands.

Option — query

Option identification query. Answered with one data field for each reportable option. Absent options are indicated by a zero.

Query Syntax: ***OPT?**

• *Response Format:* <DATA> { , <DATA> }

Where:
<DATA> = the name of the option.
For example: <DATA> = PM 9579 means an enhanced accuracy board is fitted. <DATA> = 0 means no option is fitted.

Type of Command: Executing command. Forces execution of all previously deferred commands.

Unprotect

Unprotect **command**

Enables changes to Protected User Data (✱PUD), CalAdjust, SetCode, OnTime, CalData, and CalDefault if sent earlier in the same program message.

Command Syntax: **Unprotect** <string>|<blockdata>; <command to be unprotected>

Where <string>|<blockdata> is the password necessary to gain access to the protected information. The default password is 1875. The code can be changed with SetCode. (String data must be surrounded by " " or ' ' as in the example below.)

- *Example:* Unprotect "1875" ; ✱PUD # 241 Calibrated 1990-01-22, inventory No.02027> ↵

See Also: ✱PUD, OnTime, CalData, Caldefault, CalAdjust, and SetCode.

Protected User Data

command/query

Protected user data. This is a data area in which the user may write only after unprotecting it, but the contents may always be read. A typical use would be to hold calibration information, usage time, environmental conditions, inventory control numbers etc.

Command Syntax: **✱PUD** <blockdata>

• *Example:*

Unprotect "1875" ; ✱PUD #241 Calibrated 1990-04-17, inventory No.02027 ↵

Where:

<string> is the password necessary to get access to the calibration functions of the pulse generator. The default password is 1875. (String data must be surrounded by " " or ' ' as in the example above.)

means that <arbitrary block program data> will follow.

2 means that the two following digits will specify the length of the data block

41 is the number of characters in this example message.

Query Syntax: ✱PUD?

• *Response Format:* <BLOCKDATA> ↵

Type of Command: Immediate command. Executed immediately without executing previous commands.

See Also: Unprotect on page 13-82.

This page is intentionally left blank.

Test & Calibration

Define Test

Define Test **command/query**

Defines which test should be performed when ✱TST? is received.

Command Syntax: **DefTest** (All IPROM I RAMI FRONT I COUNTER)

- *Example:* DefTest PROM ↵

Query Syntax: DefTest?

- *Response Format:* [DEFTEST](ALLIPROMIRAMIFRONTICOUNTER) ↵

Self–Test — query

The self-test query causes an internal self-test and generates a response indicating whether or not the device completed the self-test without any detected errors.

Query Syntax: ***TST?**

- *Response Format:* <integer> ↵

Where:
<integer> = the sum of any detected errors, according to the table below.

<Integer> =	Error
0	No error
1	PROM Failure
2	RAM 1 Failure
4	RAM 2 Failure
8	LCD Failure
16	Keyboard Failure
32	Counter Failure

Type of Command: Executing command. Forces execution of all previously deferred commands.

Calibration Verification

Calibration Verification **query**

Calibration Query. Causes the device to perform an internal calibration verification and generate a response that indicates whether or not the device completed the internal calibration verification without error.

Query Syntax: ***CAL?**

- *Response Format:* <integer>

Where:

<integer> = a number showing the sum of the error type and the parameter where the error occurred. See the table below.

If no error occurred during calibration verification, an instrument without PM 9579, internal calibration unit will respond with '8000H' and an instrument with PM 9579 will respond with '0'.

All other responses indicate an error.

See error explanation table on page 13-90.

Type of Command: Executing command. Forces execution of all previously deferred commands.

Calibration Report **query**

Performs a complete measurement of all parameters and returns a measurement protocol. This function requires the internal calibration unit PM 9579.

Query Syntax: **CalReport?**

- *Response Format:* [CALREPORT] <string>

Where <string> is the protocol with one headline for each parameter and one row for each adjustable DAC-setting.

- *Example:*

Parameter Range

PER, 1,
8.0E–09,7.934E-09,–6.546E-11,–8.183E-03,
9.8E–09,9.785E-09,–1.514E–11,–1.545E-03,
1.26E–08,1.259E–08,–6009E-12,–4.769E–04,
2.0E–8,2.008E–08,8.142E–11,4.071E–03,

Relative error = $\frac{\textit{Absolute error}}{\textit{expected value}}$

Absolute error = *Measured value–expectedvalue*

Measured value

Expected parameter value

Calibration Adjust

Calibration Adjust **query**

Performs a complete measurement and adjustment of all parameters and returns a calibration status code. You must use unprotect before using this command. This function requires the internal calibration unit PM 9579.

Query Syntax: **CalAdjust?**

- *Example:* Unprotect "1875" ; CalAdjust ↵
- *Response Format:* [CALADJUST] <dec. data>

Where: <dec. data> is a number showing the sum of all present calibration errors according to the table below:

	Bit No.	Weight	Explanation
Error Type	15	32768	No calibration system installed
	14	16384	Default calibration
	13	8192	Not used
	12	4096	Not used
	11	2048	Measuring error
	10	1024	Inaccuracy (Call Servicel)
	9	512	Non monotonic (Call Service)
	8	256	Out of DAC-range error. Hardware adjustment required, call Service.
Parameter	7	128	Attenuator calibration
	6	64	Offset calibration
	5	32	Amplitude calibration
	4	16	Trailing edge calibration
	3	8	Leading edge calibration
	2	4	Duration calibration
	1	2	Delay calibration
	0	1	Period calibration

Calibration Status **query**

Returns the calibration status (0 if the calibration status is OK and an other number, according to the table below, if not.)

Query Syntax: **CalStatus?**

- *Response Format:* [CALSTATUS]<dec. data>

Where: <dec. data> is a number showing the sum of all present calibration errors according to the table below:

	Bit No.	Weight	Explanation
Error Type	15	32768	Always 0
	14	16384	Default calibration
	13	8192	Not used
	12	4096	Not used
	11	2048	Measuring error
	10	1024	Inaccuracy (Call Servicel)
	9	512	Non monotonic (Call Service)
	8	256	Out of DAC-range error Hardware adjustment required, call Service.
Parameter	7	128	Attenuator calibration
	6	64	Offset calibration
	5	32	Amplitude calibration
	4	16	Trailing edge calibration
	3	8	Leading edge calibration
	2	4	Duration calibration
	1	2	Delay calibration
	0	1	Period calibration

Calibration Error

Calibration Error **query**

Returns the error descriptions resulting from errors in the last calibration adjustment performed. One error is returned each time the query is issued until no errors remains to be reported. This function requires the internal calibration unit PM 9579.

Query Syntax: **CalError?**

- *Response Format:* {CALERROR] <string>

  Where: <string>=the error description.

- *Example:* "CalErr 1: DELAY Range:2, Low range end point out of range"

  "CalErr 2: AMPLITUDE Mode:NRM, Measuring error in Lo point"

  "CalErr 0: , No more errors"

Type of Command: Executing command. Forces execution of all previously deferred commands.

Calibration Data **command/query**

Calibration data transfer command. This command is used to make a backup copy of the calibration data of the instrument in your controller.* . You must unprotect the calibration data area to be able to write into it.

Command Syntax: **CalData** <Block data>

- *Example:* Unprotect "1875" ; CalData <Block data> ↵

Query Syntax: CalData?

- *Response Format:* [CALDATA] <Block data>

Type of Command: Executing command. Forces execution of all previously deferred commands.

See Also: Block data in the 'System Functions' chapter, and unprotect in this chapter.

* Block data can be difficult to store in the controller as it may well contain data that the controller interprets as 'end of file' etc. Consult your controller manual for correct procedure for storing block data.

Calibration Temperature

Calibration Temperature **query**

Returns the temperature of the pulse generator (inside the cabinet) the last time it was calibrated.

Query Syntax: **CalTemp?**

- *Response Format:* [CALTEMP] <dec. data>

Where <dec. data> is an integer indicating the temperature in degrees Centigrade.

Type of Command: Executing command. Forces execution of all previously deferred commands.

Temperature **query**

Measures the temperature inside the cabinet of the pulse generator and gives a response in degrees Centigrade.

Query Syntax: **Temperature?**

- *Example:* Temperature? ↵
- *Response Format:* [TEMPERATURE] <Decimal data> ↵

Where:

<Decimal data> = the temperature in degrees Centigrade.

Type of Command: Executing command. Forces execution of all previously deferred commands.

Calibration Time

Calibration Time **query**

This query gives the time when calibration was last performed.

Query Syntax: CalTime?

• *Response Format:* [CALTIME] <String> ↵

Type of Command: Executing command. Forces execution of all previously deferred commands.

On Time — command/query

Total elapsed power–on time. This data should only be read. You can write only after unprotecting it.

Command Syntax: **OnTime** <string data>

Where <string data> is the elapsed time in Hours [: Minutes [: Seconds]], minutes and seconds are optional. (String data must be surrounded by " " or ' ' as in the example below.)

- *Example:* Unprotect "1875" ; OnTime "15:10:30" ↵

Query Syntax: OnTime?

- *Response Format:* [ONTIME] <String> ↵

See Also: Unprotect on page 13-82.

Calibration Default **command**

This command returns the instrument to the default calibration stored in Read–Only Memory inside the instrument. This command is only useful if someone seriously corrupted or destroyed the calibration data and the instrument temporarily must be made operative until proper adjustment can be performed. Calibration default is set automatically if the checksum for the calibration data is incorrect.

> **CAUTION: If this command is sent the calibration is lost and the instrument must be recalibrated.**

Command Syntax: **CalDefault**

- *Example:* Unprotect "1875" ; Caldefault ↵

Type of Command: Executing command. Forces execution of all previously deferred commands.

See also: Unprotect

Status & Event

See also System Functions, Status on page 11-19.

Status Byte Query

Status Byte Query **query**

Reads out the value of the service request enable register. The Master Summary Status bit (MSS) is reported on bit 6, not the Request Service (RQS). The MSS is set if the instrument has one or more reasons for requesting service.

Query Syntax: ***STB?**

- *Response Format:* <integer>↵

Where <integer> = the sum (between 0 and 255) of all bits that are set. See table below:

Status Byte Register (1 = true)			
Bit	**Weight**	**Name**	**Condition**
7	128	OPR	Enabled operation status has occurred.
6	64	MSS	Reason for requesting service
5	32	ESB	Enabled event status condition has occurred
4	16	MAV	An output message is ready
3	8	QUE	The quality of the output signal is questionable
2–0	4–1	Not used	

Type of Command: Executing command. Forces execution of all previously deferred commands.

See also: If you want to read the status byte with the RQS bit, use serial poll.

Service Request Enable — command/query

The Service Request Enable command sets the Service Request Enable Register bits. This enable register contains a mask value for the bits to be enabled in the Status Byte Register. A one in the enable register enables the corresponding bit in the Status Byte Register to generate a Service Request.

