Bildik Tech Lab
OSIP Project

OSIP PSU4010A

SCPI Programming Reference

Visit the project website for more details
https://lab.ardabildik.com/osip-project/psu4010a/

SCPI Command List

1. *IDN? command

Returns manufacturer, module name, serial number, firmware and hardware revision.

Command:
“*IDN?”

Return:
<manufacturer>,
<module name>,
SN:<serial number>,
FW:<firmware version>,
HW:<hardware revision>
“BildikTechLab,PSU4010A,SN:DF6254209F56442D,FW:v0.0.1,HW:preRevA”

2. *RST command

Soft-power cycles the module to reset it.

Command:
“*RST”

Return:
“OK”

3. SYSTem:VERSion? command

Returns firmware and hardware revision of the module.

Command:
“SYSTem:VERSion?”

Return:
FW:<firmware version>,HW:<hardware revision>
“FW:v0.0.1,HW:preRevA”

4. SYSTem:ERRor? command

Returns the integer value of how many errors are present.

Command:
“SYSTem:ERRor?”

Return:
“<Error Count> Errors Available”
“2 Errors Available”

5. SYSTem:ERRor:GET

If there are any, return the next unread error message with how many unread errors are left.

Command:
“SYSTem:ERRor:GET”

Return:
“<Unread Error Count> More Errors Available”
“<Error Index>, <Error Data>”
“1 More Errors Available”
“2,ADC not detected”
or
“0 Errors Available”
“0 Errors Available”

6. SYSTem:ERRor:LIST

If there are any, return all of the available error messages with corresponding indexes.

Command:
“SYSTem:ERRor:LIST”

Return:

“<Error Count> Errors Available”	or	“0 Errors Available”
“<Error Index>, <Error Data>”		“0 Errors Available”
“2 Errors Available”		
“0,DAC not detected”		

7. SYSTem:ERRor:CLR

Clears all error messages in the memory.

Command:

“SYSTem:ERRor:CLR”

Return:
“OK”
“OK”

8. SYSTem:SCReen:BRIGhtness?

Returns the integer value of screen brightness from 0-100 if the screen is installed and dimmable. Returns “Unknown Command” otherwise.

Command:

“SYSTem:SCReen:BRIGhtness?”

Return:
“<Dimming Level(0-100)>” // “Unknown Command”
“35” // “Unknown Command”

9. SYSTem:SCReen:BRIGhtness [brightness]

Sets the screen brightness from 0-100 if the screen is installed and dimmable. Returns “Unknown Command” otherwise.

Argument: brightness: Receives an integer value between 0-100

Command:

“SYSTem:SCReen:BRIGhtness [brightness]”

Return:
“OK”
“OK”

10. SYSTem:SCReen:STATe [state]

It turns on or turns off the screen if the screen is installed and dimmable. Returns “Unknown Command” otherwise.

Argument: state: Receives “TRUE”, “ON” to enable screen
“FALSE”, “OFF” to disable screen

Command:

“SYSTem:SCReen:STATe [state]”

Return:
“OK”
“OK”

11. SYSTem:SCReen:STATe?

It returns “ON” or “OFF” based on the screen state if the screen is installed and dimmable. Returns “Unknown Command” otherwise.

Command:

“SYSTem:SCReen:STATe?”

Return:
“ON” or “OFF”
“ON”

12. SYSTem:SETTings:FASTusb

It switches the data output and command format to a simpler version for high speed operation.

Command:

“SYSTem:SETTings:FASTusb”

Return:
“OK”
“OK”

13. SYSTem:SETTings:FORMat [format]

It switches the data output mode to send continuous data per second or send a single line of data from a trigger input.

Argument: format: Receives, “CONT” for continuous data output , “TRIG” for triggered data output

Command:

“SYSTem:SETTings:FORMat [format]”

Return:
“OK”
“OK”

14. SYSTem:SETTings:FORMat?

It returns the current output data mode.

Command:

“SYSTem:SETTings:FORMat?”

Return:
“CONT” or “TRIG”
“TRIG”

15. SYSTem:SETTings:SPS [sample]

It sets the SPS (samples per second) for continuous data output mode.

Argument: sample: Receives an float number upto 250SPS

Command:

“SYSTem:SETTings:SPS [sample]”

Return:
“OK”
“OK”

16. SYSTem:SETTings:SPS?

It returns the current SPS setting of the continuous data output mode.

Command:

“SYSTem:SETTings:SPS?”

Return:
“<float upto 250>SPS”
“125SPS”

17. TRIGger:EDGe [sample]

It sets the SPS (samples per second) for continuous data output mode.

Argument: sample: Receives an float number upto 250SPS

Command:

“SYSTem:SETTings:SPS [sample]”

Return:
“OK”
“OK”

18. TRIGger:EDGe? [sample]

It sets the SPS (samples per second) for continuous data output mode.

