Communication Protocol for Fotemp Trafo

Overview

This manual is valid for all Fotemp devices of the second generation. It describes the interface communication protocol using RS232, RS485 or USB (FTDI Max…) interface.

System description

All Optocon devices have an internal RS232 Interface or RS485 bus, USB is only available as converter providing a virtual serial port. Every temperatur value has to be asked separately by the end user software, parameters are set equaly. There is no timed auto-measuring/sending mode implemented in the device.

Serial port settings

The settings for the serial port are: 57600 bps connection with 8 data bits, 1 stop bit, no parity and no flow control. Please make sure to use the correct settings.

Command syntax

Fotemp devices use an ASCII protocol. A Fotemp accepts commands and requestions. Commands start with a colon, requests with a question mark, followed by function number and up to two parameters finalized by a carriage return (<CR>). Inbetween the separator • means <space> (ASCII 0x20). The basic structure of a command is `?XX•CH•PM1•PM2•PM3<CR>` .

Requests are answerd with hash plus function number, channel number (when channel specific function) and the desired value, finalized with carriage return an Line feed <CR><LF> `#XX•CH•PM1<CR><LF>` . Thereafter follows a positive ACK `*00<CR><LF>` .

In case of an error there only is a negative ACK from the device `*FF<CR><LF>` . Commands only get an ACK – positive or negative.

Content

Commands

This chapter describes all commands an requests known by Optocon Fotemp devices of the 2nd generation. Some commands depend on the devices' hardware configuration.

Temperature Readings

01 –averaged temperture of a channel

Command 01 read only

Parameter 1: Number of channel to be read, channels are numbered from 1 to the channel count of your device.

Request example: `?01•2<CR>` asks the averaged temperature of channel 2. For averaging settings refer to command 53 below.

Answer example: `#01•1•–135<CR><LF>*00<CR><LF>` The first parameter indicates the temperature state, where 1 means „new temperature" and 0 means the temperature has already been read. In the second parameter the temperature is encoded as 1/10 degree celcius: in this case -13.5°C. Negative temperature values are prefixed with a minus (ASCII 0x2D). A channel showing „9999" has a disconnected, switched of or defective sensor.

02 –averaged temperatures of all channels

Command 02 read only.

Request example: `?02<CR>` asks for the averaged temperature of all channels of the device.

Answer example: `#02•234•–114•---•2345<CR><LF>*00<CR><LF>` Every parameter represents the temperature oft he channel in the order of channel numbers (the example shows a 4 channel device). Temperatures below 0°C are prefixed with a minus (ASCII 0x2D). In the example the third channel showing „---" indicates a disconnected, switched off or defective sensor.

03 - current temperature of a channel

Command 03 read only

Parameter 1: Number of channel to be read, channels are numbered from 1 to the channel count of your device.

Request example: `?03•1<CR>` asks for the temperature of channel 1.

Answer example: `#03•1•234<CR><LF>` The first parameter indicates the temperature state, where 1 means „new temperature" and 0 means the temperatur has already been read. In the second parameter the temperature is encoded as 1/10 degree celcius. Negative values are prefixed with a minus (ASCII 0x2D). A channel showing „9999" has a disconnected, switched of or defective sensor.

04 – current temperature of all channels

Command 04 read only

Request example: `?04<CR>` asks for the temperature of all channels of the device.

Answer example: `#04•234•–114•---•2345<CR><LF>*00<CR><LF>` Every parameter represents the temperature oft he channel in the order of channel numbers. Temperatures below 0°C are prefixed with a minus (ASCII 0x2D). In the example the third channel showing „---" indicates a disconnected, switched off or defective sensor.

05 –current temperature of a channel with datetime

Command 05 read only

This command only makes sense on devices with integrated RTC, which is currently just Fotemp Trafo. Other devices will reply with `*FF<CR><LF>` .

Parameter 1: Number of the channel to be read, channels are numbered from 1 up to the channel count of your device.

Request example: `?05•6<CR>` asks for the temperature of channel 6 and its time of measurement with a precision of 1 second.