Command Syntax: **✱SRE** <dec. data>

Where <dec.data> = the sum (between 0 and 255) of all bits that are set. See table below:

Service Request Enable Register (1 = enable)		
Bit	**Weight**	**Enables**
7	128	OPR, Operation Status
6	64	RQS, Request Service
5	32	ESB, Event Status Bit
4	16	MAV, Message Available
3	8	QUE, Questionable Data/Signal Status
2–0	4–1	Not used

- *Example:* ✱SRE <16>↵

In this example a service request is generated when a message is available in the output queue.

Query Syntax: ✱SRE?

- *Response Format:* <integer>↵

Where <integer> = the sum of all bits that are set.

Type of Command: Immediate command. Executed immediately without executing previous commands.

Clear Status Command

Clear Status Command **command**

The ✱CLS common command clears the status data structures by: clearing all event registers and the error queue. It does not clear enable registers and transition filters. It clears any pending ✱WAI, ✱OPC and ✱OPC?.

Command Syntax: **✱CLS**

• *Example:* ✱CLS ↵

Type of Command: Immediate command. Executed immediately without executing previous commands.

Power-On Status Clear — command/query

Enables/disables automatic power-on clearing of:
- Service request enable register
- Event status enable register
- Operation status enable register
- Questionable data/signal enable register

Command Syntax: **✱PSC** <decimal data>

Where:

<dec. data> = a number in the range -32767 to +32767. A value which rounds to 0 turns off automatic power-on clearing. Any other value turns it on.

- *Example:* ✱PSC 1↵

This example enables automatic power on clearing.

Query Syntax: ✱PSC?

- *Response Format:* 1 | 0 ↵

Where 1 is enabled and 0 is disabled.

Type of Command: Immediate command. Executed immediately without executing previous commands.

Event Status Register

Event Status Register **query**

Reads out the contents of the standard event status register. Reading the Standard Event Status Register clears the register.

Query Syntax: **✱ESR?**

- *Response Format:* <dec. data> ↵

Where:

<dec. data> = an integer number between 0 and 255. See table on page 13-105.

Type of Command: Executing command. Forces execution of all previously deferred commands.

Standard Event Status Enable **command/query**

Sets the enable bits of the Standard Event Enable Register. This enable register contains a mask value for the bits to be enabled in the Standard Event Status Register. A one in the enable register enables the corresponding bit in the status register.
An enabled bit will set the ESB (Event Status Bit) in the Status Byte Register if the enabled event occurs. See also status reporting on page 11-19.

Command Syntax: ***ESE** <dec. data>

Where:
<dec.data> = an integer number between 0 and 255. according to the table below:

Event Status Enable Register (1 = enable)		
Bit	**Weight**	**Enables**
7	128	PON, Power ON occurred
6	64	URQ, User Request
5	32	CME, Command Error
4	16	EXE, Execution Error
3	8	DDE, Device–Dependent Error
2	4	QYE, Query Error
1	2	RQC, Request Control (not used)
0	1	Operation Complete

- *Example:* *ESE <36> ↵
  In this example the command error, bit 5, and the query error, bit 2, will set the ESB of the Status Byte if these errors occur.

Query Syntax: *ESE?

- *Response Format:* <dec. data> ↵

Standard Event Status Enable

Type of Command: Immediate command. Executed immediately without executing previous commands.

Operation Status, Condition **query**

Reads out the contents of the Operation Condition Register. See figure below.

Query Syntax: **SOC?**

• *Response Format:* [SOC]<decimal data> ↵

Where:

<decimal data> = an integer number between –32767 and +32767. See the table on page 13-108.

Type of Command: Executing command. Forces execution of all previously deferred commands.

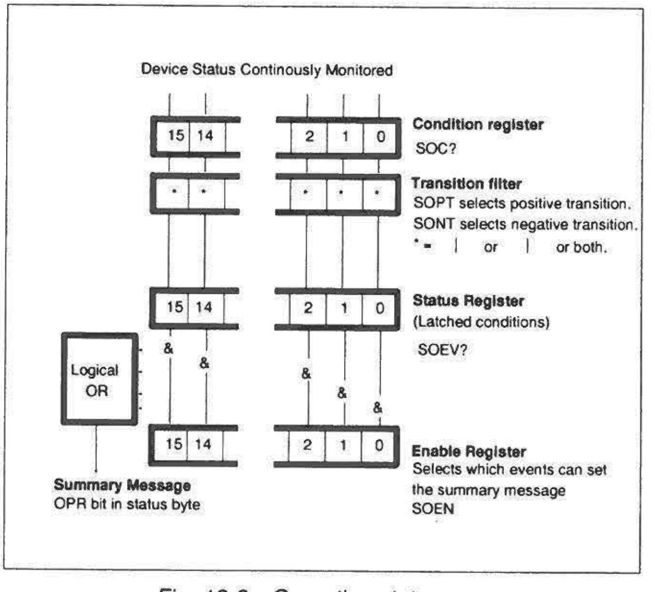


Fig. 13-3 Operation status group

Operation Status, Positive Transition

Operation Status, Positive Transition **command/query**

Transition Filter. This command lets you decide if a positive transition of a condition should set a bit in the Operation Status Event Bit Register. See figure on page 13-107.

The transition filter settings are stored in nonvolatile memory.

Command Syntax: **SOPT** <dec. data>

Where <dec. data> is according to the table below:

Query Syntax: SOPT?

- *Response Format:* [SOPT] <dec. data>

Operation Status registers (1 = enable)		
Bit	**Weight**	**Explanation**
9	512	Initial warm up, 30 seconds from power on
8	256	Warming up, The pulse generator warms up 30 minutes after power on
7-2		Not used
1	2	Settling, The instrument waits for the pulse generating hardware to stabilize
0	1	Calibrating, The instrument is currently adjusting or verifying itself

Operation Status, Negative Transition **command/query**

Transition Filter. This command lets you decide if a negative transition of a condition should set a bit in the Operation Event Bit Register. See figure on page 13-107.

The transition filter settings are stored in nonvolatile memory.

Command Syntax: **SONT** <dec. data>

Where <dec. data> is according to the table on page 13-108.

Query Syntax: SONT?

- *Response Format:* [SONT]<dec. data>

Operation Status, Event

Operation Status, Event **query**

Reads out the contents of the Operation Event Status Register. Reading the Operation Event Register clears the register. See figure on page 13-107.

If a bit should be set in this register, either a positive or a negative transition must be enabled.

Query Syntax: **SOEv?**

- *Response Format:* [SOEV]<dec. data> ↵

Where:

<dec. data> = an integer number between 0 and 65535. See the table on page 13-108.

Type of Command: Executing command. Forces execution of all previously deferred commands.

Operation Status, Enable **command/query**

Sets the enable bits of the Operation Status Enable Register. This enable register contains a mask value for the bits to be enabled in the operation status register. A one in the enable register enables the corresponding bit in the status register. See figure on page 13-107.

An enabled bit will set the OPR (Operation Status Bit) in the Status Byte Register if the enabled event occurs. See also status reporting on page 11-19.

This register will be cleared at power on if power on clearing is enabled via ✱PSC.

Command Syntax: **SOEn** <dec. data>

Where:

<dec.data> = an integer number between 0 and 65535, according to the table on page 13-108.

- *Example:* SOEn <258> ↵

In this example warm up, bit 8, and Settling, bit 1, will set the OPR-bit of the Status Byte. (This method is faster than using ✱OPC if you want to know when the pulse on the output is ready.)

Query Syntax: SOEn?

- *Response Format:* [SOEN]<dec. data> ↵

Type of Command: Immediate command. Executed immediately without executing previous commands.

Questionable Data/signal, Condition **query**

Reads out the contents of the Status Questionable Condition Register.

Query Syntax: **SQC?**

• *Response Format:* [SQC]<dec. data> ↵

Where:

<dec. data> = an integer number between 0 and 65535. See the table on page 13-113.

Type of Command: Executing command. Forces execution of all previously deferred commands.

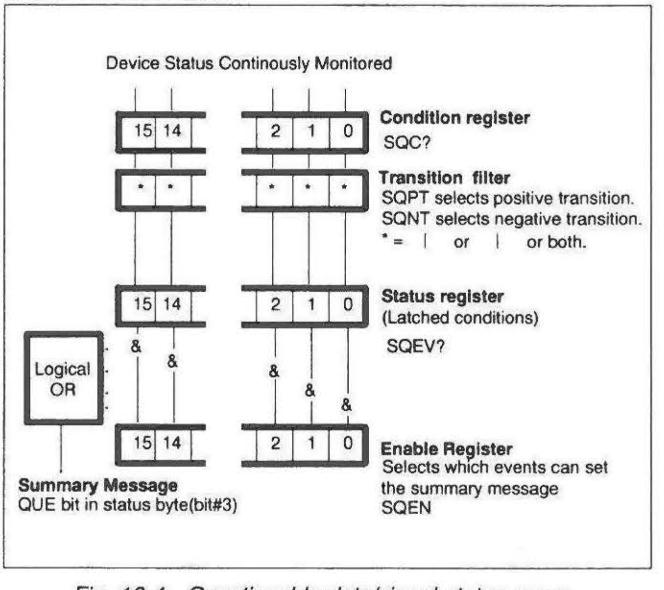


Fig. 13-4 Questionable data/signal status group

Questionable Data/signal Positive Transition **command/query**

Transition Filter. This command lets you decide if a positive transition of a condition should set a bit in the Status Questionable Register. See figure on page 13-112.

The transition filter settings are stored in nonvolatile memory.

Command Syntax: **SQPT** <dec. data>

Where <dec. data> is according to the table below and on the next page.

Query Syntax: "SQPT?

• *Response Format:* [SQPT]<dec. data>

Status Questionable Enable (1 = enable)			
Bit	**Weight**	**Enables**	**Cause for alarm**
9	512	Calibration temperatur e	Temperature outside calibration range
8	256	Calibration	Calibration data is questionable due to default or bad calibration data.
7-6		Not used	
5	32	Frequency	The frequency (pulse period) is of questionable quality due to period control loop not completed (Period control on). Warm up not completed, temperature outside calibration range or bad calibration data
4-3		Not used	

Questionable Data/signal Positive Transition

Status Questionable Enable (1 = enable)			
Bit	**Weight**	**Enables**	**Cause for alarm**
2	4	Time	Time parameters of questionable quality due to warm up not completed, temperature outside calibration range or bad calibration data
1	2	Not used	
0	1	Voltage	Output voltage questionable due to warm up not completed, temperature outside calibration range or bad calibration data

Questionable Data/signal, Negative Transition — command/query

Transition Filter. This command lets you decide if a negative transition of a condition should set a bit in the Status Questionable Event bit register. See figure on page 13-112.

