



COMMUNICATION PROTOCOL MANUAL

## Contents

Contents .....	1
TanDelta Communication Protocol .....	2
. Checksum .....	3
. Command Overview .....	4
. Read Current Readings Command "Rr" .....	5
. Read Memory Command "Rm" .....	6
. Read Config Data Command "Rc" .....	7
. Read Version and Serial Number Command "Rv" .....	9

## TanDelta Communication Protocol

The command protocol and language for the TanDelta serial communications uses a binary command format communicating over a half duplex RS232 or RS485 interface. The serial configuration must be set to 8 bit with no parity and will initially communicate at 9600 baud. The TanDelta unit operates in a slave mode, with the serial device which controls the communications (e.g. the monitoring computer) acting as master and issuing commands to which the slave will reply. TanDelta will not transmit any data except in response to a command from the master, and will expect no further commands until the last command has been replied to.

Commands are multiple sequences of bytes which must be interpreted as a complete data string before the correct

action and response can be determined. Any command which is not interpreted and verified will cause the command interpreter to reset back to its initial state, and any interruption of communications of longer than 1s will cause the same result. The master must therefore check for correct response in all cases and re-start any commands which have been corrupted or mis-interpreted. The detailed structure of commands is as detailed below.

The command structure is as follows:-

Byte 0	"!"	wake-up code
Byte 1	<count>	number of bytes to follow, including checksum but excluding <count>
Byte 2	<iaddr>	single byte instrument address
Byte 3&4	<cmd>	two byte command
Bytes n to n+1	<data>	optional, depending on the command
Byte n, n+1	<cksm>	16 bit inverse checksum of all preceding bytes including acknowledge

The response structure is as follows:

Byte 0:	"A" or "E"	Acknowledge character: acknowledgment or Error
Byte 1:	<count>	number of bytes to follow, including checksum but excluding <count>
Bytes 2 to n-1	<response>	optional, depending on the command
Byte n,n+1:	<cksm>	16 bit inverse checksum of all preceding bytes including acknowledge

## Checksum

The checksum is calculated by creating an unsigned 16 bit sum of all preceding data bytes, discarding any overflow and then subtracting the result from 65535. For example the Read Datalog Memory command string required to read 128 records from instrument 2 in cluster 1, starting at record zero, is shown below

"!",10,01,02,"R","d",00,00,127,<cksm> (ASCII ! = 33<sub>D</sub>, ASCII R = 82<sub>D</sub>, ASCII d = 100<sub>D</sub>)

The checksum here would be 65535 -  
(33+10+1+2+82+100+0+0+127) = 65180, or FE9C<sub>H</sub>

## Command Overview

<cmd>	Description	<data>	<response>
"Rc"	Read config settings	2 byte address plus 1 byte <length>	<length> bytes
"Rm"	Read system memory	2 byte address plus 1 byte <length>	<length> bytes
"Rr"	Read current readings	byte address plus 1 byte <length>	<length> bytes
"Rv"	Read version and serial no	2 byte address plus 1 byte <length>	<length> bytes

(see below for detailed description of data and response strings)

Commands are divided into two categories:

- Read commands which read data from the TanDelta unit, beginning "R"
1. Read commands allow access to current channel and system settings for confirmation and management of the units' operation and current measurements (readings) acquired by the units. Note that all commands use exactly the same mechanism as "Read Memory", accessing system memory but adding the address onto a starting address appropriate to the command; thus "Read Channel Settings" with address 0x03 and length 0x06 reads six bytes, starting from the third byte of the Setup structure within system memory.
  2. Write commands allow the remote configuration of current channel and system settings for management of the units' operation, and modification of the units' system memory, for use in monitoring and debugging
- Write commands which write data to the TanDelta unit, beginning "W"

operations only.

<cmd>	Description	<data>	<response>
"Wc"	Write channel settings	2 byte address plus 1 byte <length>	<length> bytes
"Wm"	Write system memory	2 byte address plus 1 byte <length>	<length> bytes

## Read Current Readings Command "Rr"

This command has two items of data, a two byte <starting address> and a single byte <length>. It commands the TanDelta unit addressed by <iaddr> to transmit <length> bytes from <starting address> relative to the start of the current readings array for each of the five channels within the unit. Each reading comprises three bytes of data in 24 bit floating point format and must be interpreted as such by the receiving system. If the command is correctly interpreted, the Tan Delta unit addressed will acknowledge the command with the Ack code, echo the number of bytes it will transmit, and send a response containing the data requested, followed by a checksum. If the command is not correctly interpreted, the TanDelta unit addressed will acknowledge with the Error code and a checksum. If the unit addressed cannot be found, there will be no reply.