Answer example: `#05•1•456•14110412132456<CR><LF>*00<CR><LF>` The first parameter indicates the temperature state, where 1 means „new temperature“ and 0 means the temperature has already been read.
In the second parameter the temperature is encoded as 1/10 degree celcius: in this case -13.5°C. Negative temperature values are prefixed with a minus (ASCII 0x2D). A channel showing „9999“ has a disconnected, switched of or defective sensor.
The last parameter encodes the date and time of measurement. Two signs are one value of the date & time, as shown in the example:

Within -40°C and +85°C the RTC has an accuracy of ±6 ppm (±18 seconds per month). It’s calendar is aware of leap years.

Device Information

0F – channels of the device

Command 0F read only

Request example: `?0F<CR>` asks for the number of channels the device actually has, regardless of their state.

Answer example: `#0F•8<CR><LF>*00<CR><LF>` The only parameter indicates the channel count: 2 in the example. For todays Fotemps values between 1 and 8 are valid replies.

10 – currently active channels

Command 10 read and write.

READ:

Request example: `?10<CR>` asks the currently active channels of the device.

Answer example: In `#10•0B<CR><LF>*00<CR><LF>` the parameter the active channels are encoded bitwise, the value is presented hexadecimal. In case of the example 0x0B is in binary 0000 1011. The lowes bit represents channel 1, the highest 8, which lets the example tell channels 1, 2 and 4 are switched on, whereas channels 3, 5, 6, 7 and 8 are switched off.

WRITE:

Parameter 1: In parameter 1 the switched on and off channels are encoded. The bits of the ASCII-encoded hexadecimal byte are the channels, with bit0 representing channel 1 to bit7 for channel 8.

Command example: `:10•1E<CR>` will switch on channels 2, 3, 4 and 5. Channels 1, 6, 7 and 8 are swithed off. The binary conversion of 0x1E is 0001 1110.

40 – model name

Command: 40 read only

Request example: `?40<CR>` asks for the model name of the device.

Answer example: `#40•43•4F•4D•50•32<CR><LF>*00<CR><LF>` Every parameter encodes a byte value of the model name. Each value is an ASCII-code. In case of the example it reads „COMP2".

41 – serial number

Command: 41, read only.

Request example: `?41<CR>` asks for the serial number of the device.

Answer example: `#41•30•30•31•30•30•32•31<CR><LF>*00<CR><LF>` Every parameter encodes a byte value of the serial number where each value is an hexadecimal ASCII-code. In case of the example it reads „0010021".

42 – firmware version

Command: 42 read only.

Request example: `?42<CR>` asks for the firmware version on the device.

Answer example: `#42•32•2E•31•30•34<CR><LF>*00<CR><LF>` Every parameter encodes a byte value of the firmware version, where each value is an hexadecimal ASCII-code. In case of the example it reads „2.104".

Parameters

53 – temperature averaging

Command: 53 Read and write

READ:

Parameter 1: Channel number of which the averaging count for the moving average is asked. For compatibility reasons the parameter is optional. Up to firmware version 2.102 averaging can only be set for all channels at once and can therefor only be read at once. Leaving the channel blank in firmware versions 2.103 and newer will give the averaging count of the currently measuring channel.

Request example: `?53•3<CR>` asks the number of temperatures taken for averaging on channel 3

Answer example: `#53•3•4<CR><LF>*00<CR><LF>` The first parameter is the channel number, the second indicates the number of temperature values used for calculating the moving average temperature, 4 is the factory default.

WRITE:

Parameter 1: Channel number of which the averaging count for the moving average is asked. For compatibility reasons the parameter is optional. Up to firmware version 2.102 averaging can only be set for all channels at once. Leaving the channel blank in firmware versions 2.103 and newer will set the averaging for all channels at once. Averaging counts from 2 up to 20 are valid.

Parameter 2: Count of temperature values to calculate the moving average.

Command example: `:53•3•5<CR>` will set the averaging count of channel 3 to 5 temperature values taken for calculating the moving average.

75 – temperature offset

Command 75 read and write.

READ:

Parameter 1: Channel number from which to read the temperature offset.

Request example: `?75•4<CR>` asks temperature offset of channel number 4.