The transition filter settings are stored in nonvolatile memory.

Command Syntax: **SQNT** <dec. data>

Where <dec. data> is according to the table on page 13-113.

Query Syntax: "SQNT?

- *Response Format:* [SQNT]<dec. data>

Questionable Data/signal, Event

Questionable Data/signal, Event — query

Reads out the contents of the Status Questionable Event Register. Reading the Status Questionable Event Register clears the register. See figure on page 13-112.

Query Syntax: **SQEv?**

- *Response Format:* SQEV <dec. data> ↲

Where:

<dec. data> = an integer number between 0 and 65535. See table on page 13-113.

Type of Command: Executing command. Forces execution of all previously deferred commands.

Questionable Data/signal, Enable — command/query

Sets the enable bits of the Status Questionable Enable Register. This enable register contains a mask value for the bits to be enabled in the Status Questionable Register. A one in the enable register enables the corresponding bit in the status register. See figure on page 13-112.

An enabled bit will set bit #3, QUE (Questionable Status Bit), in the Status Byte Register if the enabled event occurs. See also status reporting on page 11-19.

This register will be cleared at power on if power on clearing is enabled via ✱PSC.

Command Syntax: **SQEn** ␣ <dec. data>

Where:
<dec.data> = an integer number between 0 and 65535, according to the table on page 13-113.

• *Example:* SQEn ␣ <32> ↵

In this example frequency, bit 5, will set the QUE-bit of the Status Byte when the frequency of the output pulse is OK. This method of knowing when the pulse on the output is OK is better than ✱OPC or ✱WAI since the controller can service other instruments while the pulse generator is settling.

Query Syntax: SQEn?

• *Response Format:* [SQEN] ␣ <dec. data> ↵

Type of Command: Immediate command. Executed immediately without executing previous commands.

This page is intentionally left blank.

Synchronization

Operation Complete

Operation Complete command

The Operation Complete command causes the device to generate the operation complete message in the Standard Event Status Register when all pending selected device operations have been finished.

Command Syntax: ***OPC**

- *Example:* *OPC ↵

Type of Command: Executing command. Forces execution of all previously deferred commands.

Operation Complete Query — query

The Operation Complete query places an ASCII character 1 into the device's Output Queue when all pending selected device operations have been finished.

Query Syntax: ***OPC?**

- *Response Format:* '1' ↵

Type of Command: Executing command. Forces execution of all previously deferred commands.

Wait-to-continue

Wait-to-continue **command**

The Wait-to-Continue command prevents the device from executing any further commands or queries until execution of all previous commands or queries have been completed.

Command Syntax: ***WAI**

- *Example:* PERIOD 1.00E-3 ; *WAI ; OUTPUT ON , A ↵

In this example, *WAI makes sure the PERIOD setting is ready before switching on output A.

Type of Command: Executing command. Forces execution of all previously deferred commands.

Command Set, Alphabetical

Introduction

This chapter contains an alphabetical subset of the "Command Reference" chapter.

Command/Query	Description
***CAL**?	Causes the device to perform an internal calibration. 0 = OK.
***CLS**	Clear Status clears the status data structures.
***DDT** <blockdata>	Define Device Trigger stores command sequence executed when group execute trigger (GET) or *TRG is received.
***DMC** <string> , <blockdata>	Define Macro command assigns a sequence of program message units to a macro label.
***EMC** <dec. data>	Enable Macros enables and disables expansion and execution of macros.
***ESE** <dec. data>	Standard Event Status Enable. An enabled bit will set the ESB (Event Status Bit) in the Status Byte Register if the enabled event occurs.
***ESR?**	Event Status Register reads out the contents of the standard event status register.
***GMC?** <string>	Get Macro query response with current definition for the given macro label.
***IDN?**	Identify, answered with three fields containing manufacturer, model, and firmware revisions.

Command/Query	Description
✱LMC?	Learn Macro Query, makes the instrument output all macro labels defined in the instrument.
✱LRN?	Learn Device Setup Query. Causes a response message which is directly resendable and will place the instrument in the state it was when the ✱LRN? query was made.
✱OPC	Operation Complete causes the device to generate the operation complete message in the Standard Event Status Register when all pending selected device operations have been finished.
✱OPT?	Option is answered with one data field for each reportable option. Absent options are indicated with a zero.
✱PMC	Purge Macros removes all macro definitions.
✱PSC <decimal data>	Power-On Status Clear, Enables/disables automatic power-on clearing of Service Request Enable Register and Standard Event Status Enable Register
✱PUD <blockdata>	Protected user data. This is a data area in which the user may write only after unprotecting it, while the contents may always be read.
✱RCL <dec. data>	Recall restores the state of the device from a copy stored in local memory.
✱RST	Reset command performs a device reset. Return to XRCL 0 settings.
✱SAV <decimal data>	Saves the current settings of the instrument in internal memory
✱SRE <dec. data>	Service Request Enable. A one (1) in the enable register enables the corresponding bit in the Status Byte Register to generate a Service Request.

Command/Query	Description
***STB?**	Status Byte Query reads out the value of the status byte.
***TRG**	Trigger command is the device-specific equivalent of the IEEE 488.1 defined Group Execute Trigger, and has exactly the same effect as a GET when received, parsed, and executed by the device.
***TST?**	Self Test. Causes an internal self-test and generates a response indicating whether or not the device completed the self-test without any detected errors.
***WAI**	Wait-to-Continue command prevents the device from executing any further commands or queries until execution of all previous commands or queries have been completed.
Autoset <parameter>	Autoset corrects the pulse generator setting, using the selected parameter as reference.
Beep / Bell	Beep makes the pulse generator beep, if beeper is enabled.
Beeper (On | Off)	Enables/Disables the beeper.
BinProg <block data>	Binary program query reads the current setting of the pulse generator and sends it to the controller in binary form.
Burst / Bur <decimal data>	Burst pulses sets the number of pulses in each burst to between 0 and 9999.
BurstPeriod / BrP <decimal data>	Sets the time between the starting point of two consecutive bursts.
CalAdjust?	Requires optional Internal calibration unit. Performs a complete measurement and adjustment of all parameters and returns a status code.

<table>
<tr><th>Command/Query</th><th>Description</th></tr>
<tr><td>CalData <Block data></td><td>Calibration data is used to restore the calibration data of the instrument from a previous backup. (Protected).</td></tr>
<tr><td>CalDefault</td><td>Calibration default returns the instrument to the default calibration in ROM. (Protected).</td></tr>
<tr><td>CalError?</td><td>Requires optional Internal calibration unit. Returns descriptions of errorors in the last CalAdjust?</td></tr>
<tr><td>CalReport</td><td>Requires optional Internal calibration unit. Performs a complete measurement of all parameters and returns a protocol.</td></tr>
<tr><td>CalStatus?</td><td>Calibration status returns the calibration status, i.e., 0 if the instrument is calibrated, and a number if not.</td></tr>
<tr><td>CalTemp?</td><td>Calibration temperature returns the temperature the pulse generator had inside the cabinet the last time it was calibrated.</td></tr>
<tr><td>CalTime</td><td>Used to store the time when the pulse generator was calibrated last.</td></tr>
<tr><td>ClockLevel / ClkLvl (TTL | ECL)</td><td>Clock output level TTL levels (0 and +2.5 V into 50 Ω) or ECL levels (-0.8 V and -1.8 V into 50 Ω)</td></tr>
<tr><td>Codes?</td><td>Codes returns all user commands understood by the instrument.</td></tr>
<tr><td>Complement / Compl (Off | On)</td><td>Switches between normal and complementary waveform on both A and B outputs.</td></tr>
<tr><td>Cursor / Cur <parameter></td><td>Moves the cursor to the selected parameter.</td></tr>
<tr><td>Debug (Off | On)</td><td>The characters that are received on the bus are echoed to the display. Used to check that communication is OK.</td></tr>
</table>

<table>
<tr><th>Command/Query</th><th>Description</th></tr>
<tr><td>DefSequence / DefSeq (A | B | C | D) , Empty|<dec. data></td><td>Defines the settings to be recalled in sequence (using 'Next' and 'Previous').</td></tr>
<tr><td>DefTest (All | PROM | RAM |Front | Counter)</td><td>Defines which test to be performed when *TST? is received.</td></tr>
<tr><td>Delay / Dly <decimal data></td><td>Sets delay between pulse on the clock output and the pulse outputs. If double pulse, the first pulse is synchronous with the clock pulse and the second pulse is generated after the delay set with this command.</td></tr>
<tr><td>DeleteMemory / DelMem (<decimal data> |All) { <decimal data> }</td><td>Deletes the selected settings from the nonvolatile memory in the pulse generator.</td></tr>
<tr><td>DispFormat / DispFmt (Exp / Exponent | Unit)</td><td>Selects exponent or prefix+unit representation of parameter values in the display.</td></tr>
<tr><td>DispMode (Off | State | All)</td><td>ALL show all values in the different displays.
STATE, central display disabled, all other indicators work.
OFF, only bus indicator works</td></tr>
<tr><td>Duration / Dur / Width <decimal data></td><td>Sets the duration of the output pulse within the range 4 ns to 100 ms.</td></tr>
<tr><td>Error?</td><td>Reads out an error message from a FIFO queue (First In First Out)</td></tr>
<tr><td>Errorheck / ErrChk (Off | On)</td><td>Used for maximum speed of already debugged programs.</td></tr>
</table>

Command/Query	Description
ErrorDisp ON/OFF	Selects if GPIB error messages should be shown on the display
Help?	Help query gives a brief help text, mentioning other available queries in the help category.
HighLevel / HiLvl / HiL <decimal data>	Sets the high level of the output pulse to any value between –9.8 V and +10 V.
Impedance / Imp (High / HilLow / Lo)	Selects the input impedance of the external input.
Increment / Inc	Increments the parameter that is selected with the SETINCREMENT command.
LeadingEdge / LeE <decimal data>	Specifies the transition time, from 10% to 90% of the pulse amplitude, within the range 2 ns to 100 ms.
Limiter / Lim (Off I On)	Enables/disables the limiter function.
Locking / Lock (Off I [On) {<parameter> }]	Prevents unintentional change of pulse parameters.
LowLevel / LoLvl / LoL <decimal data>	Sets the low level of the output pulse to any value between –10 V and +9.8 V.
ManualGate / ManGate (CloseIOpen)	Opens and closes the gate when Trigger Source is set to Manual and Trigger Function to Gate or Duration.
ManualTrig/ ManTrg	Triggers the pulse generator when Trigger-Source is set to Manual, and Trigger-Function to Trig or Burst. When the Trigger-Function is set to Gate or Duration, ManTrig toggles the gate open the first time, and closed the next time you send it.

