

# **Pundit Lab/Lab+ Remote Control Interface**

## **Documentation**

Document Revision: 5

Revision Date: 03-June-2014

## Revision History

Rev	Date	Comments
1	02-March-2010	pre-released for review
2	12-April-2010	Detailed description of serial interface related issues, data structures and individual commands
3	20-July-2011	Fixed some inconsistencies in data structures and remote command descriptions. Additional probe frequency of 250kHz (>V1.2.4) Pundit Lab+: Data structure for conversion curves added (TConvCurve) Measurement and device setup data adapted accordingly. New probe gain levels for Pundit Lab+
4	24-October-2011	SET_DEVICE_SETUP: Description of command, response and example corrected and clarified (two responses, no CRC)
5	03-June-2014	Update of conversion curve structure, which has changed for firmware version $\geq$ V2.3.0.

## Table of Contents

<b>Pundit Lab/Lab+ Remote Control Interface .....</b>	<b>1</b>
<b>Documentation .....</b>	<b>1</b>
<b>1. Introduction.....</b>	<b>3</b>
1.1 Scope / intended use.....	3
1.2 Boundary conditions.....	3
1.3 Abbreviations.....	3
<b>2. Overview .....</b>	<b>4</b>
2.1 Purpose .....	4
2.2 Notation .....	4
2.3 Requirements, prerequisites, limitations .....	4
2.4 Suggestions, bugs, improvements .....	4
<b>3. Serial interface.....</b>	<b>5</b>
3.1 Virtual COM port driver.....	5
3.2 Serial Interface parameters .....	5
3.3 Detect Pundit Lab/Lab+ device .....	5
<b>4. Data structures .....</b>	<b>6</b>
4.1 Data types .....	6
4.2 Conversion curves data (TConvCurve).....	6
4.3 Measurement data .....	7
4.4 Device setup data.....	9
4.5 Curve data .....	11
<b>5. Remote commands description.....</b>	<b>12</b>
5.1 Overview.....	12
5.2 Individual remote commands .....	12

## 1. Introduction

### 1.1 Scope / intended use

This document describes the Pundit Lab/Lab+ remote control interface. It consists of a set of binary commands which allow the user to configure the Pundit Lab/Lab+ device, to trigger measurements and to collect measurement data using a remote PC.

The target audience is software engineers and Proceq customers who want to embed Pundit Lab/Lab+ remote functionality into their own PC applications.

### 1.2 Boundary conditions

This document relies on the Pundit Lab/Lab+ Firmware Version V2.4.0 but is also valid for older versions. For further conditions see [2.3 Requirements, prerequisites].

### 1.3 Abbreviations

Term	Description
USB	Universal serial bus
UART	Universal Asynchronous Receiver Transmitter
PL	Pundit Lab
PLP	Pundit Lab+

## 2. Overview

### 2.1 Purpose

The purpose of this document is to describe all necessary interface facts to enable customers to use the Pundit Lab/Lab+ device in a most flexible way according to their needs. Possible applications are:

- Integration of Pundit Lab/Lab+ into a (semi-) automated production line.
- Controlling Pundit Lab/Lab+ and collecting measurement data from the device using own software and databases to be further processed or stored without the need of the Pundit Link software and the overhead of unneeded manual work.
- Remote setup of Pundit Lab/Lab+ into a specific, well defined measurement mode with one click.

### 2.2 Notation

The different functions can be started through remote commands described later in this document. The description of the individual commands is based on the following rules:

Each command consists of a defined number of bytes. These bytes are always described in hexadecimal syntax (e.g. 0x12 = 18 decimal).

Variable byte-values within a command are described by replacements characters (e.g.: PP).

For number values with two bytes the low-order byte is presented by lower-case letters and the high-order byte by upper-case letters (e.g.: pp PP).

For number values with more than two bytes a number is added to the replacement characters. Numbering starts with 0, which represents the low-order byte (e.g.: ZZ0 ZZ1 ZZ2).

### 2.3 Requirements, prerequisites, limitations

To avoid problems during remote connection please take care of the following:

- Make sure the Pundit Lab/Lab+ is powered and connected to the PC
- Do not disconnect the Pundit Lab/Lab+ while communication with the PC is active
- Do not try to start/stop measurements locally by pressing any Pundit Lab/Lab+ button while measurements are being triggered remotely from a PC.

