TP9000 – iDDS-Translator

DDS COMMAND & CONFIGURATION
MANUAL

e-mail:- info@chell.co.uk

Visit the Chell website at:
http://www.chell.co.uk

900241-1.0

Please read this manual carefully before using the instrument.

 Use of this equipment in a manner not specified in this manual may impair the user's protection.

Contents

1. Introduction.

This manual details the DDS communications protocol available in an appropriately licensed TP9000 DDS Translator. Outlined is the description of the configuration file used over DDS for the device and also a list of commands available which can perform tasks or read current settings, etc. It is expected that the user has some level of knowledge of DDS principles and as such they are not covered here.

The DDS sub-system is provided by the CoreDX middleware from TwinOaks and the Rolls Royce iDDS wrapper which is based around the TEDS Real-time Backbone Wire Protocol - EIM03069 issue 7. The DDS command structure is based around the TEDS Real-time Backbone Command Definitions – EIM03869 Issue 2 Draft 4 [*1], with a few changes as necessary to accommodate some of the additional commands required for the TP9000.

This document version supports V1.0.0 of TP9000 firmware

[*1] It is intended that the TP9000 will support the currently released EIM03869 V2.1 in a future firmware release.

2. DDS Configuration File

The DDS Configuration File is stored on a remote Configuration Server and downloaded by the TP9000 as part of the DDS initialisation routine. The format of the configuration file should be standard XML. What follows is a description of the available element entities with examples of expected values where appropriate.

Branch	Entity	Value example	Description
Header	Manufacturer	Chell Instruments Ltd	This is mandatory
	ManufacturerURL	http://www.chell.co.uk	This is mandatory
	ConfigFileAuthor	Your Name Here	Enter the name of the file creator, usually the calibration or maintenance department of the end user company
	Descriptor	A Sample file	Description for why the file exists or application name, etc
Internal.system	NetScannerIP	192.168.1.178	The IP address of the NetScanner 9216 that is being translated.
	NetScannerSub	255.255.255.0	The Subnet mask of the NetScanner 9216 that is being translated.
	EnPTP	1	IEEE1588 PTP mode. 0=Off, 1=On
	Rate	10	Sets the data stream rate (0 to 15) 0 = Off 8 = 100 Hz 1 = n/a (250Hz) 9 = 50 Hz 2 = n/a (250Hz) 10 = 33 Hz 3 = n/a (250Hz) 11 = 25 Hz 4 = n/a (250Hz) 12 = 20 Hz 5 = 250 Hz 13 = 10 Hz 6 = 200 Hz 14 = 5 Hz 7 = 150 Hz 15 = 1 Hz (Values 1 to 4 are currently not applicable to the TP9000 and are reserved for future use).
	RateMode	0	Sets the stream mode. 0=Silent on reset, 1= Autostream on reset
Channel.n	PressParameterName	ABC12345	Pressure Measurement Parameter Name (This variable is passed to the DDS Config Server to obtain a volatile ParameterID, max 64 chars).
	TempParameterName	ABC12345	Temperature Measurement Parameter Name (This variable is passed to the DDS Config Server to obtain a volatile ParameterID, max 64 chars).

3. DDS Command Definitions

3.1. Introduction

The TP9000 DDS commands are based on and taken from the TEDS Real-time command definitions – EIM03869 Issue 2 Draft 4 – and that document should be referenced alongside this as appropriate. DDS command structure is based on the XML-RPC standard with core functionality incorporated from other standards such as SCPI and XML-DA. It is assumed that the user has some previous knowledge of this terminology prior to reading this manual. Some core structuring is taken from that manual and repeated here for reference.

3.2. Core structures

3.2.1. Request

Command requests use the following structure:

```
<methodCall>
  <methodName>NameOfCommand</methodName>
  <params>
    <param>
      <name>NameOfParameter</name>
      <value><Int16>18</Int16></value>
    </param>
  </params>
</methodCall>
```

The `<methodCall>` root element indicates a command call and not a response and must contain a `<methodName>` element which indicates the name of the command (this should be represented as a 'catalogue name' followed by a decimal point, followed by the name of the command itself, in upper camel case, eg. `General.GetStatus`). The following element is the `<params>` structure which can contain none, one or multiple child elements (`<param>`) depending on the amount of parameters required for a particular command. Each `<param>` element structure consists of a `<name>` child element and a `<value>` child element structure, which consists of the actual value within a value type element (`<Int16>` in the above example).