Command/Query	Description
MemProg <dec. data> , <block data>	Memory program, MemProg? query reads one of the settings (saved with *SAV) and sends it to the controller in binary form.
Next	Recalls the next setting in the used sequence.
Offset_B <decimal data>	Adds an offset voltage between –19.9 V and +19.9 V to the high and low levels of output B.
OnTime <string data>.	Total elapsed power on time.
Output / Outp (Disable / Off I [Enable / On) { (A I B I Clock / Clk) }]	Enables and disabled output A, B, and Clock.
ParLimits? / ParLim ? <parameter>	Returns the present max. and min. limits for the selected parameter.
Period/ Per <decimal data>	Sets the pulse period to any value between 8 ns and 10 s.
PeriodCtrl / PerCtrl (On I Off)	Disables/enables the period control loop.
Previous / Prev	Recalls the previous setting in the used sequence.
PulseMode / PlsMod (Square / Sqr I Single / Sgl I Double / Dbl)	Selects Pulse Mode.
ResHdr (None / No I Short I Long)	Response header selects long, short, or no header of bus responses.

Command/Query	Description
SetIncrement / SetInc [<parameter> , (Absolute / Abs I Relative / Rel)	Sets parameter to be changed, whether the change shall be absolute or relative, and the absolute/relative change that shall be made when INCREMENT is received.
SOC?	Reads out the contents of the operation condition register.
SOEn <dec. data>	Operation Status Enable. A one (1) in the enable register enables the corresponding bit in the status register. An enabled bit will set the OPR (Operation Status Bit) in the Status Byte Register when the enabled status occurs.
SOEv?	Reads out the contents of the operation event status register and clears it.
SONT <dec. data>	Operation Negative Transition. Decides if a negative transition of a condition should set a bit in the Operation Event bit register.
SOPT <dec. data>	Operation Positive Transition. Decides if a positive transition of a condition should set a bit in the Operation Event bit register.
SQC?	Reads out the contents of the Status Questionable Condition register.
SQEn <dec. data>	Sets the enable bits of the status questionable enable register.An enabled bit will set the QUE (Questionable Status Bit) in the Status Byte Register as long as the instrument is in the questionable status.
SQEv?	Reads out the contents of the status questionable event register. Reading the Status Questionable Event Register clears the register.

Command/Query	Description
SQNT <dec. data>	Transition Filter. This command lets you decide if a negative transition of a condition should set a bit in the Status Questionable register. The bit must also be enabled in the 'operation status enable' register.
SQPT <dec. data>	Transition Filter. This command lets you decide if a positive transition of a condition should set a bit in the Status Questionable register. The bit must also be enabled in the 'operation status enable' register.
Syntax? <string>	Gives a response containing the syntax of the command specified in the query. The response is in a simplified EBNF format.
Temperature?	Measures the temperature inside the cabinet of the pulse generator and gives a response in degrees Centigrade.
Text (Off I <string>)	Writes a text on the pulse generator display (Max 80 characters). The text is removed either when a button is pushed or when a new bus command is received or when TEXT OFF is received.
Tracking / Track (Off I [On) { , <parameter> }]	Selects which parameters shall be tracked to each other. Time parameters are tracked to time parameters; they are kept in the ratio that they have when the command is received. High and low level are tracked with a constant difference.
TrailingEdge / TrE <decimal data>	Specifies the transition time from 90% to 10% of the pulse amplitude.
TrigFunction / TrgFnc (Trig / Trg I Gate I Burst / Bur I Duration / Dur)	Selects: • Triggered by the trigger source. • Gated by the trigger source. • Trig the burst generator with the trigger source. • Duration controlled by the trigger source

<table>
<tr><th>Command/Query</th><th>Description</th></tr>
<tr><td>TrigLevel / TrgLvl (<decimal data> | TTL | ECL | CMOS)</td><td>Sets the trigger level of the external input to any value between +5 V and -5 V.</td></tr>
<tr><td>TrigSlope / TrgSlp (Pos | Neg)</td><td>Selects between triggering on the positive or negative going slope of the input signal.</td></tr>
<tr><td>TrigSource / TrgSrc (Internal / Int | External / Ext | Manual / Man)</td><td>Selects internal, external, or manual trigger source.</td></tr>
<tr><td>Unprotect <string></td><td>Enables changes to SetCode, Protected User Data (*PUD), OnTime, CalAdjust, CalData, and CalDefault if sent earlier in the same program message.</td></tr>
<tr><td>UseSequence / UseSeq Memory / Mem | A | B | C | D</td><td>Selects the sequence that will be recalled when Next and Previous are received. UseSequence Memory means that all defined settings, except the default setting, are recalled in sequence.</td></tr>
<tr><td>VoltLimits / VLim <dec. data> , <dec. data></td><td>Sets min. and max. limits for the output voltages of channel A and B to any value between +10 V and -10 V.</td></tr>
</table>

This page is intentionally left blank.

Program Examples

Introduction

The program examples in this manual are written for an IBM-PC compatible computer equipped with a Philips PM 2201 GPIB interface board. This board is compatible with the IBM General Purpouse Interface Bus Adapter and National Instruments PC-IIA.

The exampes are written in GW-basic. Use IBCONF.EXE to setup a device with the name "DEV1" according to the following:

```
National Instruments          | Device Characteristics       | IBM AT, PS/2-25/30
Device: DEV1                  | Access: GPIB0 | SELECT (use right/left arrow keys):
Primary GPIB Address .............17          | 0 to 30
Secondary GPIB Address...........NONE         |
Timeout setting..................T10s         |
EOS byte..........................00H         |
Terminate Read on EOS ............no          |
Set EOI with EOS .................no          |
Type of compare on EOS...........7-bit        |
Set EOI w/last byte of Write.....yes          |

F1: Help      F2: Explain Field      F5: Reset value      F9: Return to Map
```

NOTE: That lines 20 to 40 in each example initiates the controller board (PM 2201). These lines can be added to your program by merging your program with the file DECL.BAS delivered with the GPIB software to the controller board.

Example 1

A simple program that first sets the PM 5781 to default setting, then makes some modifications from the default setting and sweeps the pulse period.

```
10 REM Lines 20, 30 and 40 are available in DECL.BAS
20 CLEAR ,60000! : IBINIT1=60000! : IBINIT2=IBINIT1+3 : BLOAD "bib.m",IBINIT1
30 CALL IBINIT1(IBFIND,IBTRG,IBCLR,IBPCT,IBSIC,IBLOC,IBPPC,IBBNA,IBONL,IBRSC,
IBSRE,IBRSV,IBPAD,IBSAD,IBIST,IBDMA,IBEOS,IBTMO,IBEOT,IBRDF,IBWRTF,IBTRAP,IBDEV,
IBLN)
40 CALL IBINIT2(IBGTS,IBCAC,IBWAIT,IBPOKE,IBWRT,IBWRTA,IBCMD,IBCMDA,IBRD,IBRDA,
IBSTOP,IBRPP,IBRSP,IBDIAG,IBXTRC,IBRDI,IBWRTI,IBRDIA,IBWRTIA,IBSTA%,IBERR%,IBCNT%)

50 PGNAME$ = "DEV1"
60 CALL IBFIND (PGNAME$, PG%)
70 REM
80 REM Use NEWLINE as program message terminator
90 NL$ = CHR$(10)
100 REM
110 REM ---- SET PULSE GENERATOR TO DEFAULT SETTING ----
120 WRT$ = "*RST" + NL$
130 CALL IBWRT (PG%, WRT$)
140 REM
150 REM ---- MODIFY PULSE GENERATOR SETTING ----
160 WRT$ = "Delay 500E-9; Duration 250E-9; PulseMode Double" + NL$
170 CALL IBWRT (PG%, WRT$)
180 WRT$ = "HighLevel 2.5; LowLevel -2.5" + NL$
190 CALL IBWRT (PG%, WRT$)
200 WRT$ = "Output On, A, Clk" + NL$
210 CALL IBWRT (PG%, WRT$)
220 REM
230 REM ---- SWEEP PERIOD BETWEEN 1E-6 AND 100E-6 ----
240 FOR I = 1 TO 100
250     WRT$ = "Period " + STR$(I) + "E-6" + NL$
260     CALL IBWRT (PG%, WRT$)
270     PRINT WRT$
280 NEXT I
290 END
```

To make program execution faster, add these lines.

```
211 WRT$ = "DispMode Off; ErrorCheck Off" + NL$
212 CALL IBWRT (PG%, WRT$)
```

This turns off the front panel displays and the error checking. Error checking should only be disabled when you know that the values you send are correct.

To make it even faster add these lines too.

```
215 WRT$ = "*DDT #19Increment; SetIncrement Period, Absolute, 1E-6" + NL$
216 CALL IBWRT (PG%, WRT$)
```

and replace lines 250, 260 and 270 with

```
250 CALL IBTRG(PG%)
```

This defines the device trigger as "Increment", i.e. the PM 5781 will execute the "Increment" command each time a GET (group execute trigger) is received. The size of the increment is set with "SetIncrement".

Example 2

Shows the simplest synchronization method. An *OPC? query is added at the end of the programming message. The timeout of the controllers read statement is used to wait for the PM 5781 to have stabilized.

```
10 REM Lines 20, 30 and 40 are available in DECL.BAS
20 CLEAR ,60000! : IBINIT1=60000! : IBINIT2=IBINIT1+3 : BLOAD "bib.m",IBINIT1
30 CALL IBINIT1(IBFIND,IBTRG,IBCLR,IBPCT,IBSIC,IBLOC,IBPPC,IBBNA,IBONL,IBRSC,
IBSRE,IBRSV,IBPAD,IBSAD,IBIST,IBDMA,IBEOS,IBTMO,IBEOT,IBRDF,IBWRTF,IBTRAP,IBDEV,
IBLN)
40 CALL IBINIT2(IBGTS,IBCAC,IBWAIT,IBPOKE,IBWRT,IBWRTA,IBCMD,IBCMDA,IBRD,IBRDA,
IBSTOP,IBRPP,IBRSP,IBDIAG,IBXTRC,IBRDI,IBWRTI,IBRDIA,IBWRTIA,IBSTA%,IBERR%,
IBCNT%)
50 PGNAME$ = "DEV1"
60 CALL IBFIND (PGNAME$, PG%)
70 REM
80 REM Use NEWLINE as program message terminator
90 NL$ = CHR$(10)
100 REM
110 REM ---- SET PULSE GENERATOR TO DEFAULT SETTING ----
120 WRT$ = "*RST" + NL$
130 REM CALL IBWRT (PG%, WRT$)
140 REM
150 REM ---- MODIFY PULSE GENERATOR SETTING ----
160 REM Set triangle wave and enable tracking of period, duration, leading
170 REM and trailing edge
180 WRT$ = "Per 100E-9; Dly 8E-9; Dur 50E-9; LeE 40E-9; TrE 40E-9; PlsMod
Sgl" + NL$
190 CALL IBWRT (PG%, WRT$)
200 WRT$ = "Tracking On, Period, Duration, LeadingEdge, TrailingEdge" + NL$
210 CALL IBWRT (PG%, WRT$)
220 WRT$ = "Output On, A, Clk" + NL$
230 CALL IBWRT (PG%, WRT$)
240 REM
250 REM ---- SWEEP PERIOD BETWEEN 100E-9 AND 10E-6 ----
260 RD$ = SPACE$(10)
270 PRINT "SWEEPING"
280 FOR I = 1 TO 100
290     WRT$ = "Period " + STR$(I) + "00E-9; *OPC?" + NL$
```

```
300     CALL IBWRT (PG%, WRT$)
310     REM
320     REM ---- WAIT FOR PULSE GENERATOR TO STABILIZE ----
330     CALL IBRD (PG%, RD$)
340     REM
350     REM ---- PERFORM MEASUREMENT........ ----
360 NEXT I
370 END
```