### 2.4 Suggestions, bugs, improvements

We hope that with this solution we can fulfill most customer needs regarding the matter of automation and remote control of Pundit Lab/Lab+.

Should there be any bugs, missing commands, questions and other suggestions regarding this add-on, please feel free to write an email to: [software@proceq.com](mailto:software@proceq.com)

We appreciate your feedback and we will gather any incoming information and try to help whenever possible.

## 3. Serial interface

### 3.1 Virtual COM port driver

The Pundit Lab/Lab+ offers a USB UART interface by which it can be remotely controlled from a PC. I.e. physically, the device appears like a normal USB device, but internally the Pundit Lab/Lab+ uses a UART, which is an asynchronous, serial interface (similar to RS232). A so called USB bridge device on the Pundit Lab/Lab+ makes the conversion from UART to USB signals and vice versa.

On the PC side it is necessary to install a virtual COM port driver, which causes the Pundit Lab/Lab+ to appear as if it was connected to a standard serial COM port to the PC. This allows application software running on the PC to access the Pundit Lab/Lab+ in the same way as it would access a standard COM port.

To install the virtual COM port driver you can either:

- Get and install Proceq's Pundit Link software. The virtual COM port driver is installed automatically.

or:

- Get the virtual COM port driver setup program from FTDI's home page (<http://www.ftdichip.com/Drivers/VCP.htm>)

Provided that the PC has an active internet connection, some newer versions of the Windows operating system (XP, Vista and 7) automatically get and install the needed driver when the USB device is connected.

### 3.2 Serial Interface parameters

To be able to communicate to the Pundit Lab/Lab+ device, the following serial interface parameters must be set:

- 115200 baud
- 8 data bit
- 1 stop bit
- No parity

### 3.3 Detect Pundit Lab/Lab+ device

When the Pundit Lab/Lab+ device is connected to a PC, the Windows operating system automatically assigns the next available COM port number. To find the correct COM port from within an application program, it is best to scan the COM ports starting at COM1, COM2, and so on. By sending the GET\_DEVICE\_INFO command (with sub-command 0x00 for device name and 0x04 for device signature) it's possible to see if the Pundit Lab/Lab+ is connected to that particular port.

Once connected, all of the remote commands described in this document can be sent to the Pundit Lab/Lab+.

## 4. Data structures

### 4.1 Data types

The following table lists the types used to characterize data exchanged between the Pundit Lab/Lab+ and a remote PC.

Type specifier	Description
INT8U	Unsigned integral type with a size of 1 byte
INT8S	Signed integral type with a size of 1 byte
INT16U	Unsigned integral type with a size of 2 bytes
INT16S	Signed integral type with a size of 2 bytes
INT32U	Unsigned integral type with a size of 4 bytes
INT32S	Signed integral type with a size of 4 bytes
INT64U	Unsigned integral type with a size of 8 bytes
INT64S	Signed integral type with a size of 8 bytes

### 4.2 Conversion curves data (TConvCurve)

Conversion curves for compressive strength calculation are only available for Pundit Lab+. The following table list the data bytes of such a conversion curve structure. It will be used in the measurement and device setup data structures described further down in this document.

Byte-Nr	Structure Element	Description	Units / Range
1	INT8U version	Structure version	0x10 (< V2.3.0) 0x21 (≥ V2.3.0)
2	INT8S curveType	Conversion curve type -1 undefined 0 polynomial ( $a * v^3 + b * v^2 + c * v + d$ ) 1 exponential ( $a * e^{b*v}$ ) 2 SONREB ( $a * v^b * R^c$ )  $v$ : Pulse velocity	
3...34	INT64S coeff[4]	Array of 4 coefficients ( $a, b, c, d$ in the above equations), each of type INT64S.  <V2.3.0: The effective float value of the coefficients results by converting each INT64S array element to a float and dividing it by $10^{-12}$ .  ≥ V2.3.0: Each INT64S array element represents a 32-bit float value according to IEEE 754.	
35..36	INT16S min	Lower limit of pulse velocity ( $v$ )	[m/s]
37..38	INT16S max	Upper limit of pulse velocity ( $v$ )	[m/s]
37..49	INT8S name[11]	Array of 11 characters specifying the name of the conversion curve. The 11 <sup>th</sup> character is the string (null-) terminator.	

