

Cursor

Parameter_Custom, PACU
Command/Query

Description

The Parameter_Custom command controls the timing parameters. Only these parameters are described here. However, statistical and other standard parameters are described in the instrument's *Remote Control Manual*.

The measured value of a parameter that was set up with **PACU** may be read using **PAVA**.

<param>	definition:	<qualifier> list:
DPLEV	delta period at level (Clock)	<source>,<input_type>,<edge>,<level>
	delta period at level (Data)	<source>,<input_type>,<edge>,<level>,<for_all>,<ref_type>,<freq>
DULEV	duty cycle at level	<source>,<edge>,<level>
EDLEV	delta delay	<source>,<edge>,<level>
PLEV	period at level (Clock)	<source>,<input_type>,<edge>,<level>
	period at level (Data)	<source>,<input_type>,<edge>,<level>,<for_all>,<ref_type>,<freq>
FREQLEV	frequency at level (Clock)	<source>,<input_type>,<edge>,<level>,<for_all>,<ref_type>,<freq>
	frequency at level (Data)	<source>,<input_type>,<edge>,<level>,<for_all>,<ref_type>,<freq>
TIELEV	time interval error at level (Clock or Data)	<source>,<input_type>,<edge>,<level>,<scale>,<for_all>,<ref_type>,<freq>
WIDLEV	width at level	<source>,<edge>,<level>

Where:

<source>: = {C1, C2, C3, C4, TA, TB, TC, TD}

<input_type>: = CLK when Clock mode is to be used

: = DATA when Data mode is to be used

<edge>: = {POS, NEG}

<level>: = 1 to 99 if level is specified in percent (PCT). If PCT is not specified, the level is in the units of the <source> waveform.



<for_all>: = {YES, NO}
<scale>: = {UI, S}
<ref_type>: = {STD, CUST}
CUST <freq>: = 10 to 1e9 Hz
STD <freq>: = {1.5M, 2M, 8M, 34M, 44M, 52M, 139M, 155M}

Command Syntax

PParameter_Custom <line>, <parameter>, <qualifier>,
[<qualifier>, ...]
<line>: = 1 to 5
<parameter>: = {a parameter from the table}
<qualifier>: = Measurement qualifier(s) specific to each.

Note:

CUST1, CUST2, CUST3, CUST4, CUST5 refer to the custom line numbers of the selected parameters.

Query Syntax

PParameter_CUstom? <line>

Response Format

PParameter_CUstom <line>, <parameter>, <qualifier>
[,<qualifier>,...]

Availability

<source>: = {C3, C4} available only on four-channel instruments.

Appendix C: Remote Control Commands

Examples

DPLEV

Command:

1. PACU 1, DPLEV, TA, DATA, POS, 0E-3 V, 0.5 DIV, NO, CUST, 1.125E+8 HZ
2. PACU 1, DPLEV, TA, CLK, POS, 0E-3 V, 0.5 DIV

Query/Response:

PACU? 1 returns:

- 1: PACU 1, DPLEV, TA, DATA, POS, 0E-3 V, 0.5 DIV, NO, CUST, 1.125E+8 HZ
- 2: PACU 1, DPLEV, TA, CLK, POS, 0E-3 V, 0.5 DIV

PAVA? CUST1 returns:

C2: PAVA CUST1, 1E-9 S, OK

Examples

TIELEV

Command:

PACU 1, TIELEV, TA, DATA, POS, 1 PCT, 0.5 DIV, UI, YES, STD, 34M

Query/Response:

PACU? 1 returns:

PACU 1, TIELEV, TA, DATA, POS, 1 PCT, 0.5 DIV, UI, YES, STD, 34M

Related Commands

- Parameter_Delete
- Parameter_Value?
- Parameter_Statistics
- Parameter_Clr
- Pass_Fail_xxx



Function

Define, DEF Command/Query

Description

The Define command controls JitterPro and JTA functions. This command is also used to control all functions in the standard instruments, as well as the statistical functions common to JitterPro, JTA, and WP03 options. (See the *Remote Control Manual*.)

Command Syntax

<function> : DEFine EQN, "<equation>," [<attribute_name>,<value>,...]

Note:

Function attributes are grouped in pairs. The first in the pair, <attribute_name>, names the variable to be modified, while the second one <value>, gives the new value to be assigned. Pairs can be given in any order.

Query Syntax

<function>: DEFine?

Response Format

<function>: DEFine EQN,"<equation>," [<attribute_name>,<value>,...]