Example 3

Reads the setting of PM 5781 using binprog. Waits for the user to change the setting, then sends the binprog setting back to PM 5781.

```
10 REM Lines 20, 30 and 40 are available in DECL.BAS
20 CLEAR ,60000! : IBINIT1=60000! : IBINIT2=IBINIT1+3 : BLOAD "bib.m",IBINIT1
30 CALL IBINIT1(IBFIND,IBTRG,IBCLR,IBPCT,IBSIC,IBLOC,IBPPC,IBBNA,IBONL,IBRSC,
IBSRE,IBRSV,IBPAD,IBSAD,IBIST,IBDMA,IBEOS,IBTMO,IBEOT,IBRDF,IBWRTF,IBTRAP,IBDEV,
IBLN)
40 CALL IBINIT2(IBGTS,IBCAC,IBWAIT,IBPOKE,IBWRT,IBWRTA,IBCMD,IBCMDA,IBRD,IBRDA,
IBSTOP,IBRPP,IBRSP,IBDIAG,IBXTRC,IBRDI,IBWRTI,IBRDIA,IBWRTIA,IBSTA%,IBERR%,IBCNT%)
50 PGNAME$ = "DEV1"
60 CALL IBFIND (PGNAME$, PG%)
70 REM
80 REM Use NEWLINE as program message terminator
90 NL$ = CHR$(10)
100 REM
110 REM Set result header to short to make the binprog block resendable
120 WRT$ = "reshdr short" + NL$
130 CALL IBWRT (PG%, WRT$)
140 REM
150 REM ---- READ BINPROG BLOCK ----
160 WRT$ = "BINPROG?" + NL$
170 CALL IBWRT (PG%, WRT$)
180 BPR$ = SPACE$(132)
190 CALL IBRD (PG%, BPR$)
200 REM
210 REM ---- CHANGE SETTING ----
220 CALL IBLOC(PG%)
230 INPUT "Change setting then press <enter>", A
240 REM
250 REM ---- SEND BINPROG BLOCK BACK ----
260 CALL IBWRT(PG%, BPR$)
270 END
```

Example 4

Some examples of macros.

```
10 REM Lines 20, 30 and 40 are available in DECL.BAS
20 CLEAR ,60000! : IBINIT1=60000! : IBINIT2=IBINIT1+3 : BLOAD "bib.m",IBINIT1
30 CALL
IBINIT1(IBFIND,IBTRG,IBCLR,IBPCT,IBSIC,IBLOC,IBPPC,IBBNA,IBONL,IBRSC,IBSRE,IBRSV,I
BPAD,IBSAD,IBIST,IBDMA,IBEOS,IBTMO,IBEOT,IBRDF,IBWRTF,IBTRAP,IBDEV,IBLN)
40 CALL
IBINIT2(IBGTS,IBCAC,IBWAIT,IBPOKE,IBWRT,IBWRTA,IBCMD,IBCMDA,IBRD,IBRDA,IBSTOP,IBRP
P,IBRSP,IBDIAG,IBXTRC,IBRDI,IBWRTI,IBRDIA,IBWRTIA,IBSTA%,IBERR%,IBCNT%)
50 PGNAME$ = "DEV1"
60 CALL IBFIND (PGNAME$, PG%)
70 REM
80 REM Use NEWLINE as program message terminator
90 NL$ = CHR$(10)
100 REM
110 REM ---- PURGE MACROS AND DISABLE MACROS ----
120 WRT$ = "*PMC; *EMC 0" + NL$
130 CALL IBWRT (PG%, WRT$)
140 REM
150 REM ---- DEFINE MACROS ----
160 REM TTL and ECL output
170 WRT$ = "*DMC 'ttl', 'hil 3.5;lol 0;lee 2e-9;tre 2e-9'" + NL$
180 CALL IBWRT (PG%, WRT$)
190 WRT$ = "*DMC 'ecl', 'hil -0.9;lol -1.75;lee 2e-9;tre 2e-9'" + NL$
200 CALL IBWRT (PG%, WRT$)
210 REM
220 REM ECL leading and trailing edge
230 WRT$ = "*DMC 'ecllee', 'lee ($1/.75)'" + NL$
240 CALL IBWRT (PG%, WRT$)
250 WRT$ = "*DMC 'ecltre', 'tre ($1/.75)'" + NL$
260 CALL IBWRT (PG%, WRT$)
270 REM
280 REM Frequency
290 WRT$ = "*DMC 'freq', 'per (1/$1)'" + NL$
300 CALL IBWRT (PG%, WRT$)
310 REM
320 REM "True" duration (measured at 50% of amplitude)
330 WRT$ = "*DMC 'ttre', 'tre $1;dur (dur+(5/8*(tre-$1)))'" + NL$
340 CALL IBWRT (PG%, WRT$)
```

```
350 WRT$ = "*DMC 'tlee', 'lee $1;dur (dur+(5/8*($1-lee)))'" + NL$
360 CALL IBWRT (PG%, WRT$)
370 WRT$ = "*DMC 'tdur', 'dur ($1+(5/8*(lee-tre)))'" + NL$
380 CALL IBWRT (PG%, WRT$)
390 REM
400 REM Amplitude and offset (Offset at 50% of amplitude)
410 WRT$ = "*DMC 'ampl', 'hil ((hil+lol+$1)/2);lol ((hil+lol-$1)/2)'" + NL$
420 CALL IBWRT (PG%, WRT$)
430 WRT$ = "*DMC 'offs', 'hil ($1+(hil-lol)/2);lol ($1+(lol-hil)/2)'" + NL$
440 CALL IBWRT (PG%, WRT$)
450 REM
460 REM Duty factor
470 WRT$ = "*DMC 'duty', 'dur (per*($1/100))'" + NL$
480 CALL IBWRT (PG%, WRT$)
490 REM
500 REM ---- ENABLE MACROS ----
510 WRT$ = "*EMC 1" + NL$
520 CALL IBWRT (PG%, WRT$)
530 END
```

Example 5

Sets up the status reporting for service request on Message available and Command, Execution and Query error. Reads a command from the keyboard and sends it to PM 5781, then checks the status byte using serial poll, determines the reason and reads query responses and error messages.

```
10 REM Lines 10, 20, 30 are available in DECL.BAS
20 CLEAR ,60000! : IBINIT1=60000! : IBINIT2=IBINIT1+3 : BLOAD "bib.m",IBINIT1
30 CALL IBINIT1(IBFIND,IBTRG,IBCLR,IBPCT,IBSIC,IBLOC,IBPPC,IBBNA,IBONL,IBRSC,
IBSRE,IBRSV,IBPAD,IBSAD,IBIST,IBDMA,IBEOS,IBTMO,IBEOT,IBRDF,IBWRTF,IBTRAP,IBDEV,
IBLN)
40 CALL IBINIT2(IBGTS,IBCAC,IBWAIT,IBPOKE,IBWRT,IBWRTA,IBCMD,IBCMDA,IBRD,IBRDA,
IBSTOP,IBRPP,IBRSP,IBDIAG,IBXTRC,IBRDI,IBWRTI,IBRDIA,IBWRTIA,IBSTA%,IBERR%,IBCNT%)

50 PGNAME$ = "DEV1"
60 CALL IBFIND (PGNAME$, PG%)
70 REM
80 REM Use new line as program message terminator
90 NL$ = CHR$(10)
100 REM
110 REM ---- CLEAR STATUS ----
120 WRT$ = "*cls" + NL$
130 CALL IBWRT (PG%, WRT$)
140 REM
150 REM ---- SET EVENT STATUS ENABLE ----
160 REM Enable Command Error, Execution Error and Query Error
170 WRT$ = "*ese 52" + NL$
180 CALL IBWRT (PG%, WRT$)
190 REM
200 REM ---- SET SERVICE REQUEST ENABLE ----
210 REM Enable Service Request on Event Status and Message Available
220 WRT$ = "*sre 48" + NL$
230 CALL IBWRT (PG%, WRT$)
240 REM
250 REM ======== MAIN LOOP ==========================================
260 WHILE 1
270     REM
280     REM ---- ENTER COMMAND STRING AND SEND TO PULSE GENERATOR ----
```

```
290     LINE INPUT "Enter command string (<CR> to end):", CMD$
300     IF CMD$ = "" GOTO 750
310     CMD$ = CMD$ + NL$
320     CALL IBWRT (PG%, CMD$)
330     REM WAIT for execution
340     FOR I=1 TO 100 : NEXT I
350     REM
360     REM ---- READ STATUS BYTE ----
370     CALL IBRSP (PG%, SPR%)
380     IF SPR% <> 0 THEN PRINT "Status byte = "; SPR%
390     ELSE GOTO 740
400         REM
410         REM ---- CHECK MESSAGE AVAILABLE BIT ----
420         WHILE SPR% AND 16
430             PRINT "  Message available bit set"
440             MSG$ = SPACE$(255)
450             CALL IBRD (PG%, MSG$)
460             LFPOS = INSTR(MSG$, CHR$(10))
470             IF LFPOS <> 0 THEN PRINT "Response = " LEFT$(MSG$, LFPOS)
480             IF LFPOS = 0 THEN PRINT "Response = "; MSG$
490             CALL IBRSP (PG%, SPR%)
500         WEND
510         REM
520         REM ---- CHECK EVENT STATUS BIT ----
530         IF NOT SPR% AND 32 GOTO 740
540             PRINT "  Event status bit set"
550             WRT$ = "*esr?" + NL$
560             CALL IBWRT (PG%, WRT$)
570             ESR$ = SPACE$(255)
580             CALL IBRD (PG%, ESR$)
590             ESR% = VAL(ESR$)
600             IF ESR% AND 32 THEN PRINT "     Command error"
610             IF ESR% AND 16 THEN PRINT "     Execution error"
620             IF ESR% AND 4  THEN PRINT "     Query error"
630             REM
640             REM ---- READ ERROR MESSAGES ----
650             WRT$ = "error?" + NL$
660             ERRMESS$ = SPACE$(255)
670             CALL IBWRT (PG%, WRT$)
680             CALL IBRD (PG%, ERRMESS$)
690             WHILE NOT INSTR(ERRMESS$, "No error") <> 0
700                 PRINT LEFT$(ERRMESS$, INSTR(ERRMESS$, CHR$(10)))
710                 CALL IBWRT (PG%, WRT$)
720                 CALL IBRD (PG%, ERRMESS$)
```

```
730             WEND
740 WEND
750 PRINT "PROGRAM TERMINATED"
760 END
```

Error Messages

Front Panel Operation

 # + Beeper

When a parameter value or function mode is changed, the sign # may appear on the display together with a sound and a message saying that one value ought to be lower or higher than a certain limit. The message indicates the limit of the allowed range for the parameter at the cursor.