### 4.3 Measurement data

The following table lists the sequential data bytes sent by Pundit Lab/Lab+ when a remote measurement is triggered. They are described in the order they are received on the remote PC. Extensions valid only for Pundit Lab+ are gray shaded in the table and marked with PLP.

Structure size: Pundit Lab: 50 Bytes, Pundit Lab+: 109 Bytes

Byte-Nr	Structure Element	Description	Units / Range
1	INT8U version	Structure version	0x10 ( $\leq$ V1.2.5) 0x20 ( $\geq$ V2.0.4)
2	INT8U measType	Measurement type 0 undefined 1 direct measurement (default type in case of a remote controlled measurement) 2 surface measurement 3 crack measurement	
3...10	INT64U Reserved1	Reserved, always 0	
11...14	INT32U measId	Measurement object id	0...0xFFFFFFFF
15...16	INT16U corrFactor	Correction factor (temperature, moisture, etc...)	1/100 70...130 (0.7...1.3)
17...18	INT16U pulseLength	Pulse length of transmitter trigger impulse	1/10 [ $\mu$ s] 1...1000 (0.1...100.0 $\mu$ s)
19	INT8S pulseAmpl	Amplitude of transmitter trigger pulse -1 Undefined 0 125 V 1 250 V 2 350 V 3 500 V 4 AUTO (automatic amplitude setting)	[V]
20	INT8S probeFreq	Resonance frequency of connected probes -1 Undefined 0 24 kHz 1 37 kHz 2 54 kHz 3 82 kHz 4 150 kHz 5 200 kHz 6 220 kHz 7 500 kHz ( $\leq$ V1.2.4) 7 250 kHz ( $>$ V1.2.4) 8 500 kHz ( $>$ V1.2.4)	[kHz]
21...24	INT32U measDistance	Distance between measurement probes (i.e. path length). If propSpeed is given this value will be calculated as result of the measurement.	1/100 [mm] 0...999999 (0...9.99999m)
25...28	INT32U crackDepth	Crack depth	[mm] 0...9999 (0...9.999m)
29...32	INT32U propTime1	Propagation time 1 as the result of a measurement	1/100 [ $\mu$ s] 0...999999 (0...9999.99 $\mu$ s)
33...36	INT32U propTime2	Propagation time 2 as result of a measurement. For direct measurement this time is always 0. For surface velocity and crack measurement the value will be different from 0.	1/100 [ $\mu$ s] 0...999999 (0...9999.99 $\mu$ s)
37...40	INT32U propSpeed	Propagation speed (pulse velocity) If measDistance is given this value will be calculated as result of the measurement	1/100 [m/s] 0...1000000 (0...10000.00m/s)

Byte-Nr	Structure Element	Description	Units / Range
41	INT8S rxProbeGain	Receiver probe gain -1 Undefined 0 1 1 10 2 100 3 AUTO (automatic gain setting)  Pundit Lab+: -1 Undefined 5 1 6 2 7 5 8 10 9 20 10 50 11 100 12 200 13 500 14 1000 15 AUTO (automatic gain setting)	
42	INT8U result	Type of calculated result 1 calculated result is measDistance 2 calculated result is propSpeed	
43...44	INT16S calibTimeOfs	Calibration time offset	
45...46	INT16U pulseAmplValue	Integral value of pulse amplitude (useful if pulseAmpl = AUTO)	125...500
47...48	INT16U rxProbeGainValue	Integral value of probe gain (useful if rxProbeGain = AUTO)	PL: 1...100 PLP: 1...1000
49...50	INT16U nrOfCurveSamples	Number of curve samples sent following this measurement data structure. Always 0 for measurements stored on Pundit Lab/Lab+. Possibly ≠ 0 for remotely triggered measurements (see command TRIGGER_MEASUREMENT).	0
51	INT8U reserved	PLP: Undefined	
52...53	INT16S ambientTemp	PLP: Ambient temperature (from temperature sensor of internal real-time-clock device)	1/10 [°C] -1280...+1278 (-128.0...+127.8°C)
54...57	INT32U compStrength	PLP: Compressive strength	1/10 [MPa]
58...106	TConvCurve curve	PLP: Currently active conversion curve data structure according to TConvCurve.	
107...108	INT16U rebValue	PLP: Currently active rebound value corresponding to the currently active conversion curve. The value is only valid if the conversion curve is of type SONREB.	1/10 0...1000 (0...100.0)



