

Communication Protocol Manual JOFRA DTI

© Copyright 2003 AMETEK DENMARK A/S

Visit **TRANSCAT**.com >

sales@transcat.com | 800.828.1470

Contents

1.1 SET OF COMMANDS :	4
1.2 CHECK SUM	6
1.3 CONCERNING THE SEQUENCE OF BYTES	6
1.4 WHEN WRITING TO THE DTI	7
1.5 SETUP OF SERIAL PORTS	7
1.6 SOFTWARE - HANDSHAKE	7
1.7 TEST OF DOWNLOADED DATA (SET VALUES)	8
1.8 LOW BATTERY VOLTAGE	8
1.9 REGARDING DOWNLOAD OF SENSOR CONSTANTS	10

1.1 Set of commands:

Protocol for serial communication with DTI firmware version 2.0 :

All commands to the DTI consist of 1 byte. The DTI will always return the command byte immediately after this has been received. The sets of command are as follows:

No.:	Telegram type:	Format:	V1.60	V2.0X
(Read - commands)				
96	Read the firmware version.	1 floating-point number is received.		X
97	Read the resistance value in Ohm of the 2 platinum sensors	8 bytes (2 floating-point numbers) are received.	X	X
98	Read the Celsius temperature of the 2 platinum sensors	8 bytes (2 floating-point numbers) are received.	X	X
99	Read the 4 ITS68 constants in binary format of sensor 1, if present	16 bytes (4 floating-point numbers) and 16 bytes ASCII sensor id. are received. The sequence is: R0, R0*A,R0*B,R0*C.	X	X
100	Read the 4 ITS68 constants in ASCII format of sensor 1	56 bytes are received in ASCII format.	X	
101	Read the 4 ITS68 constants in binary format of sensor 2, if present	16 bytes (4 floating-point numbers) and 16 bytes ASCII sensor id. are received. The sequence is: R0, R0*A,R0*B,R0*C.	X	X
102	Read the 4 ITS68 constants in ASCII format of sensor 2	56 bytes are received in ASCII format.	X	
103	Read the zero point & resolution	8 bytes (2 floating-point numbers) are received (check sums are not returned).	X	X ¹
104	Read the serial number of the instrument	32 bytes are received in ASCII format.	X	X
105	Read the last calibration date.	9 bytes are received.	X	X
106	Read the 8 ITS90 constants in binary format of sensor 1, if present	32 bytes (8 floating-point numbers) and 16 bytes serial No. are received. The sequence is: RTPW, aLR, bLR, cLR, aHR, bHR, cHR, dHR..		X
107	Read the 8 ITS90 constants in binary format of sensor 2, if present	32 bytes (8 floating-point numbers) and 16 bytes serial No. are received. The sequence is: RTPW, aLR, bLR, cLR, aHR, bHR, cHR, dHR..		X

¹ Dummy;

(Set - commands)

48	Set to normal RS232 communication at low battery voltage. At normal battery voltage this command is ineffective	No further bytes are transmitted.	X	X
65	Write new ITS68 constants in binary format of sensor 1	4 x (floating-point + check sum (1 byte)) = 20 bytes sensor constants, 56 bytes sensor constants in ASCII format, and 16 bytes ASCII sensor id., are transmitted. The sequence is: R0, R0*A,R0*B,R0*C.	X	X ²
66	Write new ITS68 constants in binary format of sensor 2	4 x (floating-point + check sum (1 byte)) = 20 bytes sensor constants, 56 bytes sensor constants in ASCII format, and 16 bytes ASCII sensor id., are transmitted. The sequence is: R0, R0*A,R0*B,R0*C.	X	X ³
67	Write new zero point & resolution	2 x (floating-point + check sum (1 byte)) = 10 bytes are transmitted. The 2 floating-point numbers represent the zero point i.e. the number degrees centigrade corresponding to 0 Volt of the analog outputs, as well as resolution in millivolt per degrees centigrade.	X	X
68	Write new ITS90 constants in binary format of sensor 1	8 x (floating-point + check sum (1 byte)) = 40 bytes and 16 bytes ASCII sensor id. are transmitted. The sequence is: RTPW, aLR, bLR, cLR, aHR, bHR, cHR, dHR..		X
69	Write new ITS90 constants in binary format of sensor 2	8 x (floating-point + check sum (1 byte)) = 40 bytes and 16 bytes ASCII sensor id. are transmitted. The sequence is: RTPW, aLR, bLR, cLR, aHR, bHR, cHR, dHR..		X

Example : You wish to read the resistances of the 2 platinum sensors:
Transmit the decimal numeral 128 (i.e. 10000000 binary),

² Contents of sensor constants in ASCII format are dummy.

³ Contents of sensor constants in ASCII format are dummy.

You receive 128 decimal numeral (the command byte is returned)
You receive further 8 bytes through the serial connection.

Comments to the set of commands :

Read & Set commands can be used from the user's own programs. Protected Read and Set commands are not intended to be accessed by other users through their own programs. The enclosed PC program trace.exe uses all commands minus 104 and 162 corresponding to respectively reading and writing of the serial number of the instrument. These two commands can only be reached through lager.exe, which is only for internal use at Ametek Denmark A/S. Hint, when floating point numbers are received the order of the bytes are reversed compared to common practice.

