

Philtec RS232/USB API
ASCII Command Line Interface
2021-03-09

General command information:

The DMS command line interface uses command strings with “label value” pairs to send and receive configuration and target data. All commands sent to the sensor are prefixed with a forward slash character ‘/’ (ASCII 47) and terminated with a line feed (ASCII 10) or carriage return (ASCII 13). Responses from the sensor are terminated with a line feed. **Commands and labels are case sensitive.**

Default terminal RS-232 configuration:

Format :8bits, no parity, 1 stop bit
Flow Control :none
Default speed :19.2Kbps

USB Virtual Com port configuration:

Format :8bits, no parity, 1 stop bit
Flow Control :none
Default speed muDMS 250Kbps
 microDMS 1Mbps

Note: FTDI USB driver is included in Windows 10. Installer can be downloaded from www.ftdichip.com if needed.

Quick Start using terminal program:

1. Attach power (+12VDC nominal, 7-24V) and RS-232/USB cable.
2. Use a terminal program to connect to the com port (see configuration above)
 Note: Any application that can communicate with the com ports should work.
3. Apply power to sensor. Sensor will send a boot message.
4. Send “/idn?” + LF (ASCII 10). Sensor will respond with model and serial information.
5. Send “/getConfig” + LF. Configuration settings will be displayed.
6. Send “/getTarget” + LF. Target and temperature data will be displayed.

Summary of top level commands:

/getTarget	Reads target distance, amplifier temperature, and sensor signals
/T	Shortened command string of /getTarget (same sensor response)
/getConfig	Reads all configuration data except calibrations.
/getCal	Reads calibration table data and descriptions.
/setConfig	Sets configuration data.
/setCal	Loads new calibration table into sensor memory.
/setFactoryConfig	Sets serial number and optical cable type (RC/D). Note: Changes to serial and cable type are not recommended (Factory only)
/idn?	Reads sensor model and serial number
/stop	Stops target stream data
/reboot	Reboots sensor firmware

Summary of label value pairs:

Read and write (getConfig, setConfig)

avg	averaging rate of internal ADC
avgDef	default average. avg set to this value after power up or reboot
bps	bps setting of RS-232 port (USB virtual com port bps is not adjustable)
calTable	calibration table for distance lookup
Dpeak	peak signal level. Must be set for each target surface before measurements
gain	light source intensity adjustment
posCode	position code. Used for multiple sensors on single com port
setTemp	temperature set point. Heater is on if temp<setTemp
sign	signature string. 24 character max user defined string for sensor
Tformat	Target format. Used to define getTarget response
TformatDef	default target format. Tformat set to this value after power up or reboot
uom	unit of measure

2DMS model additions

calTable2	calibration table for distance lookup channel 2
Dpeak2	peak signal level channel 2
gain2	light source intensity channel 2
sign2	signature string channel 2
TchSelect	channel selection for multichannel sensors

Read Only (getConfig)

bpsRange	list of available bps values
calTableMax	maximum calibration slot number available
cmdLenMax	maximum command line length including line feed
fwVer	firmware version
HWcode	hardware code of sensor
modelCode	model code of DMS sensor
RCDcode	RC or D optics flag (R=RC, D=D)
sampleClkPer	sample clock period, internal ADC period
serial	serial number of DMS sensor
snrMax	maximum snr value from getTarget command (snrMax = 100% receive power)

2DMS model additions

chCnt	channel count (2DMS sensor = 2, otherwise = 1)
-------	--

Read Only (getTarget or T)

distN Distance from calibration table (D near side or RC)
distF Distance from calibration table (D far side)
signal Raw signal from sensor. Used with calibration table to determine distance
snr 8bit value (0-255) representing receive power
snrp percentage of receive power relative to calibration target.
temp current amplifier temperature
TpckCnt binary packet target read count. Sent at start of "stream bin" option reads.

2DMS model additions

distN2 channel 2 distance from calibration table (D near side or RC)
distF2 channel 2 distance from calibration table (D far side)
signal2 channel 2 raw signal from sensor
snr2 channel 2 8bit value (0-255) representing receive power
snrp2 channel 2 percentage of receive power relative to calibration target.

