

Pundit Lab Remote Control Interface

Documentation

This document describes the Pundit Lab remote control interface

Key Words: Pundit Lab, Communication, Documentation

Document Revision: 1
Revision Date: 12-April-2010
Document State: draft
Author(s): Thomas Schlegel thomas.schlegel@proceq.com
 Enrico Castignone enrico.castignone@proceq.com

Company: Proceq SA
 Ringstrasse 2
 CH-8603 Schwerzenbach
 Switzerland

Classification: for internal use and customers who bought a Pundit Lab package

Revision History

Rev	Date	Author, Comments
01	02-March-2010	Thomas Schlegel, pre-released for review
02	12-April-2010	Enrico Castignone, Detailed description of serial interface related issues, data structures and individual commands

Table of Contents

Pundit Lab Remote Control Interface.....	1
Documentation	1
1. Introduction.....	3
1.1 Scope / intended use.....	3
1.2 Boundary conditions.....	3
1.3 Abbreviations.....	3
1.4 Syntax, style.....	Error! Bookmark not defined.
1.5 Open points in this document release.....	Error! Bookmark not defined.
2. Overview	4
2.1 Purpose	4
2.2 Notation	4
2.3 Requirements, prerequisites, limitations	4
2.4 Suggestions, bugs, improvements	4
3. Serial interface.....	5
3.1 Virtual COM port driver.....	5
3.2 Serial Interface parameters.....	5
3.3 Detect Pundit Lab device	5
4. Data structures	6
4.1 Data types	6
4.2 Measurement data	6
4.3 Device setup data.....	7
4.4 Curve data.....	9
5. Remote commands description.....	9
5.1 Overview.....	9
5.2 Individual remote commands	9

1. Introduction

1.1 Scope / intended use

This document describes the Pundit Lab remote control interface. It consists of a set of binary commands which allow the user to configure the Pundit 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 remote functionality into their own PC applications.

1.2 Boundary conditions

Software versions described in this document:

Pundit Lab: Firmware Version: 1.0.6
General: See [2.3 Requirements, prerequisites]

1.3 Abbreviations

Term	Description
HW	Hardware
SW	Software
USB	Universal serial bus
UART	Universal Asynchronous Receiver Transmitter
TBD	To Be Defined

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 device in a most flexible way according to their needs. Possible applications are:

- Integration of Pundit Lab into a (semi-) automated production line.
- Controlling Pundit Lab and collecting measurement data from Pundit Lab 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 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 is powered on and connected to the PC
- Do not disconnect the Pundit Lab while communication with the PC is active
- Do not try to start/stop measurements locally by pressing any Pundit 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.

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

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 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 uses a UART, which is an asynchronous, serial interface (similar to RS232). A so called USB bridge device on the Pundit Lab makes the conversion from UART to USB signals and vice versa.

On the PC side there it is necessary to install a virtual COM port driver, which causes the Pundit Lab (USB) device to appear as an additional COM port available to the PC. This allows application software running on the PC to access the Pundit 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 device, the following serial interface parameters must be set:

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

3.3 Detect Pundit Lab device

When the Pundit Lab device is connected to a PC, the Windows operating system automatically assigns a 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 is connected to that particular port. Once connected, all of the remote commands described in this document can be sent to the Pundit Lab.

4. Data structures

4.1 Data types

The following table lists the types used to characterize data exchanged between the Pundit 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 Measurement data

The following table lists the sequential data bytes that are received from Pundit Lab when a measurement is triggered remotely. They are described in the order received.

Byte-Nr	Structure Element	Description / Range
1	INT8U version	Structure version 0x10 for firmware version V1.0.6
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
15...16	INT16U corrFactor	Correction factor (temperature, moisture, ...) 1/100 70 ... 130 → 0.7 ... 1.3
17...18	INT16U pulseLength	Pulse length of transmitter trigger impulse 1/10 [µs] 1 ... 1000 → 0.1 ... 100.0 µs
19	INT8S pulseAmpl	Amplitude of transmitter trigger pulse [V] -1 Undefined 0 125 V 1 250 V 2 350 V 3 500 V 4 AUTO (amplitude is set automatically by Pundit Lab)
20	INT8S probeFreq	Resonance frequency of connected probes [kHz] -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

21...24	INT32U measDistance	Distance between measurement probes (=path length) [m] 9If propSpeed is given this value will be calculated as result of the measurement
25...28	INT32U crackDepth	Crack depth [mm] 0 ... 9999
29...32	INT32U propTime1	Propagation time 1, which is the result of a me
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.
37...40	INT32U propSpeed	Propagation speed [m/s] If measDistance is given this value will be calculated as result of the measurement
41	INT8U rxProbeGain	Receiver probe gain -1 Undefined 0 1 1 10 2 100 3 AUTO (gain is set automatically by Pundit Lab)
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 / 250 / 350 / 500
47...48	INT16U rxProbeGainValue	Integral value of probe gain (useful if rxProbeGain = AUTO) 1 / 10 / 100
49...50	INT16U reserved	Always 0

4.3 Device setup data

The following table lists the sequential data bytes that are received from Pundit Lab when it's setup is interrogated remotely. They are described in the order received.