## 4.4 Device setup data

The following table lists the sequential data bytes sent by Pundit Lab/Lab+ when it's setup is interrogated remotely. They are described in the order they are received on the remote PC.

Extensions valid only for Pundit Lab+ are gray shaded in the table and marked with PLP.

Structure size: Pundit Lab: 59 Bytes, Pundit Lab+: 322 Bytes

Byte-Nr	Structure Element	Description / Range	Units / Range	Note
1	INT8U version	Structure version	0x10 ( $\leq$ V1.2.5) 0x20 ( $\geq$ V2.0.4)	4)
2	INT8U reserved	always 0	0	3)
3...6	INT32U measId	Current measurement object id. Incremented by Pundit Lab/Lab+ (see command TRIGGER_MEASUREMENT)		4)
7...10	INT32U nrOfStoredMeas	Number of measurements stored on Pundit Lab/Lab+		4)
11...14	INT32U reserved	Always 0	0	3)
15...18	INT32U presetMeasDistance	Preset measurement distance between probes for direct measurement	1/100 [mm]	1)
19...22	INT32U presetCrackDistance	Preset measurement distance between probes for crack measurement	1/100 [mm]	1)
23...26	INT32U presetSurfaceDistance	Preset measurement distance between probes for surface velocity measurement	1/100 [mm]	1)
27...28	INT16U corrFactor	Correction factor (temperature, moisture, etc.)	1/100 70...130 (0.7...1.3)	
29...32	INT32U calibTime	Target calibration time. Should be same as on the calibration calibration stick.	1/100 [ $\mu$ s]	1)
33...34	INT16S calibTimeOfs	Calibration time offset. Calculated during calibration, which is not possible to be done remotely.	1/100 [ $\mu$ s]	1)
35...36	INT16U pulseLength	Pulse length of transmitter trigger pulse	1/10 [ $\mu$ s]	
37...40	INT32U reserved	Always 0	0	3)
41	INT8U lenUnit	Length unit to be displayed on device 0 m 1 ft		2)
42	INT8S intRxProbeGain	Receiver probe gain -1 Undefined 0 1 1 10 2 100 3 AUTO (automatic gain setting)  Pundit Lab+: -1 Undefined 5 1 6 2 7 5 8 10 9 20 10 50 11 100 12 200 13 500 14 1000 15 AUTO (automatic gain setting)		
43	INT8U reserved	always 0		3)

Byte-Nr	Structure Element	Description / Range	Units / Range	Note
44	INT8S pulseAmpl	Amplitude of transmitter trigger pulse -1 Undefined 0 125 V 1 250 V 2 350 V 3 500 V 4 AUTO (automatic ampl. setting)	[V]	
45	INT8S probeFreq	Resonance frequency of connected probes -1 Undefined 0 24 kHz 1 37 kHz 2 54 kHz 3 82 kHz 4 150 kHz 5 200 kHz 6 220 kHz 7 500 kHz ( $\leq V1.2.4$ ) 7 250 kHz ( $> V1.2.4$ ) 8 500 kHz ( $> V1.2.4$ )	[kHz]	
46	INT8S measMode	Measurement mode -1 Undefined 0 Continuous Mode 1 Burst Mode		
47...50	INT32U measDistance	Distance between measurement probes If setup data is set by means of the appropriate remote command and if propSpeed is given, this value must be set to 0 and will then be calculated as result of the measurement.	1/100 [mm] 0...999999 (0...9.99999m)	
51...54	INT32U propSpeed	Propagation speed. If setup data is set by means of the appropriate remote command and if measDistance is given this value must be set to 0 and will be calculated as result of the measurement.	1/100 [m/s] 0...1000000 (0...10000m/s)	
55...56	INT16U reserved	Always 20	20	3)
57...58	INT16U samplingFreq	ADC sampling frequency. Always 2000.	[kHz] 2000	4)
59	INT8U reserved	Always 5	5	3)
60...61	INT16U zeroMeasValue	PLP: Zero value of the signal found during calibration. See chapter 4.5 "Curve data" for more details.	$\approx 2048$	4)
62...65	reserved	PLP: Undefined		3)
66	INT8U pressUnit	PLP: Compressive strength unit to be displayed on device 0 MPa 1 N/mm <sup>2</sup> 2 kg/cm <sup>2</sup> 3 psi		
67	INT8U convCurveIndex	PLP: Array index of the currently active conversion curve and currently active rebound value.	0...4	
68...312	TConvCurve curves[5]	PLP: Array of 5 conversion curves data structures according to TConvCurve.		
313...322	INT16U reb[5]	PLP: Array of 5 rebound values, each corresponding to one of the 5 conversion curves. The appropriate value is only valid if the corresponding curve is of type SONREB.	1/10 0...1000 (0...100.0)	