Label Value pair definitions:

Read and Write (getConfig, setConfig)

avg

default: avgDef value

range: 1-12

Description:

Multiple internal readings are used to create target data. The value of 'avg' determines the amount of averaging with higher values decreasing speed but increasing resolution. Total reads to be averaged are 2^{avg} (see table).

Total sample period = $2^{\text{avg}} * \text{sampleClkPer}$

avg setting	Number of Internal readings
12	4096
11	2048
10	1024
9	512
8	256
7	128
6	64
5	32
4	16
3	8
2	4
1	2

avgDef

default: 12

range: 1-12

Description:

avgDef will be loaded into avg at power up or reboot.

calTable

default:1

range:1-calTableMax

Description:

The DMS sensor has 24 calibration table slots for different target surfaces. The standard factory calibrations are a mirror (specular) surface in calTable 1 and a dull (diffuse) surface in calTable 2. Most targets will use table 2 but additional calibrations can be added with the /setCal command.

Note: future sensor firmware versions (>3.102) will use "calTable" for both getConfig and setConfig for consistency.

setTemp

default:35

range:0-60

Description:

The sensor has a low power heater that can be used to stabilize the temperature of the amplifier.

Maintaining a constant temperature will reduce sensor drift. The heater can only increase the amplifier temperature a few degrees above ambient conditions therefore the temperature may not reach the set point in low ambient temperatures. Setting the temperature below ambient will turn off the heater.

gain

default:25

range:0-100

Description:

The sensor's light source intensity is controlled by the "gain" setting. A setting of 100 is full power (used for most targets). Reflective targets may require a reduction in gain to avoid saturating the optical input. Optical input can be monitored with the snr value (see below). A gain setting of 0 will turn off the sensor's LED.

uom

default:micron

range: micron, um, mm, nm, ml (note:um=micron, ml=mill-Inch)

Description:

Distance values will be sent in the selected unit of measure. Micron can be selected by either the full text "micron" or the abbreviation "um". Configuration output labels will always use "um".

Dpeak (note: used for D type sensor only, ignored by RC sensors)

Default:1.0 (note: MUST be reset on target surface being measured)

Range: 0.001 – 7.9999

Description:

D type sensors are reflectance dependent. For each target surface the optical peak must be set before the sensor can correctly interpret the calibration tables.

Setting Dpeak:

Option 1: Adjust distance between target and sensor tip while receiving target data (see getTarget command). Target distance should be adjusted until signal data "signal" reaches maximum value. Send "/setConfig Dpeak" + LF to sensor and this signal value will be saved to Dpeak.

Option 2: If the maximum signal level is known it can be sent directly. Send "/setConfig Dpeak x.xxx" + LF where x.xxx is the maximum signal level for the target in use.

posCode

Default:0

Range:0-63

Description:

Position Code (posCode) can be used to communicate with a sensor when multiple sensors are connected to the same serial port. Sending "pc xx" before a command (xx = position code of sensor) will send that command only to the addressed sensor. When each sensor has its own serial port (or USB), position code is not used.

sign

Default:"" (blank)

Range:string of up to 24 characters

Description:

Each sensor can save a description string which is called the signature. This string is not used by the sensor but can be useful for describing the sensor's purpose.

bps

Default: 19200

Range: bpsA value

Description:

The sensor's RS-232 communication bps is controlled by the "bps" value. Sending a value not in the range list will leave the bps setting at its current value. When a new value is sent with the setConfig command, the sensor will confirm the change using the current bps speed and then switch to the new setting for all further communications. Note: the sensor always starts at 19200 at power up and then switches to the final setting. If you are unsure of the sensor's bps setting, connect using bps 19200 and the sensor will display the final bps setting before switching.

Note: bps value for USB virtual com port is not adjustable

Tformat

TformatDef

Default:127

Range:0-127

Description:

Tformat controls which information is sent for the getTarget command. This value is set to TformatDef (Tformat default) after power up or reboot. Changes to TformatDef are saved permanently while Tformat changes are lost at power down or reboot. Tformat is a set of bit flags with each bit controlling which data is sent.