Availability

<sourceN>: C3 and C4 — available only on four-channel instruments.

Related Commands

Find_Ctr_Range, Function_Reset, Inr?

Appendix C: Remote Control Commands

Jitter (Cycle-to-Cycle—) <function>: Def Eqn, "JitterCC (<source>)", Slope, <slope>, Level, <level>, Input_Type, <input_type>, [For_All, <for_all>, Freq_Std, <std_freq> (or Freq, <cust_freq>)]

Where:

<source>: = {C1, C2, C3,C4,TA, TB, TC, TD}
<function>: = {TA, TB, TC, TD}
<slope>: = {POS, NEG}
<level>: = 1 to 99 if level is specified in percent (PCT), or level in the units of the <source> waveform
<input_type>: = {CLK, DATA}
<for_all>: = {YES, NO}
<std_freq>: = {1.5M, 2M, 8M, 34M, 44M, 52M, 139M, 155M}
<cust_freq>: = 1 to 1e+9 HZ

Note: Scaling of the JitterTrack™ function Find Jitter Range can be executed using the Find_Ctr_Range (FCR) command. Example: TA:FCR scales JitterTrack on Trace A.

Command Examples

Clock Input Type:	TA:DEF EQN, "JITTERCC(C1)," SLOPE, POS, LEVEL, 0E-3, HYST, 0.5 DIV, TYPE, CLK
Data Input Type, with custom frequency:	TA:DEF EQN, "JITTERCC(C1)," SLOPE, POS, LEVEL, 0E-3, 0.5 DIV, TYPE, DATA, FOR_ALL, YES, FREQ, 1 .125e+08 HZ
Data Input Type, with standard frequency:	TA:DEF EQN, "JITTERCC(C1)," SLOPE, POS, LEVEL, 0E-3, 0.5, TYPE, DATA, FOR_ALL, YES, FREQ_STD, 1.5M

Query/Response Examples TA:DEF? returns a string similar to the command examples.



Jitter (Duty Cycle–)

<function>: Def Eqn, "Jitterduty (<source>)," Slope, <slope>,
Level, <level>

Where:

<source>: = {C1, C2, C3,C4,TA, TB, TC, TD}

<function>: = {TA, TB, TC, TD}

<slope>: = {POS, NEG}

<level>: = 1 to 99 if level is specified in percent (PCT), or level in
the units of the <source> waveform

Note:

Scaling of the JitterTrack™ function Find Jitter Range can be
executed using the Find_Ctr_Range (FCR) command.
Example: TA:FCR scales JitterTrack on Trace A.

Command Examples

Absolute level: TA:DEF EQN, "JITTERDUTY(C2)," SLOPE, POS, LEVEL,
213.9E-3, 0.5

Relative level: TA:DEF EQN, "JITTERDUTY (C2)," SLOPE, POS, LEVEL, 42
PCT, 0.5

Query/Response Examples TA:DEF? returns:

Absolute level: TA:DEF EQN, "JITTERDUTY (C2)," SLOPE, POS, LEVEL,
213.9E-3 V, 0.5

Relative level: TA:DEF EQN, "JITTERDUTY (C2)," SLOPE, POS, LEVEL, 42
PCT, 0.5

Appendix C: Remote Control Commands

Jitter (Interval Error–)

<FUNCTION>: DEF EQN, "JITTERIE (<SOURCE>)," SLOPE, <SLOPE>, LEVEL, <LEVEL>, VUNIT, <VUNIT>, INPUT_TYPE, <INPUT_TYPE>, FOR_ALL, <FOR_ALL>, FREQ_STD, <STD_FREQ> (OR FREQ, <CUST_FREQ>)

Where:

<source>: = {C1, C2, C3, C4, TA, TB, TC, TD}
<function>: = {TA, TB, TC, TD}
<slope>: = {POS, NEG}
<level>: = 1 to 99 if level is specified in percent (PCT), or level in the units of the <source> waveform
<vunit>: = {UI, S}
<input_type>: = {CLK, DATA}
<std_freq>: = {1.5M, 2M, 8M, 34M, 44M, 2M, 139M, 155M}
<cust_freq>: = 10 to 1e9 Hz.

Note:

Scaling of the *JitterTrack*™ function Find Jitter Range can be executed using the Find_Ctr_Range (FCR) command. Example. TA:FCR scales *JitterTrack* on Trace A.