- 1) Of practical use only for setup operations executed directly on Pundit Lab/Lab+
- 2) Internally Pundit Lab/Lab+ always stores and uses [m] or [m/s] for distance or speed values
- 3) Reserved values are either for internal use of the device or for use with the Pundit Link PC tool. They always must be left unchanged. Otherwise correct operation of the Pundit Lab/Lab+ device is not guaranteed.
- 4) Read-only values

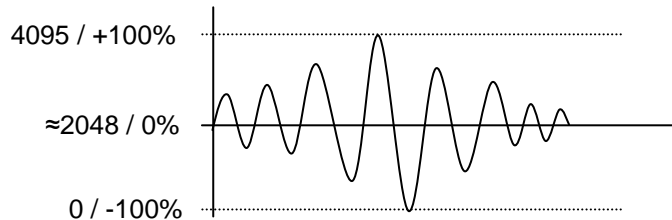
## 4.5 Curve data

The curve data is transmitted as a sequence of INT16U values. Each value represents a 12-bit data sample output by the analog-to-digital converter (ADC). An ADC value ranges from the lowest value of 0 to the maximum value of 4095 (0x0000 ... 0x0FFF).

In this context an ADC value of 0 corresponds to the most negative curve value (-100%) and a value of 4095 corresponds to the most positive curve value (+100%).

Since the sampled signal is bipolar its zero ADC value in theory is 2048. But in practice the value is somewhere around 2048. In addition it is variable for different devices and different probes. The exact zero ADC value for a given device and for given probes gets determined during calibration. To be able to consider this fact the Pundit Lab+ transmits this value with the device setup data (zeroMeasValue).

The following image shows the relation:



## 5. Remote commands description

### 5.1 Overview

Command	ID	Description
SOFTWARE_RESET	0x01	Perform a reset on the Pundit Lab/Lab+
TRIGGER_MEASUREMENT	0x05	Trigger measurement
GET_DEVICE_INFO	0x0A	Get device specific info
GET_DEVICE_SETUP	0x0C	Get device specific setup data
SET_DEVICE_INFO	0x0D	Set device specific setup data
GET_NR_MEASUREMENT	0x0E	Get number of stored measurements on Pundit Lab/Lab+
ERASE_ALL	0x10	Erase all measurements stored on Pundit Lab/Lab+
GET_ALL_MEASUREMENTS	0x11	Download all measurements stored on Pundit Lab/Lab+

### 5.2 Individual remote commands

#### 5.2.1 Software-Reset

SOFTWARE_RESET (ID 0x01)
<b>Description:</b> Performs a reset and makes the device firmware restart.
<b>Command Syntax:</b> 0xC0 0x01
<b>Response:</b> 0x00 ok