Bit position	Label	getTarget Data to send
0 (value=2 ⁰ =1)		1=labels sent, 0=labels not sent
1 (value=2 ¹ =2)	temp	current temperature of amplifier, heater will try to reach setTemp value
2 (value=2 ² =4)	signal	signal power received with 255=maximum (signal clipping at snrMax)
3 (value=2 ³ =8)	snr	signal, raw non-linear sensor response
4 (value=2 ⁴ =16)	distn	distance near side(D type) or distance (RC type)
5 (value=2 ⁵ =32)	distf	Distance far side (D model), not used for RC type sensor
6 (value=2 ⁶ =64)	snrp	Snr percentage of reflectance

Example: For a getTarget response that includes labels, temperature, and distance (far side) the resulting Tformat value would be labels(1) + temperature(2) + distance far side(32) =35

Other labels and values will be omitted.

Read Only (getConfig)

avgMax

Description:

Maximum available avg value. Most sensors have an avgMax of 12 ($2^{12} = 4096$ readings, see avg key)

bpsRange

Description:

List of available bps values. List value is quote encapsulated.

calTableMax

Description:

Maximum calibration slot available in sensor (slots from 1 to calTableMax value).

Note: Most sensors have 24 calibration slots but some models (2DMS or custom) can modify this value.

chCnt

Description:

Channel count is the number of channels available. For 2DMS models this value is 2. All other DMS sensors have a chCnt of 1.

cmdLenMax

Description:

Maximum command line length including line feed character. Sensor has a circular buffer which will discard characters once this line length has been reached.

HWcode

Description: Model code of DMS hardware.

DMS	full size DMS
DMSv2	full size DMS version 2
mDMS	mini DMS (4 position gain system)
mDMSv2	mini DMS (single gain system)
muDMS	muDMS (USB)
microRS	microDMS with RS-232
microUSB	microDMS with USB

sampleClkPer

Description:

The ADC (analog to digital converter) sample period. The full sample period can be calculated with the avg value. Full sample period = $(2^{\text{avg}}) * \text{sampleClkPer}$

serial

Description:

Serial number of DMS sensor.

snrMax

Description:

The maximum snr value before signal clipping or distortion results. If the snr value approaches snrMax, the gain value should be reduced.

Read Only (getTarget or T)

distn

Description:

Distance from sensor tip to target surface calculated with currently selected table (calTable value). For D type sensors the near side (positive slope) of the calibration table will be used. RC type sensors only have a near side.

distf

Description:

Distance from sensor tip to target surface calculated with currently selected table (calTable value). Far side (negative slope) of the calibration curve will be used. This value only available for D type sensors.

signal

Range: 0-7.99999

Description:

Signal is the raw, non-linear data from the sensor. This value is used with the calibration table to calculate distance. For D type sensors, this value is used to set the optical peak for the current target surface.

snr

Range:0-255

Description:

snr is a measurement of the amount of light entering the sensor. This is a byte value with snrMax representing maximum input power before clipping or distortion. Ideal values for snr are between 5 and snrMax. snr can be adjusted by changing the "gain" setting with a higher/lower gain (more/less power emitted) resulting in higher/lower snr.

snrp

Description:

RC type: Percentage of reflectance of target surface compared to calibration surface.

Example 100%=current target reflectance equal to calibration target. 50%=current target half as reflective. 200%=current target twice as reflective.

D type: Percentage of reflectance of target surface compared to optical peak. This value will change with distance and reflectance. To compare surface reflectance the distance must be kept static.

temp

range:0-60

Description:

Current temperature of the amplifier. If value is lower than setTemp the amplifier heater will be turned on.

TpckCnt

range:1-256

Description:

Number of reads per binary stream packet. Always 1 for ascii stream data.

Command Definitions:

Command syntax notes:

Optional parameters surrounded with square brackets [].

Parameters requiring user selected data are surrounded with angle brackets <>.

Optional parameters with user selected data are surrounded by both [<>].

Sensor responses depend on variable data (distance, temperature).

Values are substituted with value codes in format description as follows:

String: sss

Integer: ##

Float: #.### (note: sensor uses . (ASCII 46) for decimal mark)

/getConfig

Description: returns all configuration data in "label value" pairs.