1.2 Check sum

The check sums are not used for testing disturbances of the serial connection, but are used internally by the DTI to test, if certain sets of values have previously been written to the DTI. The check sum is calculated by the following formula :

$$\text{Check sum} = (\text{byte1} + \text{byte2} + \text{byte3} + \text{byte4}) \text{ modulo } 256,$$

where modulo 256 refers to the remainder at the integral division by 256.

1.3 Concerning the sequence of bytes

When transmitting the floating-point numbers, the most important bytes are first transmitted, and upon receipt, the most important bytes are first read. The floating-point numbers are stored in the PC programs at the highest RAM address, the next most important at the next most important place, etc. (Besides the sequence is contrary to the sequence used in the micro controller used by the DTI and belonging to the Motorola M68HC11 family).

1.4 When writing to the DTI

When you write to the DTI, it takes a little while to delete places in the memory, and to write the bytes received on the places that have just been deleted. In the PC program trace.exe, a delay of 200 milliseconds is used between each byte transmitted.

It is not recommended to use a shorter delay than this, as you will otherwise risk errors when the bytes are written.

A delay of 500 milliseconds should be held between successive commands to the DTI.

When calibrating sensors, reference resistances, as well as analog outputs, it is strongly recommended to use the PC program trace.exe for this purpose.

However you can unconditionally use the data collection through the serial connection, e.g. for data collection through your own programs.

1.5 Setup of serial ports

You need a transmission speed of 2400 Baud. The format is 8 data bits + 1 stop bit, even parity and no interruptions. The DTI does not use the parity bit for testing the bytes received through the RS232. Therefore when something is written to the DTI, it has to be re-read for control. When you read from the DTI, the parity bit is always generated. This can be used for testing the individual bytes.

If you write directly to the registers of the serial port, it is recommended to write 1 at the 2 less important bits in the Modem control register. However this is not strictly necessary. Besides the CTS and the DTR are connected to the DTI through the serial cable. So far these 2 connections are not used in the protocol.

1.6 Software - Handshake

Handshake is carried out by returning the first byte (the command byte) that has been transmitted from the PC to the DTI. Both when reading from the DTI and writing to it, the command byte will be returned immediately after the transmission. When writing, the returned byte can be used for testing, whether the DTI is capable of

receiving a series of bytes. If the ASCII signal "?" has been returned (ordinal value 63), the command has been misunderstood due to noise, or the user has transmitted a command, which does exist in the set of commands. If the command has been misunderstood, writing can be interrupted immediately, and you can try again with the same command.

If 1 or more bytes are missing after a Set or Download command, the DTI will wait for the missing bytes for up to 2 seconds. If these bytes still have not been received, the program of the DTI will return to the major lobe, and the DTI is now ready to receive the new command, while the measurements of the 2 sensors are continued.

1.7 Test of downloaded data (Set values)

When the DTI is switched on, it automatically tests whether the previously downloaded data in the floating-point format are correct by calculating their check sum and comparing this result with a 5th byte, which is also stored internally in the DTI. When writing the ASCII texts there is no later test of these texts, of which some or maybe all are shown in the display of the DTI. You can avoid showing erroneous texts by re-reading downloaded texts, by use of the relevant commands, and by comparing these with the correct texts.

1.8 Low battery voltage

The first time the DTI ascertains that the voltage of the battery has fallen menacingly, the following text will be shown in the display "BATTERY LOW PRESS ANY KEY". At the same time, the DTI will switch off the normal RS232 communication. This means that if you transmit any command (apart from 48 ('0' in the ASCII table)) the DTI will only return 48 or '0' and nothing else.

In a PC program this can be used for detecting, whether the battery is falling to a too low level. In this case, the user can choose to terminate the PC program and switch off the DTI and exchange the battery. As the DTI is capable of continuing the measurements for a while, the user can choose to continue his measurements, but he needs to transmit the command 48 ('0') to re-establish the normal RS232 communication. Please note that the latter command does not change the status of the display of the DTI, i.e. the user needs to

press one of the keys, before the result of the measurements (or the choice of menu) is again shown in the display.

1.9 Regarding download of sensor constants

The resistance of each of the platinum sensors can be determined as a function of the Celsius temperature.

$$R(t) = R_0 * (1 + A * t + B * t^2 + C * (t - 100) * t * t^2),$$

$C = 0$ at $t > 0$ °C

where R_0 is the resistance at 0 °C.

When the constants are to be downloaded in a binary form, the constant is R_0 , and the products $R_0 * A$, $R_0 * B$, $R_0 * C$, as the DTI uses these for internal calculations. This is primarily done to save the 2 * 3 multiplications in the built-in microcontroller of the DTI, but also because these 6 calculations can be carried out with a higher precision in e.g. a PC program than in the 4 byte long floating-point number, used internally in the DTI. Besides the PC program trace.exe uses a double precision at all calculations, i.e. floating-point calculations with 8 bytes long numbers.

When the constants are to be downloaded in the ASCII format, the constants R_0 , A , B , C are to be transformed to ASCII strings of 14 characters for each constant, or a total of 56 characters for one set of constants of one of the sensors.

Visit **TRANSCAT**.com >

sales@transcat.com | 800.828.1470