#### 5.2.2 Trigger/stop measurement

TRIGGER_MEASUREMENT (ID 0x05)
<b>Description:</b> Triggers a new measurement or stops a currently running measurement.
<b>Command Syntax:</b> 0xC8 0x05 0x01 0xFFFF 0x02 nn NN MM 0x00 nn NN    Number of curve samples to return 0x000            Return only measurement data without any curve samples 0x0001...0x4E20    Return specified number of curve samples (=1...20000) 0xFFFF            Return maximum number of curve samples (20000) MM        0x00            do not increment measurement object ID on Pundit Lab 0x01            increment measurement object ID on Pundit Lab
<b>Response:</b> 0xEF 0x00 ZZ0 ZZ1 ZZ2 yy YY GG HH II JJ KK ...mm MM mm MM.... pp PP 0xEF 0x00    identifier for long data blocks ZZ0 ZZ1 ZZ2    3 bytes overall length information (#bytes, measurement and curve data) including 2 Bytes checksum ( <i>Len1</i> ) 0x000000...0xFFFFFFFF yy YY            2 bytes length information (# bytes) of measurement data ( <i>Len2</i> ) 0x0000 ... 0xFFFF (0 ... 65535) If <i>Len2</i> = <i>Len1</i> - 2 →no curve data has been transmitted GG HH II J... Measurement data as sequence of bytes ("Measurement Data" description) mm MM...        curve data as a sequence of INT16U values pp PP            checksum (CRC-16 over all measurement and curve data)

**Response in case of an error:**

- 0xFB Execution error
- 0xFC Transmission error (timeout)
- 0xFE Error in command parameter

**Example 1:**

Command: Trigger measurement, return 1024 (0x400) curve samples  
 0xC8 0x05 0x01 0xFF 0xFF 0x02 0x00 0x04 0x01 0x00

Response: 0xEF 0x00 0x36 0x08 0x00 0x32 0x00 0x00 0x00 0x00 0x03 0x06 0xAA ...  
 ... 0x01 0x08 0x00 0x08 0x01 0x08 0x03 0x08 ... 0x32 0x46

A
B
C

D
E
F

- A: Overall length information (0x000836 → 2102 bytes)
- B: Length information of measurement data (0x0032 → 50 bytes)
- C: Measurement data (50 bytes)
- D: Curve sample 1
- E: Curve sample 2
- F: Checksum

**Example 2:**

Command: Trigger measurement, return 16000 (0x3E80) curve samples  
 0xC8 0x05 0x01 0xFF 0xFF 0x02 0x08 0x3E 0x01 0x00

Response: 0xEF 0x00 0xB6 0x3E 0x00 0x32 0x00 0x00 0x00 0x00 0x03 0x06 0xAA ...  
 ... 0x00 0x08 0x00 0x08 0xFF 0x07 0x01 0x08 ... 0x45 0x78

A
B
C

D
E
F

- A: Overall length information (0x007D36 → 32054 bytes)
- B: Length information of measurement data (0x0032 → 50 bytes)
- C: Measurement data (50 bytes); D: Curve sample 1; E: Curve sample 2; F: Checksum

### 5.2.3 Query device information

GET_DEVICE_INFO (ID 0x0A)	
<b>Description:</b> Query device specific information of Pundit Lab/Lab+.	
<b>Command Syntax:</b> 0xC1 0x0A YY YY Device information to query 0x00 Device name 0x01 Device serial number 0x02 Hardware serial number 0x03 Hardware revision number 0x04 Device signature (Pundit Lab/Lab+ : "09000000") 0x05 Firmware version	
<b>Response:</b> GG HH II JJ KK ... 0x00 Device specific information as a NULL-terminated string  0xFC Transmission error (timeout) 0xFE Error in command parameter	
<b>Example 1:</b> Command: Query device name (Pundit Lab) 0xC1 0x0A 0x00 Response: 0x50 0x75 0x6E 0x64 0x69 0x74 0x20 0x4C 0x61 0x62 0x00 P u n d i t L a b <\0>	
<b>Example 2:</b> Command: Query device name (Pundit Lab+) 0xC1 0x0A 0x00 Response: 0x50 0x75 0x6E 0x64 0x69 0x74 0x20 0x4C 0x61 0x62 0x2B 0x00 P u n d i t L a b + <\0>	
<b>Example 3:</b> Command: Query device serial number 0xC1 0x0A 0x01 Response: 0x50 0x4C 0x30 0x31 0x2D 0x30 0x30 0x30 0x2D 0x30 0x30 0x30 0x30 0x00 P L 0 1 - 0 0 1 - 0 0 0 1 <\0>	
<b>Example 4:</b> Command: Query device signature (Pundit Lab/Lab+) 0xC1 0x0A 0x04 Response: 0x30 0x39 0x30 0x30 0x30 0x30 0x30 0x30 0x00 0 9 0 0 0 0 0 0 <\0>	
<b>Example 5:</b> Command: Query firmware version 0xC1 0x0A 0x05 Response: 0x32 0x2E 0x30 0x2E 0x34 0x00 2 . 0 . 4 <\0>	