Response format:

```
getConfig avg ## calTable ## uom ss setTemp ## gain ### Dpeak #.### TformatDef ### Tformat ###  
fwVer #.### serial ##### modelCode ss sign "sss" bps #####
```

/getTarget [stream {bin | ascii}]

/T [stream]

Note: /T and /getTarget are the same command.

Description: returns current distance and temperature data. The "label value" data sent by getTarget is determined by the "Tformat" value.

Example:

To Sensor: /getTarget

Response: T signal x.xxxx snr xxx temp xx.x distn xxx.xx distf xxx.xx snrp x.xxx

stream ascii option:

If command argument "stream ascii" is added, the sensor will continuously send target data until the "/stop" command is received. "stream ascii" will send data in ascii format with each target read terminated with a line feed.

Example:

To Sensor: /getTarget stream ascii

Response: T stream ascii TpckCnt 1 signal x.xxxx snr xxx temp xx.x distn xxx.xx distf xxx.xx snrp x.xxx <LF>

Response: T signal x.xxxx snr xxx temp xx.x distn xxx.xx distf xxx.xx snrp x.xxx <LF>

Note: second response line is repeated until /stop command is sent to sensor.

stream bin option:

If command argument "stream bin" is added, the sensor will send binary packets of distance data until /stop command is sent to the sensor. "label" data is never sent and the "label" bit of Tformat is ignored.

Example:

To Sensor: /getTarget stream bin

Response: T stream bin TpckCnt ### signal x.xxxx snr xxx temp xx.x distn xxx.xx distf xxx.xx snrp x.xxx <LF>

Response: <binary packet>

Note: second response line is repeated until /stop command is sent to sensor.

See "Binary Data formats" section for packet description.

/setConfig [<label>] [<value>]

Description: sets configuration data (except serial and model type).

setConfig uses "label value" pairs to set configuration. Command line arguments are separated by a space (ASCII 32). Multiple "label value" pairs can be used on a single line but total line length cannot exceed 250 characters. Each configuration change will be confirmed in the response line.

Example:

Change calibration table to 3:

To Sensor: /setConfig cal 3

Response: setConfig cal 3

Set Dpeak to the current signal level:

Note: Dpeak is the only label that does not require a value. If no value is given, the current sensor signal will be used.

To Sensor: /setConfig Dpeak

Response: setConfig Dpeak x.xxx

Set average value to 12 (slowest speed but highest resolution)

To Sensor: /setConfig avg 12

Response: setConfig avg 12

/getCal [<calibration slot>] [calFmt {binTable | asciiTable | descr}]

Description: returns calibration table data.

Command arguments, if any, determine which data is returned and in what point table format. By default, the currently selected calibration point table (calTable) is sent in ASCII format. The "calFmt descr" argument will limit the response to only the calTable value, description, gain, and number of points (point data omitted). See "Binary Data Formats" below for details of the "calFmt binTable" format.

Example:

Send the description for the currently selected calibration (calTable):

To Sensor: /getCal descr

Response: getCal calTable xx descr "sss" gain xxx points xxx

Send the description for all available calibrations:

To Sensor: /getCal descr all

Response: getCal calTable xx descr "sss" gain xxx points xxx

Above line repeated for all available calibrations

Response after last calibration: getCal end

Without the "descr" argument the getCal command also sends the calibrations points (distance, signal, snr).

Example:

Send all data for currently selected calibration table:

To Sensor: /getCal

Response: getCal calTable xx descr "sss" gain xxx points xxx "<distance> <signal> <snr>"

Note: distance, signal, and snr will be repeated for each point of the calibration.

The entire group of calibration points are quote encapsulated.

Send all data for all available calibrations:

To Sensor: /getCal all

Response: getCal calTable xx descr "sss" gain xxx points xxx "<distance> <signal> <snr>"

Note: data repeated for all available calibrations

Response after last calibration: getCal end

The gain displayed is the gain that was used for the calibration, not the currently selected gain.
The unit of measure for distance is the currently selected uom (may be different than the uom used during calibration)

/setCal calTable ## gain ### uom <uom> descry "sss" [channel #]

Description: create new calibration table with 255 points maximum.
Please contact factory for software to add new calibrations.