Argument: sample: Receives an float number upto 250SPS

Command:

“SYSTem:SETTings:SPS [sample]”

Return:
“OK”
“OK”

19. TRIGger:SOURce [sample]

It sets the SPS (samples per second) for continuous data output mode.

Argument: *sample:* *Receives an float number upto 250SPS*

Command:

“SYSTem:SETTings:SPS [sample]”

Return:
“OK”
“OK”

20. TRIGger:SOURce? [sample]

It sets the SPS (samples per second) for continuous data output mode.

Argument: *sample:* *Receives an float number upto 250SPS*

Command:

“SYSTem:SETTings:SPS [sample]”

Return:
“OK”
“OK”

21. TRIGger:ENABle [sample]

It sets the SPS (samples per second) for continuous data output mode.

Argument: *sample:* *Receives an float number upto 250SPS*

Command:

“SYSTem:SETTings:SPS [sample]”

Return:
“OK”
“OK”

22. TRIGger:DISable [sample]

It sets the SPS (samples per second) for continuous data output mode.

Argument: *sample:* *Receives an float number upto 250SPS*

Command:

“SYSTem:SETTings:SPS [sample]”

Return:
“OK”
“OK”

23. TRIGger:STATus? [sample]

It sets the SPS (samples per second) for continuous data output mode.

Argument: *sample:* *Receives an float number upto 250SPS*

Command:

“SYSTem:SETTings:SPS [sample]”

Return:
“OK”
“OK”

24. STATus:TEMPerature?

Returns the current temperature of the integrated temperature sensor in float.

Command:

“STATus:TEMPerature?”

Return:
“<float temp>C”
“53.25C”

25. STATus:FANSpeed?

Returns the current speed of the cooling fans as integer between 0-100.

System drive value is used instead of the tachometer for compatibility with 2pin or 3pin fans.

Command:

“STATus:FANSpeed?”

Return:
“OK”
“OK”

26. STATus:OUTPut?

It returns the ON/OFF state of the output with current set voltage and set current values.

Command:

“STATus:OUTPut?”

Return:
“OFF,<Vset>V,<Iset>A” or “ON,<Vset>V,<Iset>A” or “ON,<Vset>V,<Iset>mA”
“OFF,14.00V,53.50mA”

27. OUTPut:ENABle [enable]

It turns ON/OFF the output of the PSU with current output settings.

Argument: enable: Receives “ON”,”TRUE” to enable the output
”OFF”,”FALSE” to disable the output

Command:

“OUTPut:ENABle [enable]”

Return:
“OK”

“OK”

28. OUTPut:ENABle?

It returns the ON/OFF state of the PSU output.

Command:

“OUTPut:ENABle?”

Return:

“<state>”

“ON”

29. OUTPut:VOLTage [voltage]

It sets the voltage output of the PSU output to the given voltage value.

.

Argument: voltage: Receives an float number between 0-40

Command:

“OUTPut:VOLTage [voltage]”

Return:

“OK”

“OK”

30. OUTPut:VOLTage?

It returns the set voltage output value of the PSU.

Command:

“OUTPut:VOLTage?”

Return:

“<voltage>V”

“12.12345V”

31. OUTPut:CURRent [current]

It sets the current output of the PSU output to the given current value.

.

Argument: current: Receives an float number between 0-10

Command:

“OUTPut:CURRent [current]”

Return:

“OK”

“OK”

32. OUTPut:CURRent?

It returns the set current output value of the PSU.

Command:

“OUTPut:CURRent?”

Return:

“<current>A” or “<current>mA”

“6.12345mA”

33. OUTPut:MODE?

It returns the control mode of the PSU output for CC,CV,OFF.

Command:

“OUTPut:MODE?”

Return:

“CC”, “CV”, “OFF

“CV”

34. OUTPut:RANGe:VOLTage [range]

It sets the voltage output range between 0-40V / 0-10V ranges.

Argument: range: “1”, “HI” to select 0-40V range
“0”, “LO” to select 0-10V range

Command:

“OUTPut:RANGe:VOLTage [range]”

Return:
“OK”
“OK”

35. OUTPut:RANGe:VOLTage?

It returns the voltage output range between 0-40V / 0-10V ranges.

Command:

“OUTPut:RANGe:VOLTage?”

Return:
“HI”, “LO”
“HI”

36. OUTPut:RANGe:CURRent [range]

It sets the voltage output range between 0-10A / 0-100mA ranges.

Argument: range: “1”, “HI” to select 0-10A range
“0”, “LO” to select 0-100mA range

Command:

“OUTPut:RANGe:CURRent [range]”

Return:
“OK”
“OK”

37. OUTPut:RANGe:CURRent?

It returns the voltage output range between 0-10A / 0-100mA ranges.

Command:

“OUTPut:RANGe:CURRent?”