Command Examples

Clock Input Type, in UI,
and custom frequency:

```
TA:DEF EQN, "JITTERIE(C2)," SLOPE, POS, LEVEL, -6E-3V, 0.5,  
VERT_UNIT, UI, TYPE, CLK, FREQ_GLOBAL, YES, FREQ, 1.99987e+06  
HZ
```

Clock Input Type, in S,
with custom frequency:

```
TA:DEF EQN, "JITTERIE(C2)," SLOPE, POS, LEVEL, -6E-3V, 0.5,  
VERT_UNIT, S, TYPE, CLK, FREQ_GLOBAL, YES, FREQ, 1.99987e+06
```

Data Input Type, in S,
with standard frequency:

```
TA:DEF EQN, "JITTERIE(C2)," SLOPE, POS, LEVEL, -6E-3V, 0.5,  
VERT_UNIT, S, TYPE, DATA, FREQ_GLOBAL, YES, FREQ_STD, 2M
```

Query/Response Examples TA:DEF? returns a string similar to the command examples.



Jitter (Period--)

<function>: Def Eqn, "Jitterp (<source>)," Slope, <slope>, Level, <level>, Input_Type, <input_type>,[For_All, <for_all>, Freq_Std, <std_freq> (or Freq, <cust_freq>)]

Where:

<source>: = {C1, C2, C3, C4, TA, TB, TC, TD}
<function>: = {TA, TB, TC, TD}
<slope>: = {POS, NEG}
<level>: = 1 to 99 if level is specified in percent (PCT), or level in the units of the <source> waveform
<input_type>: = {CLK, DATA}
<for_all>: = {YES, NO}
<std_freq>: = {1.5M, 2M, 8M, 34M, 44M, 52M, 139M, 155M}
<cust_freq>: = 1 to 1e+9 HZ

Note:

Scaling of the *JitterTrack*™ function Find Jitter Range can be executed using the Find_Ctr_Range (FCR) command. Example. TA:FCR scales *JitterTrack* on Trace A.

Command Examples

Clock Input Type: TA:DEF EQN, "JITTERP(C1)," SLOPE, POS, LEVEL, 0E-3, 0.5 DIV, TYPE, CLK

Data Input Type, with custom frequency: TA:DEF EQN, "JITTERP(C1)," SLOPE, POS, LEVEL, 0E-3, 0.5 DIV, TYPE, DATA, FOR_ALL, YES, FREQ, 1.125e+08 HZ

Data Input Type, with standard frequency: TA:DEF EQN, "JITTERP(C1)," SLOPE, POS, LEVEL, 0E-3, 0.5, TYPE, DATA, FOR_ALL, YES, FREQ_STD, 1.5M

Query/Response Examples TA: DEF? returns a string similar to the command examples.

Appendix C: Remote Control Commands

Jitter (Frequency—)

<function>: Def Eqn, "Jitterfreq (<source>)," Slope, <slope>,
Level, <level>, Input_Type, <input_type>, [For_All, <for_all>,
Freq_Std, <std_freq> (or Freq, <cust_freq>)]

Where:

<source>: = {C1, C2, C3, C4, TA, TB, TC, TD}
<function>: = {TA, TB, TC, TD}
<slope>: = {POS, NEG}
<level>: = 1 to 99 if level is specified in percent (PCT), or level in
the units of the <source> waveform
<input_type>: = {CLK, DATA}
<for_all>: = {YES, NO}
<std_freq>: = {1.5M, 2M, 8M, 34M, 44M, 52M, 139M, 155M}
<cust_freq>: = 1 to 1e+9 HZ

Note:

Scaling of the *JitterTrack*[™] function Find Jitter Range can be
executed using the Find_Ctr_Range (FCR) command. Example.
TA:FCR scales *JitterTrack* on Trace A.

Command Examples

Clock Input Type:	TA:DEF EQN, "JITTERFREQ(C1)," SLOPE, POS, LEVEL, 0E-3, 0.5 DIV, TYPE, CLK
Data Input Type, with custom frequency:	TA:DEF EQN, "JITTERFREQ(C1)," SLOPE, POS, LEVEL, 0E- 0.5 DIV, TYPE, DATA, FOR_ALL, YES, FREQ, 1.125e+08 HZ
Data Input Type, with standard frequency:	TA:DEF EQN, "JITTERFREQ(C1)," SLOPE, POS, LEVEL, 0E-3, 0.5, TYPE, DATA, FOR_ALL, YES, FREQ_STD, 1.5M

Query/Response Examples TA:DEF? returns a string similar to the command examples



Jitter (Width-)

<function>: Def Eqn, "Jitterw (<source>)," Slope, <slope>, Level, <level>

Where:

<source>: = {C1, C2, C3, C4, TA, TB, TC, TD}

<function>: = {TA, TB, TC, TD}

<slope>: = {POS, NEG}

<level>: = 1 to 99 if level is specified in percent (PCT), or level in the units of the <source> waveform

Note:

Scaling of the *JitterTrack*TM function Find Jitter Range can be executed using the Find_Ctr_Range (FCR) command. Example. TA:FCR scales *JitterTrack* on Trace A.