3.2.2. Response

A response uses the following structure:

```
<methodResponse>
  <code>0</code>
  <params>
    <param>
      <value><Double>18.24668429131</Double></value>
    </param>
  </params>
</methodReponse>
```

The `<methodResponse>` structure contains a `<code>` child element to indicate a success or fault response. Depending on the command sent, the response may also include a `<param>` child structure (as in the example above), but note that the `<params>` parent element will always exist, even if there are no parameters to return (it will be an empty element in this case, ie. `<params/>`).

A fault response will return a non-zero `<code>` value, along with a `<param>` structure with a `<name>` of `message` and a `<String>` value describing the fault code, eg.

```
<methodResponse>
  <code>1</code>
  <params>
    <param>
      <name>message</name>
      <value><String>Command Not Supported</String></value>
    </param>
  </params>
</methodResponse>
```

Possible support fault codes are listed below.

Error code	Error message
0	OK (No Error)
1	Command not supported
2	Parameters incorrect
3	Parameter too low
4	Parameter too high
5	Unable to comply - Incorrect state
6	Unable to comply - Operation in Progress
7	Attribute not found
8	Attribute write failed - Parameter read-only
9	Attribute write failed - Parameter incorrect data type
255	Unspecified Error

3.3. Supported Commands

What follows is a list of all supported commands in the TP9000 with their parameters, types, and descriptions (command and response). Due to the generic nature of EIM03869 (around various types of instrumentation), the TP9000 does not support all commands and some currently only have partial implementation.

EIM03869 Ref	Command	Parameter			Command Description	Response Notes (params returned on command Success, fault code on command Fault)
		Name	Type	Value(s)		
General Catalogue						
12.1.1	General.HardReset	None			Perform an OS reset of the device	Code '0' response occurs on reception of the command, not after command action. Current system implementation means no other operations could cause a corruption, therefore no fault codes can occur.
12.1.2	General.SoftReset	None			Restart the iDDS-Translator application (effectively resets DDS state machine to 'Ready' once initialisation is complete).	Code '0' response occurs on reception of the command, not after command action. Current system implementation means no other operations could cause a corruption, therefore no fault codes can occur.
12.1.3	General.Identify	None			Queries the device identity	6 parameter (Strings) response: Manufacturer: Chell Instruments Ltd Model: iDDS-Translator-DC *or* iDDS-Translator-POE(13W) *or* iDDS-Translator-POE(25W) Serial: <*7 digit serial number*> Device Type: Measurement Equipment Model Version: <*8 digit build version*> Software Version: <x.x.x> (e.g. 1.0.0)
12.1.4	General.StartConfiguring	ConfigMode	String	Man	Switches DDS state machine from 'Ready' to 'Configuring'. Required to issue Configuration commands. Currently only Manual mode is supported.	Fault code '5', if current state is not 'Ready'.
12.1.5	General.StopConfiguring	None			Switches DDS state from 'Configuring' to 'Ready'	Fault code '5', if current state is not 'Configuring'.
12.1.8	General.StartOperating	None			Switches DDS state machine from 'Ready' to 'Operating'. This will start data streaming at the pre-configured rate.	Fault code '5', if current state is not 'Ready'.
12.1.9	General.StopOperating	None			Switches DDS state machine from 'Operating' to 'Ready'. This will stop any currently streaming data.	Fault code '5', if current state is not 'Operating'.
12.1.10	General.NOP	None			No operation command. Ping device.	
12.1.11	General.GetStatus	None			Returns the current DDS state machine state.	1 parameter (String) response: State: <*DDS state*> 2 parameter (String) response, if Faulted state: State: Faulted ErrorDescription: <*error*>