The limitation may be due to the specification of that particular parameter or because some other parameter values make the combination impossible. The error message also use the # sign to indicate if there is another parameter limiting the range.

If the cursor is moved to another # marked parameter the limits of that parameter will be indicated.

The faulty value can be shown on the display, but the parameter will not change at the outputs as long as the value is outside the limit. The pulse configuration on the outputs will remain on the last legal combination set. This is to avoid impossible settings or unstable pulses.

When you change the pulse parameter values to a new combination, you may experience one or several error messages during the setting procedure. These messages may be neglected during the setting. What counts is that the final combination is accepted by the generator.

Corrections:	An erroneous setting may be corrected in several ways.
• *AUTO SET*	The AUTO SET function will make a correction of the other parameters out from the selected parameter.
• *Vernier*	With the cursor on the parameter, press the correct "fast" VERNIER button to set the value. The parameter value will jump back to within the range.
• *Reset*	When several parameters are faulty, press RESET and the last correct front panel setting is repeated.

Error messages:

Power On	**Remarks**
LCD test failed	Report to service
Keyboard/led test failed	Report to service
PROM test failed	Report to service
RAM 1 test failed	Report to service
RAM 2 test failed	Report to service
Counter test failed	Report to service
Real time clock test failed	Report to service
Calibration data lost	Battery empty? report to service
GPIB address and sequences lost	Battery empty? report to service
Error setting up GPIB hash table	Battery empty? report to service
Error in macro checksum, macros purged	Battery empty? report to service
Active setting erroneous, default set	Set the error check ON.
Uncalibrated, default constants used	New calibration requested
Bad calibration status	Read: Internal calibration, Reference about calibration errors
Internal calibration voltage reference data lost	Battery empty? report to service
Access code lost, default code set	Battery empty? report to service

Tests	Remarks
PROM test passed	
RAM 1 test passed	
RAM 2 test passed	
Front panel test passed	
Counter test passed	
PROM test failed	Report to service
RAM X test failed, Address = YYYY:YYYY Hex	Report to service
LCD test failed	Report to service
Keyboard/LED test failed	Report to service
Counter test failed	Report to service

Temperature and Warm Up	Remarks
Warm up in progress	Wait 30 seconds
Initial warm up completed	The generator is ready for use but to a poor specification
Warm up completed	Ready for use according the specification
Temperature inside internal calibration range	
Temperature low, internal calibration recommended	New calibration is requested
Temperature high, internal calibration recommended	New calibration is requested
WARNING Over temperature	Switch the generator off, read chapter 6 in this operating manual

Locking	Remarks
Cannot lock parameter, out of range	
Unlock parameter before changing	
Cannot lock parameter with volt error	

Tracking	**Remarks**
Cannot change track status of locked parameter	Unlock
Parameter not trackable	Read: Track in Reference
Illegal time track relation (*Period* $\geq$ 2 $*$ *Delay*)	Read: Track in Reference
Tracking removed, track combination impossible	Read: Track in Reference
Tracking removed from inactive parameter(s) out of range	Read: Track in Reference
Cannot track parameter with error	

Auto Set	**Remarks**
Auto setting	
Auto setting, No error present	
Auto set failed	Correct the setting manually

Settings	**Remarks**
Selected setting deleted	
All settings deleted	
Cannot save, active setting erroneous	Read: Save in Reference
Memory 0 is read only	Select another number (0=Default)
Memory number out of range (0...49)	Select another number
Setting not defined	
Sequence empty	
Sequence full	
Sequence array x deleted	
Checksum error in setting	Selected setting has been destroyed and can not be recalled (Service)
Checksum error in active setting	Active setting is destroyed (Service)

Volt Limits	Remarks
Volt limits impossible, UpperLimit-LowerLimit > 0.20 Volt	The difference must be more than 0.2 V
Volt limits in conflict with current setting	Check the limits and the current setting

Calibration	Remarks
Wrong access code ! Try again	
Access codes not equal ! Try again	
Internal calibration system not installed	
Warm up not completed Adjustment not recommended	Wait
Warm up not completed Verification not recommended	Wait
Internal calibration voltage reference uncalibrated default constants used	New calibration is requested for volt references
Adjustment completed with XX errors	Read internal calibration in Reference
Internal adjustment error	
Verification completed with XX errors	Read internal calibration in Reference
CalErr 1: Parameter: LEADEDGE range 4	Hardware adjustment req. (service)
CalErr 0: No more errors	Read internal calibration in Reference
DAC-value calculation error in XX point	Hardware adjustment req. (service)
Inaccuracy error in XX point	Hardware adjustment req. (service)
Measuring error in XX point	Hardware adjustment req. (service)
Verification, Adjustment not allowed	
Internal verification/adjustment aborted	
Default calibration constants set	The accuracy is now poor
Both range end points out of range	Hardware adjustment req. (service)
Low range end point out of range	Hardware adjustment req. (service)
High range end point out of range	Hardware adjustment req. (service)

Calibration	**Remarks**
Illegal combination of DAC-values	Hardware adjustment req. (service)
WARNING ! Calibration data save error	Report to service
WARNING ! Stack overflow	Report to factory

GPIB

Introduction

The following list contains all error messages that can occur during GPIB control of the instrument. The instrument shows the error code and message on the display* and beeps when an error occurs. The message disappears as soon as the instrument need the display to show other information. However, the message is available in the error queue that you can read via the ERROR? command, see the 'Command reference' chapter.

Errors in instrument functions begin with 1 and errors in the GPIB begins with 2.

Instrument functions errors

Error code	Message	Explanation
10000	Parameter out of range	An attempt was made to set a pulse parameter or trig level outside it's legal range
10001	Volt limit error	An attempt was made to set output voltages outside the voltage limits
10002	Coupled error between time parameters	
10003	Coupled error between volt parameters	

---

* Displaying error messages can be enabled/disabled with the ERROR DISP command.

Error code	Message	Explanation
10004	Cannot change locked parameter	
Trigger errors		
10005	Illegal trig or pulse mode combination,	
10006	ManTrig or ManGate failed, wrong trig mode	Only allowed with trigsource manual
10007	ManGate? failed, wrong trig mode	Only allowed with trigsource manual and trigfunction gate or duration
Output errors		
10008	Output On without specifying channel(s) not allowed	
Saved setting errors		
10009	Memory number out of range	Legal range 1..49 for *SAV and MemProg, 0..49 for *RCL
10010	Memory 0 is read only	Contains the default setting which cannot be overwritten
10011	Save failed, active setting erroneous	Erroneous settings can not be saved, the setting was probably erroneous due to ErrorCheck Off
10012	Recall failed, setting not defined	
10013	Next or Previous failed, sequence or memory empty	
Data type errors		
10014	Block data size error	A receive of BinProg, MemProg or Caldata failed
10015	Block data checksum error	A receive of BinProg, MemProg or Caldata failed
10016	Block data version error	A receive of BinProg, MemProg or Caldata failed

Error code	Message	Explanation
10017	BinProg? failed, active setting erroneous	Erroneous settings can not be sent as response to BinProg, the setting was probably erroneous due to ErrorCheck Off
10018	MemProg? failed, cannot send default setting,	
Track errors		
10019	Track On without specifying parameter(s) not allowed	
10020	Parameter locked, cannot change track status	
10021	Illegal time parameter tracking relation	The period time has to be larger than two times the delay time
10022	/*Not used*/	
10023	/*Not used*/	
Increment error		
10024	Parameter increment failed	The increment would have resulted in an erroneous setting
Volt limit errors		
10025	Volt limit(s) out of range	The voltage limits has to be within +/-10V
10026	Volt limits impossible, difference less than min. amplitude	
10027	Volt limits in conflict with current setting	Adjust the active setting before setting the voltage limits
Lock error		

Error code	Message	Explanation
10028	Lock On without specifying parameter(s) not allowed	
Autoset error		
10029	Autoset failed	
10030	/*Not used*/	
10031	/*Not used*/	
10032	/*Not used*/	
Miscellaneous errors		
10033	Attempt to change protected data without removing protection	Use the 'Unprotect access code' to gain access to protected data. The code is set to 1875 when the instrument leaves the factory.
10034	Service commands off	
10035	Invalid time string	
Test program errors		
10036	LCD test failed	Report to service
10037	Keyboard/led test failed	Report to service
10038	PROM test failed	Report to service
10039	RAM 1 test failed	Report to service
10040	RAM 2 test failed	Report to service
10041	Counter test failed	Report to service
10042	Real time clock test failed	Report to service
10043	Uncalibrated, default calibration constants used	New calibration requested
10044	Bad calibration status	See reference about calibration errors
10045	Calibration data lost	Battery empty? report to service

Error code	Message	Explanation
10046	Calibration data save error	Report service. The part of the ram where the calibration data is saved didn't verify after a calibration operation. Notice! This part of the ram is not covered by the ram test
10047	Internal calibration system not installed	
10048	GPIB address and sequences lost	Battery empty? report to service
10049	Checksum error in active setting	Active setting is destroyed (Service)
10050	Recall failed, checksum error in setting	Selected setting has been destroyed and can not be recalled (Service)
10051	Default setting recalled	Probably due to an erroneous setting set with errorcheck off
10052	Over temperature	Switch the generator off, read chapter 6 in this operating manual
10053	Stack overflow	Report to factory
10054-10056	/*Not used*/	
10057	Internal verification/adjustment aborted	The reset button on the front panel was pushed during a calibration operation
10058	Internal calibration voltage reference uncalibrated, default constants used	Recalibrate the voltage reference before using the internal calibration system
10059	Internal calibration voltage reference calibration data lost	Battery empty? report to service

Bus errors

Error code	Message	Explanation
20000	Readout attempt from idle state.	A GPIB read was attempted when the GPIB protocol was in idle state. Make sure the readout is preceded by a valid query.
20001	Readout attempt preceded by non-query	Make sure to send a valid query to the instrument before attempting a readout.
20002	Readout attempt with unterminated query.	(Diagnostic message.) The instrument did not receive a message terminator with the last message, yet a readout was attempted.
20003	New message in while sending response.	(Diagnostic message.) A readout was started but not completed before the instrument received the next message.
20004	New message in without response readout.	(Diagnostic message.) The instrument received a query, then it received a new message instead of the expected readout of the response to the query.
20005	New message in while sending response.	(Diagnostic message.) A response was waiting in the output buffer, and readout may have started, but then a new message arrived before the readout was done.
20006	Bus interface deadlock occurred	A deadlock occurred because of bytes arriving when both input buffer and output buffer where full, and the full output buffer blocked the response formatter, execution control and parser. Query responses have been discarded.
20007	(Software error) Bus interface illegal state	(This error should never occur.)
20008	Reset during bus input	(Diagnostic message.) The parser was reset while waiting for more characters to arrive over the bus.