Command Examples

Absolute level: TA:DEF EQN, "JITTERW(C2)," SLOPE, POS, LEVEL, 213.9E-3, 0.5

Relative level: TA:DEF EQN, "JITTERW (C2)," SLOPE, POS, LEVEL, 42 PCT, 0.5

Query/Response Examples TA:DEF? returns:

Absolute level: TA:DEF EQN, "JITTERW (C2)," SLOPE, POS, LEVEL, 213.9E-3 V, 0.5

Relative level: TA:DEF EQN, "JITTERW (C2)," SLOPE, POS, LEVEL, 42 PCT, 0.5

Appendix C: Remote Control Commands

Perhist<function> : DEFine EQN, "Perhist(<source>)," SI_Dir,<dir_value>, SI_Center,<center_value>, SI_Width, <width_value>

Where:

<source>: = {C1, C2, C3, C4, TA, TB, TC, TD}
<function>: = {TA, TB, TC, TD}
<dir_value>: = {V, H}
<center_value>: = -4.00 to 4.00 DIV
<width_value>: = 1 to 255 bins for horizontal slices, or
1 to 999 pixels for vertical slices.

Command Examples

Horizontal Slice: TA:DEF EQN, "PERHIST(C2)," SL_DIR, H, SL_CENTER, -0.62
DIV, SL_WIDTH, 34

Vertical Slice: TA:DEF EQN, "PERHIST(C2)," SL_DIR, V, SL_CENTER, -0.62
DIV, SL_WIDTH, 34

Query/Response Examples

TA:DEF? returns:

Horizontal Slice: TA:DEF EQN, "PERHIST(C2)," SL_DIR, H, SL_CENTER, -0.62
DIV, SL_WIDTH, 34

Vertical Slice: TA:DEF EQN, "PERHIST(C2)," SL_DIR, V, SL_CENTER, -0.62
DIV, SL_WIDTH, 34



Pertrace (Average) <function>: Def Eqn, "Pertrace(<source>)," Ptr_Type, AVG

Where:

<source>: = {C1, C2, C3, C4, TA, TB, TC, TD}

<function>: = {TA, TB, TC, TD}

<sigma_value>: = 0.5 to 10.0

<pop_value>: = 0.5 to 100.0 PCT

Command Example

TD:DEF EQN, "PERTRACE(C2)," PTR_TYPE, AVG

Query/Response Example

TA:DEF? returns:

TD:DEF EQN, "PERTRACE(C2)," PTR_TYPE, AVG

Appendix C: Remote Control Commands

Pertrace (Sigma)

<function>: DEF EQN, "PERTRACE(<source>)," PTR_TYPE,
SIGMA, IGMA_FACT,<sigma_value>

Where:

<source>: = {C1, C2, C3, C4, TA, TB, TC, TD}

<function>:= {TA, TB, TC, TD}

<sigma_value>: = 0.5 to 10.0

<pop_value>: = 0.5 to 100.0 PCT

Command Example

TD:DEF EQN,"PERTRACE(C2)",PTR_TYPE,SIGMA,SIGMA_FACT,1.5

Query/Response Example

TA:DEF? returns:

TD:DEF EQN,"PERTRACE(C2)",PTR_TYPE,SIGMA,SIGMA_FACT,1.5



Pertrace (Range)

<function>: Def Eqn, "Pertrace (<source>)." Ptr_Type, Range, Range_Pop, <pop_value>

Where:

<source>: = {C1, C2, C3, C4, TA, TB, TC, TD}

<function>: = {TA, TB, TC, TD}

<sigma_value>: = 0.5 to 10.0

<pop_value>: = 0.5 to 100.0 PCT

Command Example

TD: DEF EQN, "PERTRACE(C2)," PTR_TYPE, RANGE, Pertrace

Query/Response Example

TA:DEF? returns:

TD: DEF EQN, "PERTRACE(C2)," PTR_TYPE, RANGE, RANGE_POP, 99.5 PCT

§ § §