Page 5

Configuration Catalogue (note multiple GetAttribute commands listed to separate different attributes, but could be issued as one multi-parameter command)						
12.2.1	Configuration.ListAttributes	Section	String	*<empty>* Range Rate ChVal ChVal.X DeviceTemp	Retrieves a list of attributes available for a given structure Section. If the Section parameter is omitted/empty then the 'Root' structure is shown. Each element in the list is a comma separated string containing the attribute and its read/write status and its volatility.	1 parameter (ListOfString) response: Attributes: <String>*attribute,read/write,volatility*</String> *read/write* = READ_ONLY, WRITE_ONLY or READ_WRITE, NONE *volatility* = NON_VOLATILE, VOLATILE, NONE NONE indicates the attribute has child elements (and therefore doesn't have a read/write or volatility state itself) Fault code '2', if section name is invalid.
12.2.2	Configuration.GetAttribute	Attribute	String	Range.*X*	Retrieves the readable limits of channel *X* in engineering units as reported by the attached NetScanner at system initialisation. *X*=1-16 for each channel	1 parameter (String) response: Range.*X: <range string>* Example *<range string>* = PSID: -1 to 1 Fault code '2', if channel number is invalid. Fault code '7', if no NetScanner found/connected at startup.
12.2.2	Configuration.GetAttribute	Attribute	String	Rate	Returns the currently set data delivery rate. Additionally indicates if auto streaming on initialisation is set (i.e. TP9000 transitions to 'Operating' state after initialisation)	1 parameter (String) response: Rate: *<ab>*: *<rate>* Hz a = 1 : Auto streaming after init a = 6 : No streaming after init b= 0: Off b= 1: n/a b= 2: n/a b= 3: n/a b= 4: n/a b= 5: 250 Hz b= 6: 200 Hz b= 7: 150 Hz b= 8: 100 Hz b= 9: 50 Hz b= 10: 33 Hz b= 11: 25 Hz b= 12: 20 Hz b= 13: 10 Hz b= 14: 5 Hz b= 15: 1 Hz
12.2.2	Configuration.GetAttribute	Attribute	String	ChVal.*X*.Press	Returns the engineering units value of the pressure of channel *X*. *X*=1-16 for each channel	1 parameter (Float) response: ChVal.*X*.Press: *<chan pressure value in psi>* Fault code '2', if channel number is invalid.
12.2.2	Configuration.GetAttribute	Attribute	String	ChVal.*X*.Temp	Returns the engineering units value of the temperature of channel *X*.	1 parameter (Float) response: ChVal.*X*.Temp: *<chan temperature value in degC>*

					X=1-16 for each channel	Fault code '2', if channel number is invalid.
12.2.2	Configuration.GetAttribute	Attribute	String	DeviceTemp	Returns the internal device temperature for health monitoring purposes.	1 parameter (String) response: DeviceTemp: <temp> degC
Not in Iss. 2, Draft 4 *(expected in final 2.1 release)*	Channel.EUTare	Channel	Byte		Performs a Rezero operation on all channels, aligning the current value to 0 engineering units. Note: The generated rezero offsets are volatile.	Fault code '2', if channel number is invalid.
13.1.1	Pressure.Purge	Time	Byte	<function time>	Switches the attached NetScanner to Purge Mode. *<function time>* is the time in seconds (1 to 255) to run the actual purge part of the routine. Note that where 1 (or a low number) is entered, the routine will actuate the purge valve and return in the minimum time possible. It is recommended this number is not set below 10 seconds	Code '0' response occurs on reception of the command, not after command action is complete
Not in Iss. 2, Draft 4	Pressure.Valve	Function	String	<action>	Performs a valve function. *<action>* is one of the following: Tare = Performs identical function to Channel.EUTare ValveTare = As Tare above but actuating the internal valve to common all ports to the reference line. This also sets a quality flag to indicate the data is bad for the duration of the command process. ValveRun = Moves the internal valve to Run mode. ValveCal = Moves the internal valve to Cal mode. This also sets a quality flag to indicate the data is bad. State = Returns the position of the internal valve Note this function cannot be actioned if in Operating Mode.	Fault Code '2' is issued where <Function> is not specified correctly. Response for the <action> = State is: 1 parameter (String) response: State: <state> where <state> = 'Run', 'Cal', 'Leak', 'Purge' or 'Err'. Fault Code '5' is issued if current state is Operating, when <action> = State