SERIAL INTERFACE - RS232

COMMUNICATION SETTINGS

Asynchronous communication, 9600 baud, 8bit, 1 start bit, 1 stop bit, no parity
Voltage level: 2,85V DC (Max 3,3V DC)

SERIAL PROTOCOL

Communitcation package:

‹SNHi› ‹SNMi› ‹SNLo›‹NetID›‹ADRHi›‹ADRLo› ‹R/W-AMOUNT›‹DATA1›‹DATAx›‹CHKSM›

Pump serial number: *‹SNHi›‹SNMi› ‹SNLo›*
Pump serial number, if you wish to address a pump in a system with more than one pump these values can be set from 1 – 16777215. However, the general call is 0 which will be default.

Pump netId: *‹NetID›*
Can also be used to address single pumps with a value from 1-255. 0 will be used as general call and that is also the default value.

Memory address (EEPROM and RAM): *‹ADRHi›‹ADRLo›*
ADRHi uppermost 2 bits is setting for which memory type to access. Lower 6 bits of ADRHi and ADRLo byte forms the memory address.
ADRHi=00XXXXXX=RAM
ADRHi=01XXXXXX=EEPROM
Memory address pointer=(ADRHi AND 00111111)*256+ADRLo

Amount of data: *‹R/W-AMOUNT›*
Number of bytes to read or write to either the RAM or the EEPROM
‹R/W-AMOUNT› = 00XXXXXX = Read from memory
‹R/W-AMOUNT› = 10XXXXXX = Write to memory
Number of bytes to read/write is the lower 6bits + 1. So if you want to read 1byte the lower 6 bits should be 0 (0+1=1), if they are set to 63 (bin XX111111) you will read 64bytes (63+1=64).

Data bytes: *‹DATAx›*
Number of databytes defined in ‹R/W-AMOUNT›.

Package checksum: *‹CHKSUM›*
Add every byte in the package (without carry/overflow) to calculate checksum.

Response from pump:

If a read command is sent to the pump a 3 byte response will be sent. The pump will at least send an integer value and a checksum. Number of bytes is defined by the chosen amount data.

‹DATA1›‹DATAX›‹CHKSUM›

If a write command is executed the pump will answer ‹165› if the command was successful executed and ‹90› if it failed.

SERIAL COMMANDS

Flow control (pump frequncy adjustment):

The pump frequency, and thus the flow, can be changed independently of the applied voltage. This is an important feature that enables flow control independently of supply voltage and thus a lower flow can be achieved without reducing maximum vacuum/pressure. This also reduces current consumption and the need of flow restrictors. It is both costeffective and power efficient.

Command package:
‹0› ‹0› ‹0› ‹0› ‹1› ‹126› ‹129› ‹Lo Byte› ‹Hi Byte› ‹CHKSM›

The frequency is defined as a 16-bit number and is actually a delay time between each new pump stroke.

Highest frequency/highest flow: 16-bit number = 0 (Default calibrated flow)
Lowest frequency/lowest flow: 16-bit number = 65 535

Example:
‹0› ‹0› ‹0› ‹0› ‹1› ‹126› ‹129› ‹232› ‹3› ‹235›
16-bit value = 1000

Response of command was successfully written: ‹165›

Read firmware: *‹0›‹0›‹0›‹0›‹192›‹0›‹1›‹0›‹0›‹193›*
Get the firmware version of the pump. 1 byte with the checksum of entire flash will be sent from the pump.

Stop pump:
First: *‹0›‹0›‹0›‹0›‹0›‹122›‹129›‹0›‹0›‹251›*
wait for *‹165›*
Then: *‹0›‹0›‹0›‹0›‹0›‹37›‹129›‹0›‹0›‹166›*
wait for *‹165›*

Reset pump: *‹0›‹0›‹0›‹0›‹128›‹0›‹1›‹0›‹0›‹129›*
Pump start with initial start up process.

Information furnished is believed to be accurate and reliable. However, Xavitech assumes no responsibility for the consequences of use of such information nor for any infringement of patents or other rights of third parties which may result from its use. No license is granted by implication or otherwise under any patent or patent rights of Xavitech. Specifications mentioned in this publication are subject to change without notice. This publication supersedes and replaces all information previously supplied. Xavitech products are not authorized for use as critical components in life support devices or systems without express written approval of Xavitech.