Error code	Message	Explanation
20009	Reset during bus output	(Diagnostic message.) The response formatter was reset while waiting for the controller to read out characters from the output buffer.
20010	Exponent too large for conversion to float	The exponent received as part of a floating point number was outside the single precision floating point range used for internal representation. (Approx 10 to –38, to 10 to +38.)
20011	Expected closing parenthesis for expression	There seems to be a closing parenthesis missing from an expression data element.
20012	Non-numeric macro parameter used in expression	A non-numeric parameter sent with a macro corresponded to a parameter placeholder in an expression data type. This is not allowed.
20013	Unexpected element type in expression	An element other than numeric or simple pulse parameter mnemonic was found in an expression.
20014	Float underflow in expression	A floating point underflow occurred while calculating the value of an expression data element.
20015	Float overflow in expression	A floating point overflow occurred while calculating the value of an expression data element.
20016	Invalid float operation in expression	An unsupported floating point operation was attempted in an expression.
20017	(Software error) Parser called when disabled	(This error should never occur.)
20018	Unexpected end of message	The instrument found a program message terminator while reading a string or definite length block data, before the end of the data element.

Error code	Message	Explanation
20019	Too long mantissa - more than 255 characters	Excluding leading zeroes, a floating point mantissa may at most contain 255 characters.
20020	Illegal character in message	
20021	Invalid message header found	
20022	Invalid numeric data found	
20023	Unrecognized data type found	The message contained a data element that could not be classified as a known data type.
20024	Missing data element in message unit	The format for a command in the message specifies more data elements than given.
20025	Too many data elements in message unit	The format for a command in the message specifies fewer data elements than given.
20026	Invalid data type in message unit	
20027	Invalid block data format	
20028	Unrecognized data mnemonic	The instrument does not recognize a mnemonic in a data field.
20029	Too large exponent (outside –32000 to 32000)	
20030	Macro recursion detected, execution interrupted	A macro was called recursively, directly or indirectly.
20031	End of data list found when comma expected	A command contained fewer comma-separated data fields than expected. Correct the syntax.

Error code	Message	Explanation
20032	Several data elements missing in message unit	
20033	New data field found when comma expected	Two apparent data fields followed each other without a separating comma.
20034	(Software error) Invalid input character class	(This error should never occur.)
20035	(Software error) Error in data type search logic	(This error should never occur.)
20036	(Software error) Error in data type search logic	(This error should never occur.)
20037	Malformed parameter placeholder found	A parameter placeholder should have the form of a '$' followed by one of the digits 1–9.
20038	Macro placeholder numbers are not contiguous	Placeholder numbers should occur in a contiguous series starting with 1.
20039	Parameter count does not match macro definition	
20040	Numeric data ASCII to real conversion error	A conversion of a numeric data field failed.
20041	Response formatter called when no query	(This error should never occur.)
20042	Indefinite length arbitrary block not last in response	You must place a query unit which will generate an indefinite length arbitrary block data as the last query in its program message unit.
20043	Invalid function code sent to response formatter	(This error should never occur.)

Error code	Message	Explanation
20044	Invalid header type sent to response formatter	(This error should never occur.)
20045	Invalid data type sent to response formatter	(This error should never occur.)
20046	ASCII to float conversion overflow	
20047	ASCII to float conversion underflow	
20048	Hash table overflow	(This error should never occur.)
20049	Hash memory initialization failure	The device RAM used for the hash tables
20050	Wrong hash table checksum found	The hash table has probably been corrupted. After this error, the macro table is purged and rebuilt.
20051	Wrong macro table checksum found	The macro table has probably been corrupted. After this error, the macro table is purged.
20052	Attempt to change protected data without removing protection	Use the 'Unprotect <access code>' to gain access to protected data. The code is set to 1875 when the instrument leaves the factory.
20053	PUD block too long (more than 64 bytes)	
20054	*DDT definition too long	
20055	Macro label table is full	You have attempted to define more than 20 macro's.
20056	Insufficient room for macro definition	The total length of macro definitions exceeded the memory available for definitions.
20057	Macro definition too long (more than 255 bytes)	

Error code	Message	Explanation
20058	Macro label too long (more than 12 characters)	
20059	Illegal character in macro label	Macro labels obeys the same limitations as other mnemonic headers.
20060	Attempt to redefine macro	
20061	Service commands off	You have to turn service commands on before you use them.
20062	Access code too long	
20063	Access code did not match	
20064	Access code not numeric	The access code has to consist of 0 to 12 decimal digits.
20065	Access code checksum error, default set	The access code checksum was wrong, the access code may have been corrupted. The factory default has been set.

Appendix

Common Commands Summary

*AAD	Accept address command*
*CAL?	Calibration query
*CLS	Clear status command
*DDT	Define device trigger command
*DDT?	Define device trigger query
*DLF	Disable listener function command*
*DMC	Define macro command
*EMC	Enable macro command
*EMC?	Enable macro query
*ESE	Standard event status enable command
*ESE?	Standard event status enable query
*ESR?	Standard event status register query
*GMC?	Get macro contents query
*IDN?	Identification query
*IST?	Individual status query*
*LMC?	Learn macro query
*LRN?	Learn device setup query
*OPC	Operation complete command
*OPC?	Operation complete query
*OPT?	Option identification query
*PCB	Pass control back command*
*PMC	Purge macro command
*PRE	Parallel poll register enable command*
*PRE?	Parallel poll register enable query*
*PSC	Power on status clear command

* Not used in PM 5781

*PSC?	Power on status clear query
*PUD	Protected user data command
*PUD?	Protected User Data Query
*RCL	Recall command
*RDT	Resource description transfer command*
*RDT?	Resource description transfer query*
*RST	Reset command
*SAV	Save command
*SRE	Service request enable command
*SRE?	Service request enable query
*STB?	Read status byte query
*TRG	Trigger command
*TST?	Self-test query
*WAI	Wait-to-continue command

---

* Not used in PM 5781

Couplings

This section describes the relations between the different settings of the instrument. The relations are expressed as mathematical formulas which can be programmed into the controller program to avoid conflicts.

Trig and pulse modes

Externally and manually triggered square pulse is not allowed. Internal duration is not allowed.

Time parameters

- *Couplings depending on the pulse mode*

Square pulse

$Period \geq 2.5 * Leading\ edge$
$Period \geq 2.5 * Trailing\ edge$

Single pulse

$Period \geq 2 * (delay - 4 * 10^{-9})$
$Period \geq 2 * Duration$
$Period \geq Duration + 1.25 * Trailing\ edge$
$Duration \geq 1.25 * Leadingedge$

Double pulse

$Period \geq 2 * Delay$
$Delay \geq 2 * Duration$
$Delay \geq Duration + 1.25 * Trailing\ edge$
$Duration \geq 1.25 * Leadingedge$

- *Couplings depending on the trig mode*

External and manual trig

All pulse mode formulas above apply, except those containing period, since period is inactive.

External and manual duration

None of the pulse mode formulas apply since period, delay and duration is inactive.

Internal burst

$Burst\ period \geq Burst * Period + 2 * 10^{-7}$

Internal gate

$Burst\ period \geq Period * 0.55$

- *Transition time couplings*

Leading and trailing edges must be within the same range.

Voltage parameters

$High\ level - Low\ level \geq 0.10$
$High\ level - Low\ level \leq 10.2$
$Offset\_B \leq 10.1 - High\ level$
$Offset\_B \geq -10.1 - Low level$

This page is intentionally left blank.

Index

D

E

F

G

H

I

K

L

M

N

O

Hinditron Services Pvt. Inc.
1st Floor, 17-B,
Mahal Industrial Estate
Mahakali Road, Andheri
East
Bombay 400093
Phone: 91 22 630-0043
Fax: 91 22 836-4682

Hinditron Services Pvt. Ltd.
204-206 Hemkunt Tower
98 Nehru Place
New Delhi 110 019
Phone: 91 011 643-3675

Hinditron Services Pvt. Ltd.
Field Service Center
Emerald House
5th Floor
114 Sarojini Devi Road
Secunderabad 500 003
Phone: 91 842-844033

Peico Electronics & Electricals Ltd.
Band Box House
254 Dr Annie Besant Road
Bombay 400 025
Phone: 91 22 851-0261
or 493-0590
Fax: 91 22 494-1698

Indonesia

P.T. Daeng Brothers
Philips House
J/n H.R. Rasuna Said Kav. 3-4
Jakarta 12950
Phone: 62 21 520 1122
Fax: 62 21 520 5189

Israel

R.D.T Electronics Engineering, Ltd.
P.O. Box 58013
Tel Aviv 61580
Phone: 972-3-548-3737
Fax: 972-3-492190

Ireland

Fluke (UK) Ltd
Colonial Way
Watford
HERTS WD2 4TT England
Phone: 44 923 240511
Fax: 44 923 225067

Italy

Fluke Italia s.r.l.
Viale Casati 23
20052 Monza (MI)
Phone: 39 39 203 6525
Fax: 39 39 203 6619

Japan

Fluke Corporation
Sumitomo Higashi Shinbashi Bldg. 1-1-1
1 Hamamatsucho
Minato-ku
Tokyo 105
Phone: 81 3 3434-0181
Fax: 81 3 3434-0170

Korea

Il Myoung, Inc.
780-46, Yeogsam-Dong
Youngdong P .O. Box 1486
Kangnam-Ku
Seoul
Phone: 82 2 552-8582-4
Fax: 82 2 553-0388

B&P International Co.,Ltd.
Geopyung Town A-1809
203-1 Nonhyun-dong
Kangnam-Ku
Seoul 135-010
Phone: 82 2 546-1457
Fax: 82 2 546-1458