### 5.2.4 Query device setup data

GET_DEVICE_SETUP (ID 0x0C)	
<b>Description:</b> Query device specific setup data of the Pundit Lab/Lab+.	
<b>Command Syntax:</b> 0xC0 0x0C	
<b>Response:</b> 0xEF 0x00 ZZ0 ZZ1 ZZ2 GG HH II JJ KK LL MM ..... pp PP 0xEF 0x00 identifier for long data blocks ZZ0 ZZ1 ZZ2 3 bytes overall length information (# bytes) including the 2 Bytes of the checksum. 0x000000...0xFFFFFFFF GG HH II... Setup data as sequence of bytes (see "Device Setup Data" description) pp PP checksum (CRC-16 over all setup data excluding length information)	
<b>Response in case of an error:</b> 0xF3 CRC error 0xFC Transmission error (timeout) 0xFE Error in command parameter	
<b>Example:</b> Command: Query device setup 0xC0 0x0C  Response: 0xEF 0x00 0x3D 0x00 0x00 0x10 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 <div style="display: flex; justify-content: space-around; width: 100%;"> <span>A</span> <span>B</span> </div> 0x00 0x00 0x00 0x00 0x00 0x20 0x4E 0x00 0x00 0x3A 0x00 0x98 0x3A 0x00 0x00 0x64 0x00 0xEC 0x00 0x00 0x00 0x00 0x5D 0x00 0x64 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x02 0x00 0x20 0x4E 0x00 0x00 0x00 0x00 0x00 0x00 0x14 0x00 0xD0 0x07 0x05 0xCA 0x6F <div style="display: flex; justify-content: space-around; width: 100%;"> <span>C</span> <span>D</span> </div>	
A: Identifier for long data blocks B: Overall length information C: Device setup data D: Checksum	

### 5.2.5 Set device setup data

SET_DEVICE_SETUP (ID 0x0D)
<b>Description:</b> Sets device specific setup data on Pundit Lab/Lab+.
<b>Command Syntax:</b> 1. Pre-Command: 0xC2 0x0D xx XX xx XX           length information of the data sent in the following data command 2. Data-Command: GG HH II JJ KK GG HH II ... Setup data as sequence of bytes (see "Device Setup Data" description )
<b>Response:</b> 0x00 ok 0xFC Transmission error (timeout) 0xFE Error in command parameter (command length mismatch, parameter out of range)
<b>Important note!</b> After each command (pre-/data-command) the device returns a response containing one of the above codes. After reception of the pre-command's 'ok' response the application must send the data-command within <b>200ms!</b> Otherwise the device will respond with an error code.
<b>Example:</b> for Pundit Lab Pre-Command: 0xC2 0x0D 0x3B 0x00  Response 1: 0x00  Data-Command: 0x20 0x00 0x01 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x20 0x4E 0x00 0x00 0x4E 0x98 0x3A 0x00 0x00 0x64 0x00 0xEC 0x09 0x00 0x00 0x00 0x00 0x5D 0x00 0x64 0x00 0x00 0x00 0x00 0x01 0x00 0x03 0x02 0x00 0x20 0x4E 0x00 0x00 0x00 0x00 0x00 0x00 0x14 0x00 0xD0 0x07 0x05  Response 2: 0x00

### 5.2.6 Get number of stored measurements

GET_NR_MEASUREMENT (ID 0x0E)
<b>Description:</b> Query number of stored measurements on Pundit Lab/Lab+.
<b>Command Syntax:</b> 0xC0 0x0E
<b>Response:</b> 0x02 zz ZZ   2 bytes response zz ZZ 0x0000 ... 0xFFFF   number of stored measurements  0xFC Transmission error (timeout) 0xFE Error in command parameter



