

Resipod Remote Control Interface Documentation

This document describes the Resipod remote control interface

Key Words: Resipod, Communication, Documentation

Document Revision: 1
Revision Date: 19-January-2012
Document State: released
Author(s): Peter Stierli peter.stierli@proceq.com
David Corbett david.corbett@proceq.com

Company: Proceq SA
Ringstrasse 2
CH-8603 Schwerzenbach
Switzerland

Classification: for internal use and customers who bought a Resipod package

Revision History

Rev	Date	Author, Comments
01	19-January-2012	David Corbett

Table of Contents

Resipod Remote Control Interface	1
Documentation	1
1. Introduction	3
1.1 Scope / intended use	3
1.2 Boundary conditions	3
1.3 Abbreviations	3
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 Resipod device	5
4. Remote commands description	6
4.1 Request device data	6
4.2 Measurement command	6

1. Introduction

1.1 Scope / intended use

This document describes the Resipod remote control interface. It consists of a set of binary commands which allow the user to extract data from the Resipod device and to collect measurement data using a remote PC.

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

1.2 Boundary conditions

Software versions described in this document:

Resipod: Firmware Version: 1.0.4
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 Resipod device in a most flexible way according to their needs. A possible application is:

- Controlling Resipod and collecting measurement data from Resipod using own software and databases to be further processed or stored without the need of the Resipod Link software and the overhead of unneeded manual work.

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). Or as a combination of hexadecimal and ASCII syntax.

2.3 Requirements, prerequisites, limitations

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

- Make sure the Resipod is powered on and connected to the PC
- Do not disconnect the Resipod while communication with the PC is active
- Do not try to start/stop measurements locally by pressing any Resipod 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 Resipod.

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 Resipod 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 Resipod uses a UART, which is an asynchronous, serial interface (similar to RS232). A so called USB bridge device on the Resipod 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 Resipod (USB) device to appear as an additional COM port available to the PC. This allows application software running on the PC to access the Resipod 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 Resipod 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 Resipod device, the following serial interface parameters must be set:

- 19200 baud
- 8 data bit
- 1 stop bit
- No parity

3.3 Detect Resipod device

When the Resipod 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_Instrument_Short_ID` it's possible to see if the Resipod is connected to that particular port.

Once connected, all of the remote commands described in this document can be sent to the Resipod.

4. Remote commands description

4.1 Request device data

Get Instrument Short ID
Description: Returns device specific information.
Command Syntax: 0x10 ID 0x0D HEX ASCII HEX
Response (ASCII): >Resipod;1.0.4;RP01-001-0001\r Name; Firmware Version; Serial Number

Get Instrument Long ID
Description: Returns device specific information.
Command Syntax: 0x10 @ID@ 0x0D HEX ASCII HEX
Response (ASCII): >Resipod;A1;RP01-001-0001;0A000000;1.0.4;0.0.0\r Name; HW Index; Serial Number; Signature; Firmware Version; OS Version

4.2 Measurement command

Readout
Description: The first Readout command switches the instrument from charge (USB connected) to measurement mode and charging. (The device display will light). This requires approximately 2 s. Following this, subsequent Readout commands will return measurement values.
Command Syntax (HEX): 0xC1 0xD2 0x21
Response (HEX): 0x02 0x00 0x00 Data Lowbyte Data Highbyte Readout data is 16 bits: Bit 0...10 0..1999kΩcm Bit 11 decimal point (0: 1999 kΩcm, 1: 199.9 kΩcm) Bit 12 derivated Bit 13..15 0=OL, 1-5=10-50μA, 6=200μA

resipod

Date: 19. January 2017

© Copyright 2012, PROCEQ SA

proceq

Made in Switzerland

... more than 50 years of know-how you can measure!