Malaysia

CNN. SDN. BHDM
17D, 2nd Floor
Lebuhraya Batu Lancang
Taman Seri Damai 1
1600 Jelutong Penang
Phone: 60 4 879584
Fax: 60 4 870835

Mexico

Mexel Mexicana De Electronica
Industrial, S.A. De C.V.
Diagonal #27.
Entre Calle de Eugenia Y Av.
Col. Del Valle
C.P. 03100, Mexico D.F.
Phone: 52 5 682-8040
Fax: 52 5 687-8695

Mexel Mexicana De Electronica
Industrial, S.A.
Av. Porvenir No.8608
Centro C San Martin Local 6
CD. Jaurez, Chihuahua.
Phone: 52 16 23-0235
Fax: 52 16 23-0235

Netherlands

Fluke Nederland B.V
Techn. Service
Hurksestraat 2C,
Building HBR
5652 AJ Eindhoven
Phone: 31 40 723220
Fax: 31 40 723337

New Zealand

Philips Customer Support
Scientific & Industrial
Private Bag 41904
St Lukes, 2 Wagener Place
Mt. Albert, Auckland 3
Phone: 64 9 849-4160
Fax: 64 9 849-7814

Norway

Fluke Norge A/S
Ole Deviks vei 2
N 0666 Oslo
Phone: 47 22 653400
Fax: 47 22 653407

Peru

Importaciones & Representaciones Electronicas S.A.
Jr. Pumacahua 955
Lima 11
Phone: 51 14 23-5099
Fax: 51 14 31-0707

Philips Peruana S.A.
Planta Caretera Central
Av. Comandante Espinar 719
Lima 18
Phone: 51 14 35-0059
Fax: 51 14 35-4200

Philippines

Spark Electronics Corp.
P.O. Box 610, Greenhills
Metro Manila 1502
Phone: 63 2 700621
Fax: 63 2 7210491

Portugal

Fluke Iberica, S.L.
Rua Dr. A. Loureiro Borges 5
Arquiparque - Miraflores
P.O. Box 300
2795 LINDA-A-VELHA
Phone: 351 1 416-3316
Fax: 351 1 416-3185

Republic of South Africa

South African Philips (Pty) Ltd.
195/215 Main Road
P.O. Box 58088
Martindale 2092
Phone: 27 11 470-5132
Fax: 27 11 470-5206

Singapore

Fluke Corporation
#27-03 PSA Building
460 Alexandra Road
Singapore 0511
Phone: 65 276-5161
Fax: 65 276-5759

Spain

Fluke Iberica, S.L.
C/Martinez Villergas, 2
28027 Madrid
Phone: 34 1 326 7512
Fax: 43 1 326 0668

Sweden

Fluke Sverige AB
Kronborgsgränd 11
Box 61
S-164 94 Kista
Phone: 46 8 751 0230
Fax: 46 8 751 0480

Switzerland

Fluke Switzerland AG
Riedstrasse 12
Postfach 360
CH-8953 Dietikon 1
Phone: 41 1 745 2244
Fax: 41 1 745 2240

Taiwan, R.O.C.

Schmidt Electronics Corp.
5th Floor, Cathay Min Sheng
Commercial Building, No. 180
Sec. 2
344 Min Sheng East Road
Taipei
Phone: 886 2 501-3468
Fax: 886 2 502-9692

Thailand

Measuretronix Ltd.
2102/31 Ramkamhaeng Road
Bangkok 10240
Phone: 66 2 375-2733 or
375-2734
Fax: 66 2 374-9965

Philips Electrical Co. of Thailand Ltd.
283, Silom Road
P.O. Box 961
Bangkok 10500
Phone 66 2 233-63309

Turkey

Turk Philips Ticaret A.S.
Inonu Caddesi 78/80
Posta Kutusu 504-Beyoglu
Istanbul
Phone: 90 1 143-5891

United Kingdom

Fluke (UK) Ltd
Colonial Way
Watford
HERTS WD2 4TT
Phone: 44 923 240511
Fax: 44 923 225067

Uruguay

Coasin Uruguaya S.A.
Casilla de Correo 1400
Libertad 2529
Montevideo
Phone: 598-2-789015
Fax: 598 2 797338

Venezuela

Coasin C.A.
Calle 9 Con Calle 4, Edif. Edinurbi
Apartado de Correos
Nr-70-136
Los Ruices
Caracas 1070-A
Phone: 58 2 241-0309, or
241-1248
Fax: 58 2 241-1939

Countries not listed

Fluke Corporation
P.O. Box 9090
Mail Stop 268C
Everett, WA 98206-9090

USA

California

Fluke Service Center
46610 Landing Parkway
Fremont, CA 94538
Phone: 510-651-5112
Fax: 510-651-4962

Fluke Service Center
16715 Von Karman Avenue
Suite 110
Irvine, CA 92714
Phone: 714-863-9031
Fax: 714-863-17263

Florida

Fluke Service Center
550 S. North Lake Blvd.
Altamonte Springs,
FL 32701-5227
Phone: 407-331-2929
Fax: 407-331-3366

Illinois

Fluke Service Center
1150 W. Euclid Avenue
Palatine, IL 60067
Phone: 708-705-0500
Fax: 708-705-9989

New Jersey

Fluke Service Center
West 75 Century Rd.
Paramus, NJ 07652-0930
Phone: 201-599-9500
Fax: 201-599-2093

Texas

Fluke Service Center
2104 Hutton Drive
Suite 112
Carrollton, TX 75006
Phone: 214-406-1000
Fax: 214-406-1072

Washington

Fluke Service Center
Fluke Corporation
Building #4
1420 - 75th St.S.W
M/S 6-30
Everett WA 98203
Phone: 206-356-5560
Fax: 206-356-6390

INTERNATIONAL

Argentina

Coasin S.A.
Virrey del Pino 4071 DPTO
E-65
1430 CAP FED
Buenos Aires
Phone: 54 1 522-5248
Fax: 54 1 551-1767

Australia

Philips Customer Support
Scientiific and Industrial
23 Lakeside Drive
Tall Ho Technology Park
East Burwood
Victoria 3151
Phone: 61 3 881-3666
Fax: 61 3 881-3636

Philips Customer Support
Scientific and Industrial
Block F, Centrecourt
34 Waterloo Road
North Ryde, N.S.W 2113
Phone: 61 2 888-0416
Fax: 61 2 888-0440

Austria

Fluke Österreich G.m.b.H.
Gutheil-Schoder-Gasse 10
A-1102 Wien
Phone: 43 1 60101-1568
Fax: 43 1 603 2165

Belgium

N.V Fluke Belgium S.A.
Langeveldpark - Unit 5 & 7
P. Basteleusstraat 2-4-6
1600 Sint-Pieters-Leeuw
B-1070 Brussels
Phone: 32 2 331 2777
Fax: 32 2 331 1489

Brazil

Hi-Tek Electronica Ltda.
Al. Amazonas 422,
Alphaville
06454-070 Barueri
Sao Paulo
Phone: 55 11 725-5822
Fax: 55 11 421-5032

Philips Industrial Electronics
Philips Medical Systems
Ltda.
Av. Interlagos. N.3493
Campo Grande
04661-200 Sao Paulo
Phone: 55 11 523-4811
Fax: 55 11 524-4873

Canada

Fluke Electronics Canada Inc.
400 Britannia Road East,
Unit #1
Mississauga, Ontario
L4Z 1X9
Phone: 905 890-7600
Fax: 905 890-6866

Chile

Intronsa Inc.
Casilla 16150
Santiago 9
Phone: 56 2 232-1886 or -4308
Fax: 56 2 232-2694

China

Fluke Service Center
Room 2111 Scite Tower
Jianguomenwai Dajie
Beijing 100004, PRC
Phone: 86 1 512-3435 or -6351
Fax: 86 1 512-3437

Colombia

Sistemas E Instrumentacion, Ltda.
Carrera 21, NO. 39A-21, Of. 101
Ap. Aereo 29583
Bogota
Phone: 57 1 287-5424
Fax: 57 1 287-2248

Industrias Philips de Colombia
Apartado Aereo 4282
Calle 13 No. 51-39
Bogota
Phone: 57 1 260-0600
Fax: 57 1 261-0139

Costa Rica

Electronic Engineering, S.A.
Carretera de Circunvalacion
Sabanilla Av. Novena
San Jose
Phone: 506-53-3759
Fax: 506-25-1286

Denmark

Fluke Danmark A/S
Ejby Industrivej 40
2600 Glostrup
Phone: 45 43 44 1900
Fax: 45 43 43 9192

Ecuador

Proteco Coasin Cia., Ltda.
P.O. Box 17-03-228-A
Ave. 12 de Octubre
2449 y Orellana
Quito
Phone: 593 2 230283 or 520005
Fax: 593 2 561980

Philips Equador S.A.
Dpto. de Systemas
Medicos/Prof
Av. Amazonas
1188Y Cordero
Phone: 593 2 565835
Fax: 593 2 564601

Egypt

Philips Egypt
10, Abdel Rahman el Rafei st.
el. Mohandessin
P.O. Box 242
Dokki Cairo
Phone: 20 2 490922

Federal Republic of Germany

Fluke Deutschland GmbH.
Miramstrasse 87
34123 Kassel
Phone: 49 561 501 1495
Fax: 49 561 501 1690

Fluke Deutschland GmbH.
Regional Repair Center
Oskar-Mesker-Strasse 18
85737 Ismaning/Munich
Phone: 49 89 960 5261
Fax: 49 89 960 5270

Fluke Deutschland GmbH
Regional Repair Center
Meiendorfer Strasse 205
D-2000 Hamburg 73
Phone: 49 40 6797-434
Fax: 49 40 6797-421

Finland

Fluke Finland Oy
Sinnikaliontie 3
SF-02631 Espoo
Phone: 358 0 5026 600
Fax: 358 0 5026 414

France

Fluke France S.A.
37 Rue Voltaire
93700 Drancy
Phone: 33 1 4896-6331
Fax: 33 1 4896-6330

Greece

Philips S.A. Hellenique
15,25th March Street
177 78 Tavros
10210 Athens
Phone: 30 1 489-4911
Fax: 30 1 481-5180

Hong Kong

Philips Hong Kong Ltd.
IE Systems Division
Site 7, Ground Floor
Whampoa Garden
Hung Hom, Kowloon
Wanchai
Phone: 852 773-5588
Fax: 852 334-5496

Schmidt & Co (H.K.) Ltd.
1st Floor, 323 Jaffe Road
Wanchai
Phone: 852 9223-5623
Fax: 852 834-1848

India

Hinditron Services Pvt. Inc.
33/44A Raj Mahal Vilas
Extension
8th Main Road
Bangalore 560 080
Phone: 91 80 348266
Fax: 91 80 345022