Return:
“HI”, “LO”
“HI”

38. OUTPut:SETTing:RSENse [remote]

It sets the voltage sense location to remote sense or local sense.

Argument: range: “1”, “ON” to select remote sense
“0”, “OFF” to select local sense

Command:

“OUTPut:SETTing:RSENse [remote]”

Return:
“OK”
“OK”

39. OUTPut:SETTing:RSENse?

It returns the voltage sense location as remote sense or local sense.

Command:

“OUTPut:SETTing:RSENse?”

Return:
“RemoteSense” or “LocalSense"
“RemoteSense”

40. OUTPut:SETTing:RSET [sample]

It sets the SPS (samples per second) for continuous data output mode.

Argument: *sample:* *Receives an float number upto 250SPS*

Command:

“SYSTem:SETTings:SPS [sample]”

Return:
“OK”
“OK”

41. OUTPut:SETTing:RSET? [sample]

It sets the SPS (samples per second) for continuous data output mode.

Argument: *sample:* *Receives an float number upto 250SPS*

Command:

“SYSTem:SETTings:SPS [sample]”

Return:
“OK”
“OK”

42. OUTPut:SETTing:DPRogrammer [enable]

It enables/disables the down programmer function of the PSU.

Argument: enable: Receives “ON”,”TRUE” to enable down prog.
”OFF”,”FALSE” to disable down prog.

Command:

“OUTPut:SETTing:DPRogrammer [enable]”

Return:
“OK”

“OK”

43. OUTPut:SETTing:DPRogrammer?

It returns the down programmer state of the PSU.

Command:

“OUTPut:SETTing:DPRogrammer?”

Return:
“ON”, “OFF”
“ON”

44. OUTPut:SETTing:TCUToff [time]

It sets a timer in seconds that disables the output after the set time has passed after enabling the output.

Argument: time: float number starting from 0 in seconds.
“ON”, “TRUE” to enable the timer function.
“OFF”, “FALSE” to disable the timer function.

Command:

“OUTPut:SETTing:TCUToff [time]”

Return:
“OK”
“OK”

45. OUTPut:SETTing:TCUToff?

It returns the timer state with time setting in seconds.

Command:

“OUTPut:SETTing:TCUToff?”

Return:
“<state>,<timer>sec”
“OFF,12sec”

46. OUTPut:SETTing:OVP [limit]

It sets the output firmware based over voltage protection limit of the PSU output.

Argument: *sample:* *Receives an float number between 0-40*

Command:

“OUTPut:SETTing:OVP [limit]”

Return:
“OK”
“OK”

47. OUTPut:SETTing:OVP? [sample]

It sets the SPS (samples per second) for continuous data output mode.

Argument: *sample:* *Receives an float number upto 250SPS*

Command:

“SYSTem:SETTings:SPS [sample]”

Return:
“OK”
“OK”

48. OUTPut:SETTing:OCP [sample]

It sets the SPS (samples per second) for continuous data output mode.

Argument: *sample:* *Receives an float number upto 250SPS*

Command:

“SYSTem:SETTings:SPS [sample]”

Return:

“OK”
“OK”

49. OUTPut:SETTing:OCP? [sample]

It sets the SPS (samples per second) for continuous data output mode.

Argument: *sample:* *Receives an float number upto 250SPS*

Command:
“SYSTem:SETTings:SPS [sample]”

Return:
“OK”
“OK”

50. OUTPut:SETTing:OPP [sample]

It sets the SPS (samples per second) for continuous data output mode.

Argument: *sample:* *Receives an float number upto 250SPS*

Command:
“SYSTem:SETTings:SPS [sample]”

Return:
“OK”
“OK”

51. OUTPut:SETTing:OPP? [sample]

It sets the SPS (samples per second) for continuous data output mode.

Argument: *sample:* *Receives an float number upto 250SPS*

Command:
“SYSTem:SETTings:SPS [sample]”

Return:

“OK”
“OK”

52. MEASure:VOLTage?

Returns the measured output voltage of the PSU.

Command:
“MEASure:VOLTage?”

Return:
“<voltage>V”
“23.43564V”

53. MEASure:CURRent?

Returns the measured output current of the PSU.

Command:
“MEASure:CURRent?”

Return:
“<current>A” or “<current>mA”
“23.43564mA”

54. MEASure:POWer?

Returns the measured output power of the PSU.

Command:
“MEASure:POWer?”

Return:
“<power>W” or “<power>mW

“23.43564mW”

55. MEASure:TIMer?

It returns time since the output has been enabled. If it is disabled it returns the last time value from enable to disable.

Command:

“MEASure:TIMer?”

Return:
“<time>s”
“234.123s”

56. MEASure:CAPacity?

It returns the consumed energy from the output of the PSU since last enabling of the output. If it is disabled it returns the energy value between last enable and disable.

Command:

“MEASure:CAPacity?”

Return:
“<capacity>mWh”
“1248648.15235mWh”