### 5.2.7 Erase all measurements

ERASE_ALL (ID 0x10)
<b>Description:</b> Erase all measurements stored on Pundit Lab/Lab+. All measurement data will be lost.
<b>Command Syntax:</b> 0xC1 0x10 SS SS: Treatment of device setup on Pundit Lab/Lab+ 0x00 Leave device setup data unchanged 0x01 Set default device setup
<b>Response:</b> 0x00 ok 0xFB Execution error 0xFC Transmission error (timeout) 0xFE Error in command parameter

### 5.2.8 Download all measurements

GET_ALL_MEASUREMENTS (ID 0x11)
<b>Description:</b> Download all measurement data stored on Pundit Lab/Lab+.
<b>Command Syntax:</b> 0xC0 0x11
<b>Response without measurement data:</b> 0x00 No measurement data stored on Pundit Lab/Lab+
<b>Response with measurement data:</b> <b>The message sent by the Pundit Lab/Lab+ consists of 3 parts:</b>
1. <b>Message header:</b> 0xEF 0x00 ZZ0 ZZ1 ZZ2... 0xEF 0x00 Identifier for long data blocks ZZ0 ZZ1 ZZ2 3 bytes overall length information (# bytes) including 2 Bytes overall checksum, 0x000000...0xFFFFFFFF
2. <b>1 to n measurement data sets:</b> ...0xEF 0x00 XX0 XX1 XX2 yy YY GG HH II JJ KK ...mm MM mm MM.... pp PP... 0xEF 0x00 Identifier for long data blocks XX0 XX1 XX2 3 bytes length information for one measurement including 2 bytes checksum, 0x000000...0xFFFFFFFF yy YY 2 bytes length information for measurement data, 0x0000...0xFFFF GG HH II J... Measurement data as sequence of bytes (see "Measurement Data" description) mm MM... curve data as a sequence of INT16U values pp PP checksum (CRC-16 over one measurement data set)
3. <b>Overall Checksum:</b> .... qqQQ overall checksum (CRC-16 over all data)
<b>Response in case of error:</b> 0xFC Transmission error (timeout) 0xFE Error in command parameter
<b>Example:</b> see example for TRIGGER_MEASUREMENT

Proceq Europe  
Ringstrasse 2  
CH-8603 Schwerzenbach  
Phone +41-43-355 38 00  
Fax +41-43-355 38 12  
info-europe@proceq.com

Proceq UK Ltd.  
Bedford i-lab, Priory Business Park  
Stannard Way  
Bedford MK44 3RZ  
United Kingdom  
Phone +44-12-3483-4515  
info-uk@proceq.com

Proceq USA, Inc.  
117 Corporation Drive  
Aliquippa, PA 15001  
Phone +1-724-512-0330  
Fax +1-724-512-0331  
info-usa@proceq.com

Proceq Asia Pte Ltd  
12 New Industrial Road  
#02-02A Morningstar Centre  
Singapore 536202  
Phone +65-6382-3966  
Fax +65-6382-3307  
info-asia@proceq.com

Proceq Rus LLC  
Ul. Optikov 4  
korp. 2, lit. A, Office 321  
197374 St. Petersburg  
Russia  
Phone/Fax + 7 812 448 35 00  
info-russia@proceq.com

Proceq Middle East  
P.O. Box: 262419  
Jebel Ali Free Zone  
Dubai, United Arab Emirates  
Phone +971 50 482 9510  
info-middleeast@proceq.com

Proceq SAO Ltd.  
South American Operations  
Rua Haddock Lobo, 746 - 5 andar  
Cerqueira Cesar, São Paulo  
Brasil Cep. 01414-000  
Phone +55 11 3083 38 89  
info-southamerica@proceq.com

Proceq China  
Unit G, 10th Floor, Huamin •Empire Plaza  
No. 728, Yan An Road(W)  
Shanghai, 200050  
Phone +86 21-63177479  
Fax +86 21 63175015  
info-china@proceq.com

**www.proceq.com**

Subject to change without notice.

Copyright © 2010 by Proceq SA, Schwerzenbach