Binary Data Formats

Data I/O is performed in ASCII format whenever possible for ease of use and debugging. There are currently 2 situations where the sensor can switch to binary mode for higher speed communication:

- data stream (/T stream bin)
- calibration table read (/getCal calFmt binTable)

Data Stream Binary Format:

Binary format is required to reach the sensor's maximum speed (5K s/s for all except microDMS, 16K s/s for microDMS). Binary mode is selected using the "stream bin" flag for the target command.

Example for high speed streaming:

Set sensor avg to highest speed (avg 1), and sending only temperature, signal, and snr data.

To Sensor: /setConfig avg 1 Tformat 14

Response: setConfig avg 1 Tformat 14

Start binary stream.

To Sensor: /T stream bin

Response: T stream bin TpckCnt 256

Target data will now be sent continuously using data packets. TpckCnt shows the number of individual target reads contained within each packet.

Stream Format:

Header (1 byte, dec 170, binary 10101010)

Packet Size (2 bytes, MSB,LSB)

Packet (bytes = packet size value above)

Checksum (2 bytes, simple checksum with all bytes of packet added together)

Note: do not include header and packet size bytes in checksum

The Tformat value is used to decode the packet (see getFormat and setFormat command). Selecting only the data that your application requires will increase the maximum attainable speed.

Packet Format:

repeated <TpckCnt> number of times in packet

signal (3 bytes, non-linear sensor response used for calibration table lookup)

snr (1 byte, signal power of returning light)

if (Tformat & 2⁴) distance, near side table (4 bytes, IEEE single)

if (Tformat & 2⁵) distance, far side table (4 bytes, IEEE single)

if (Tformat & 2⁶) reflectance, percentage compared to cal target (4 bytes, IEEE single)

Amplifier temperature (2 bytes, bit=0.0078125 C)
Status (1 byte, bit 0 set if reading was skipped, all others for future use)
Multiply the 3 byte raw signal value by 9.53674e-07 to convert back to floating point.
Signal and snr are always sent in binary stream mode.
Label (Tformat bit 0) is ignored.

Note: "Skipped readings" status flag will result if the RS-232 bps setting cannot support the average rate selected. Increase bps to a higher setting (/setConfig bps ###) to increase maximum data rate. Available bps settings are given by the bpsRange value (displayed with /getConfig). USB connected sensors always communicate at their maximum bps and, therefore, will not set the "Skipped readings" flag.

Tformat settings and data streaming speed:

The DMS amplifiers use an 8 bit processor which can be slow at floating point computations. To achieve maximum speed, most drivers and applications will request the calibration tables from the sensor (/getCal) and perform the signal to distance conversion on the PC. This is typically only needed for the highest speed settings.

Note for DMS-D type sensors: A Tformat value which calls for both the near (bit 4) and far (bit 5) side calibration lookup procedures will be very slow. This requires the sensor to load the near side table from flash to ram, perform the lookup, then repeat the process for the far side data -- a slow process. Note that this processor can only hold one calibration table in ram at a time so every reading needs these steps. Typically, you only want "near side" or "far side" (not both) and this allows the sensor to load the table once and reuse the data on further reads.

Calibration Table binary format

Many PC drivers will read all calibration data from the sensor during initialization. This can take several seconds and can become an issue if a large group of sensors are being initialized. To increase the speed of reading the calibrations, the table data can be sent in a quote encapsulated binary format. The decrease in loading time is especially noticeable in RS-232 sensors set at low BPS.

If the calFmt flag has been set to "binTable" (see /getCal command), the sensor will send table data labeled as "pointsBin" in binary format as follows:

- Distance data (4 bytes, IEEE single)
- Signal data (4 bytes, IEEE single)
- SNR data (1 byte)

Note that the byte values 10 (Line Feed), 34 (quote), and 92 ('\ backslash) are escaped to avoid conflict with the quote encapsulation. If byte 92 ('\') is found, the next byte must be read to determine the original data.

"\L" → 10 (Line Feed)

"\Q" → 34 (quote)

"\\" → 92 (backslash escapes itself)