Answer example: In `#75•001E<CR><LF>*00<CR><LF>` the parameter the 2 byte signed temperature offset is encoded as hexadecimal value. The example 0x001E decodes to 30_{10}, which is in tenth of Kelvin: +3.0 K. Giving an example for Offsets below `#75•FFE6<CR><LF>*00<CR><LF>` zero would decode to -26, meaning -2.6 K.

WRITE:

Parameter 1: Channel number on which to set the temperature offset.

Parameter 2: is the hexadecimal value in a fraction of tenth **to be added to the current temperature offset** of the channel.

Command example: `:75•4•000B<CR>` will add another 1.1 K to the temperature offset of channel number 4. To give a more descriptive example, suppose the temperature offset is set to 3.0 K. Sending 0x15 (2.1 K) to channel 4 lets the temperature offset add up to 5.1 K. To achieve an offset of 0 K one needs to add -5.1 K: as signed 2-byte value in hex it is 0xFFCD.

81 – analog output boundaries

Command 81 read and write.

With this command the analog output boundarys of each channel can be read and set. Between the low and the high boundary in degree celcius the analog current or voltage (depending on your devices configuration) is linaer to the temperature. Temperatures below keep the analog output at 4 mA / 0 V, above the output is kept at 24 mA / 10 Volts. Disconnected, switched off or defective sensors set the output to 24 mA / 10 Volts.

This command only makes sense on Fotemp devices equipped with analog outputs, but regardless of the devices configuration the command can be executed.

READ:

Parameter 1: is the channel number for which the boundaries are to be set. Channels are numbered from 1 to the channel count of your device. This parameter is optional, leaving it blank will result in as many answers as channels in your device, with the boundaries of each channel.

Request example: `?81•3<CR>` asks the analog output boundaries of channel number 3.

Answer example: `#81•3•FF9C•012C<CR><LF>*00<CR><LF>` The first parameter is the channel number for which the boundaries are shown. Parameters 2 and 3 are the low and the high analog output boundary. They are hexadecimal encoded in tenth of degree celcius. In case of the example the low boundary is -10°C and the high is 300 °C. Factory preset is 0°C to 300°C.

WRITE:

Parameter 1: channel number for which the boundaries are set. This Parameter is optional, leaving out a channel number results in setting the analog output boundaries for all channels in the device.

Parameter 2: lower analog output boundary in tenth of degree celcius.

Parameter 3: higher analog output boundary in tenth of degree celcius.

Command example: `:81•3•FC18•0064<CR>` will set the lower analog output boundary to -100°C and the higher to +100°C for channel number 3.

82 – relais output boundaries

Command 82 read and write.

With this command the relais output boundarys of each channel can be read and set. The high boundary is the switch on point of relais on rising temperature. With the low boundary the switching off on falling

temperature is controlled. That way a hysteresis can be achieved, but eqal values for both boundaries are also valid.

This command only makes sense on Fotemp devices equipped with relais outputs. On devices without relais outputs it results in an negative ACK: `*FF<CR><LF>` .

Parameter 1: channel number of which the relais output boundaries are asked from. This parameter is optional. Left without there will be send as many repies as channels in the device, each with the boundaries for one channel.

READ:

Request example: `?82•1<CR>` asks the relais output boundaries for channel number 1.

Answer example: `#82•1•00C8•00FF<CR><LF>*00<CR><LF>` The first parameter indicates channel number 1 for which the boundaries are shown. Parameters 2 and 3 are the switch-off and the switch-on relais output boundary. They are decimal encoded in tenth of degree celcius. In the example a rising temperature will switch on the relais at 25.5°C. A falling temperature switches off the relais at 20.0°C.

WRITE:

Parameter 1: channel number for which the boundaries are set.

Parameter 2: switch-off relais output boundary in tenth of degree celcius.

Parameter 3: switch-on relais output boundary in tenth of degree celcius.

Command example: `:82•1•00C6•00CA<CR>` will set the switch-on temperature to 20.2°C and the switch-off temperature to 19.8°C for the relais of channel number 1. It creates a hysteresis of 0.4°C for the relais switching.

Real Time Clock

The integrated real time clock is an optional feature of Fotemp Trafo devices.

90 – real time clock date and time

Command 90 read and write.