The command string is 10 bytes long (no. of bytes in each field as subscript):

**"I", <08<sub>1</sub>>, <iaddr<sub>1</sub>>, "R", "r", <start address<sub>2</sub>>, <length<sub>1</sub>>, <cksm<sub>2</sub>>**

The response string comprises the acknowledge character, the <count> of bytes to follow and then <length> bytes of data, followed by a 16 bit checksum. In its most common usage as used to download all five channel readings, this will comprise 15 bytes of data, as follows:

Bytes 0-2      24 bit floating point representation of Oil Temp value, in C (unless otherwise scaled)

Bytes 3-5      24 bit floating point representation of Ambient Temp value, in C (unless otherwise scaled)

Bytes 6-8      24 bit floating point representation of Oil Condition value, in %

Thus the response string is <length> + 4 bytes long (no. of bytes in each field as subscript):

**"A", <length+2<sub>1</sub>>, <length bytes of data>, <cksm<sub>2</sub>>**

Or, in case of error:

**"E", <02<sub>1</sub>>, <FFA9<sub>2</sub>>**

## Read Memory Command “Rm”

This command has two items of data, a two byte starting <address> and a single byte <length>, allowing a read of up to 256 addresses from system memory. It commands the TanDelta unit addressed by <iaddr> to transmit <length> memory bytes starting at <address>. If the command is correctly interpreted, the TanDelta unit addressed will acknowledge the command with the Ack code, echo the number of bytes it will transmit, and send a response containing all the data requested, followed by a checksum. If the command is not correctly interpreted, or the starting <address> is outside the range of system memory, the TanDelta unit addressed will acknowledge with the Error code and a checksum. If the unit addressed cannot be found, there will be no reply.

The command string is 10 bytes long (no. of bytes in each field as subscript):

**“!”,<08<sub>1</sub>>,<iaddr<sub>1</sub>>,”R”,“m”,<record no<sub>2</sub>>,<length<sub>1</sub>>,<cksm<sub>2</sub>>**

The response string comprises the acknowledge character, the <count> of bytes to follow and then <length> bytes, followed by a 16 bit checksum. The structure of each record is:

Thus the response string is (4 + <length>) bytes long (no. of bytes in each field as subscript):

**“A”,<count<sub>1</sub>>,<byte 1>,<byte 2>,...,<byte n>,<cksm<sub>2</sub>>**

or, in case of error:

**“E”,<02<sub>1</sub>>,<FFA9<sub>2</sub>>**

## Read Config Data Command “Rc”

This command has two items of data, a two byte <starting address> and a single byte <length>. It commands the TanDelta unit addressed by <iaddr> to transmit <length> bytes from <starting address> relative to the start of the channel settings and alarm array for each of the five channels within the unit, and the two relays and eight alarm definitions. Config settings are as defined below and must be correctly interpreted by the receiving system. If the command is correctly interpreted, the TanDelta unit addressed will acknowledge the command with the Ack code, echo the number of bytes it will transmit, and send a response containing the data requested, followed by a checksum. If the command is not correctly interpreted, the TanDelta unit addressed will acknowledge with the Error code and a checksum. If the unit addressed cannot be found, there will be no reply.

The command string is 10 bytes long (no. of bytes in each field as subscript):

“!”,<08<sub>11</sub>>,”R”,“c”,<start  
address<sub>2</sub>>,<length<sub>1</sub>>,<cksm<sub>2</sub>>

The response string comprises the acknowledge character, the <count> of bytes to follow and then <length> bytes of data, followed by a 16 bit checksum. The data returned will be dependent on the particular parameters addressed, as follows:-

Addresses 0 to 2 incl: Calibration Zero Point: 3 byte floating point numbers in volts

Addresses 3 to 14 incl: Temperature Coefficients: 4 \* 3 byte floating point numbers

Addresses 15 to 23 incl: Channel Max Ranges: 5 \* 3 byte

floating point numbers in EU

Addresses 24 to 32 incl: Channel Min Ranges: 5 \* 3 byte floating point numbers in EU

Address 33: Instrument Address: 1 byte address

Address 34: Serial Type: 1 byte: 0 = RS232; 1 = RS485

Addresses 35 to 37 incl: Max Temperature (Ambient): 3 byte floating point number in C

Addresses 93 to 101 incl: Channel Readings: 3 \* 3 byte floating point numbers in EU

Address 256: Software Version Number: 3 byte floating point number

Thus the response string is <length> + 4 bytes long (no. of bytes in each field as subscript):-

**"A",<length+2<sub>1</sub>>,<length bytes of data>,<cksm<sub>2</sub>>**

or, in case of error:-

**"E",<02<sub>1</sub>>,<FFA9<sub>2</sub>>**

## Read Version and Serial Number Command “Rv”

This command has two items of data, a two byte <starting address> and a single byte <length>. It commands the TanDelta unit addressed by <iaddr> to transmit <length> bytes from <starting address> relative to the software Version No. variable. Version and Serial No. information is as defined below and must be correctly interpreted by the receiving system.

If the command is correctly interpreted, the TanDelta unit addressed will acknowledge the command with the Ack code, echo the number of bytes it will transmit, and send a response containing the data requested, followed by a checksum.

If the command is not correctly interpreted, the TanDelta

unit addressed will acknowledge with the Error code and a checksum. If the unit addressed cannot be found, there will be no reply.

The command string is 10 bytes long (no. of bytes in each field as subscript):

**“!”,<08<sub>11address<sub>212</sub></sub>**

The response string comprises the acknowledge character, the <count> of bytes to follow and then <length> bytes of data, followed by a 16 bit checksum. The data returned will be dependent on the particular parameters addressed, as follows:

Addresses 0 to 2 incl:

Software Version No: 3 byte floating point version no.

Thus the response string is  $\langle \text{length} \rangle + 4$  bytes long (no. of bytes in each field as subscript):-

**"A", $\langle \text{length} + 2 \rangle_1, \langle \text{length bytes of data} \rangle, \langle \text{cksm} \rangle_2$**

or, in case of error:

**"E", $\langle 02 \rangle_1, \langle \text{FFA9} \rangle_2$**