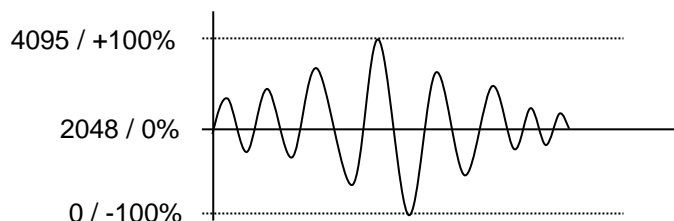
Byte-Nr	Structure Element	Description / Range	Note
1	INT8U version	Structure version	
2	INT8U reserved1	always 0	3)
3...6	INT32U measId	Current measurement object id Gets incremented automatically by Pundit Lab firmware	
7...10	INT32U nrOfStoredMeas	Number of measurements stored on Pundit Lab	
11...14	INT32U reserved2	always 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, ...) 1/100 70 ... 130 → 0.7 ... 1.3	
29...32	INT32U calibTime	Target calibration time 1/100 [μs] Corresponds to the calibration time printed on the calibration stick.	1)
33...34	INT16S calibTimeOfs	Calibration time offset 1/100 [μs] Gets calculated when the calibration procedure is executed, which is not possible to be done by means of a remote command.	1)
35...36	INT16U pulseLength	Pulse length of transmitter trigger pulse 1/10 [μs]	

37...40	INT32U reserved3	always 0	3)
41	INT8U lenUnit	Length unit to be displayed on Pundit Lab 0 Undefined 1 m 2 ft	2)
42	INT8U intRxProbeGain	Receiver probe gain -1 Undefined 0 1 1 10 2 100 3 AUTO (gain is set automatically by Pundit Lab)	
43	INT8U reserved4	always 0	3)
44	INT8U pulseAmpl	Amplitude of transmitter trigger pulse [V] -1 Undefined 0 125 V 1 250 V 2 350 V 3 500 V 4 AUTO (amplitude is set automatically by Pundit Lab)	
45	INT8U probeFreq	Resonance frequency of connected probes [kHz] -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	
46	INT8U measMode	Measurement mode -1 Undefined 0 Continuous 1 Burst	
47...50	INT32U measDistance	Distance between measurement probes 1/100 [mm] 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. 0 ... 999999	
51...54	INT32U propSpeed	Propagation speed 1/100 [m/s] 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. 0 ... 1000000	
55...56	INT16U reserved5	Always 20	3)
57...58	INT16U samplingFreq	ADC sampling frequency [kHz] Always 2000	
59	INT8U reserved6	Always 10	3)

- 1) Of practical use only for setup operations executed directly on Pundit Lab
- 2) Internally Pundit Lab always stores and uses [m] or [m/s] for distance or speed values
- 3) Reserved value must always be left unchanged. Otherwise correct operation of the Pundit Lab device is not guaranteed.

4.4 Curve data

The curve data is transmitted as a sequence of INT16U values. Each value represents an analog-to-digital converted 12-bit data sample, ranging from the lowest value of 0 to the maximum value of 4095 (0x0000 ... 0x0FFF). In this context the value 0 corresponds to the most negative curve value whereas 4095 corresponds to the most positive curve value. 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
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
ERASE_ALL	0x10	Erase all measurements stored on Pundit Lab
GET_ALL_MEASUREMENTS	0x11	Download all measurements stored on Pundit Lab

5.2 Individual remote commands

5.2.1 Software-Reset

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

5.2.2 Trigger/stop measurement

TRIGGER_MEASUREMENT (ID 0x05)	
Description: Triggers a new measurement or stops a currently running measurement.	
Command Syntax: 0xC8 0x05 0x01 0xFFFF 0x02 nn NN MM 0x00	
nn NN	Number of curve samples to return
	0x0000 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

5.2.3 Query device information

GET_DEVICE_INFO (ID 0x0A)	
Description: Query device specific information of Pundit Lab.	
Command Syntax: 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 0x05 Firmware version	
Response: GG HH II JJ KK ... 0x00 Device specific information as a NULL-terminated string 0xFC Transmission error (timeout) 0xFE Error in command parameter	
Example 1: Command: Query device name 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> 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>	

5.2.4 Query device setup data

GET_DEVICE_SETUP (ID 0x0C)	
Description: Query device specific setup data of the Pundit Lab.	
Command Syntax: 0xC0 0x0C	
Response: 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 2 Bytes 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)
Response in case of an error:	
0xF3	CRC error
0xFC	Transmission error (timeout)
0xFE	Error in command parameter
Example:	
Command:	Query device setup 0xC0 0x0C
Response:	<pre> 0xEF 0x00, 0x3D 0x00 0x00 0x10 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 A 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 B C </pre>
A:	Overall length information
B:	Device setup data
C:	Checksum

5.2.5

5.2.7 Erase all measurements

ERASE_ALL (ID 0x10)
Description: Erase all measurements stored on Pundit Lab. All measurement data will be lost.
Command Syntax: 0xC1 0x10 SS SS: Treatment of device setup on Pundit Lab 0x00 Leave device setup data unchanged 0x01 Set default device setup
Response: 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)
Description: Download all measurement data stored on Pundit Lab.
Command Syntax: 0xC0 0x11
Response without measurement data: 0x00 No measurement data stored on Pundit Lab
Response with measurement data: The message sent by the Pundit Lab consists of 3 parts:
1. Message header: 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. 1 to n measurement data sets: ...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. Overall Checksum: qqQQ overall checksum (CRC-16 over all data)
Rückantwort im Fehlerfall: 0xFC Transmission error (timeout) 0xFE Error in command parameter
Example: see example for TRIGGER_MEASUREMENT