Where available this command will read or set the current date and time of the integrated real time clock. Fotemps without real time clock reply with `*FF<CR><LF>` .

READ:

Request example: `?90<CR>` asks the current date and time of the integrate real time clock.

Answer example: `#90•14•11•05•13•12•25•37<CR><LF>*00<CR><LF>` Each parameter encodes a value of the current date and time: it's Thursday 13th of November at twelve o'clock, twenty five minutes and 37 seconds.

14	11	05	12	13	24	56
Yeah	Month	Day of Week	Day of Month	Hour	Minute	Second
From 00=2000 to 83=2083	01=January 12=December	1=Sunday 2=Monday … 7=Saturday	1…30/31 29th February exists in a leap year	24 hour format 00…23	00…59	00…59

Within -40°C and +85°C the RTC has an accuracy of ±6 ppm (±18 seconds per month). It's calendar is aware of leap years.

WRITE:

Parameters: are in the following order: year month, day of week, day of month, hour, minute and seconds. Please refer o the table above.

Command example: `:90•15•01•07•29•15•45•11<CR>` will set the real time clock to Saturday, 29th of January at 15:45 o'clock an 11 seconds.

93 – real time clock timer interval

Command 93 read and write.

Where a real time clock is available a interval timer can be set. Standard use for the timer is the SD-Card data save interval.

READ:

Parameter 1: sets the timer interval in seconds.

Parameter 2: sets a multiplier for a 2nd dependent timer function (special use on customer request), set to 1, when not in use.

Request example: `?93<CR>` asks the real time clock timer interval of the device.

Answer example: `#93•60•3<CR><LF>*00<CR><LF>` states an timer interval of 60 seconds with a multiplier of 3 for a 2nd function.

WRITE:

Parameter 1: sets the the interval time of the real time clock timer in seconds.

Parameter 2: multiplier of the interval time issueing a 2nd function.

Command example: `:93•140•2<CR>` will set the interval of the timer to 140 seconds The and the multiplier is set to 2, meaning the 2nd function is issued every 280 seconds.

Internal Flash Memory

The capability of logging data to internal flash memory is an optional feature of Fotemp Trafo devices.

B0 – read dataset from flash

Command B0 read only

Where available the Fotemp logs temperature data to internal flash memory. If not stated otherwise, its logging interval is set by the real time clock timer interval described above. To read the eldest temperature data from flash this command must be send. Every read request sends a channels temperature and time of measurement.

Request example: `?B0<CR>` asks for the eldest temperature data saved on internal flash.

Answer example: `#B0•60•3<CR><LF>*00<CR><LF>` states an timer interval of 60 seconds with a multiplier of 3 for a 2nd function.

B1 – get dataset count on internal flash

Command B1 read only

Where available the Fotemp logs temperature data to internal flash memory. With this command the count of datasets already saved can be read. A dataset consists of the temperature and time of measurement of all channels of the device.

Reading the data happens channel-wise, so the count of datasets multiplied with the channel count of the device is the number of B0-reads to read all saved datasets.

Request example: `?B1<CR>` asks for the count of datasets saved on internal flash.

Answer example: `#B1•3<CR><LF>*00<CR><LF>` states 3 datasets on internal flash. If it is an 4-channel Fotemp 12 temperatures with their time of measurement and channel number are saved.

B2 – delete datasets

Command B2 write only

This command deletes a given count of datasets on the internal flash memory of the Fotemp. Like only the eldest data on internal flash memory can be read, only the eldest datasets can be deleted too. A dataset consists of the temperature and time of measurement of all channels of the device.

Command example: :B2•2<CR> will delete the two eldest datasets (2*channel count of temperatures) on the internal flash memory.

BF – erase internal flash memory

Command BF write only

With this command the internal flash memory can be reset and cleared. All data is deleted.

Command example: :BF•<CR> will delete the two eldest datasets (2*channel count of temperatures) on the internal flash memory. Please note the trailing <space>, it is necessary!

Document History

Version - date	Comment
Rev.56 - 6/11/2015	Description of commands 01…04 corrected Table of Contents added Layout revised
Rev.52 - 3/4/2015	First publication