

Tempest TEMP Protocol

Developer Guide

Revision 0.06

May 2017

© Tempest Lighting, Inc.





1 CONTENTS

1	CONTENTS.....	ii
2	Document Revisions.....	iii
3	Introduction.....	1
4	Terms and Conventions.....	1
4.1	TEMP Controller.....	1
4.2	TEMP Responder.....	1
5	Connectivity.....	1
5.1	Physical Connection.....	1
5.2	DEC Configuration.....	2
6	Responder Addressing.....	2
7	Command Structure.....	3
7.1	Get Command Example.....	3
7.2	Get Response Example.....	3
7.3	Set Command Example.....	3
7.4	Set Response Example.....	3
7.5	Order of Precedence.....	3
7.6	Type Identifiers.....	3
7.7	Reserved Characters.....	4
8	Packet Parameters.....	4
8.1	Command and Response PIDs.....	5
8.1.1	STA.....	5
8.1.2	CFG.....	6
8.1.3	VER.....	7
8.1.4	RLY.....	7
8.1.5	LBL.....	8
9	Error Codes.....	9
9.1	SET Response.....	9

2 Document Revisions

Rev	Details	Date
0.01	First draft version for review	30 APR 2015
0.02	Second draft for review and comment	01 MAY 2015
0.03	First revision after initial implementation. Change of Baud rate for 250K to 115.2K	12 MAY 2015
0.04	Inclusion of Error Codes, corrections from internal review.	12 MAY 2015
0.05	Revision of example to match valid mnemonics. Addition of Relay status for initialisation. Addition of status parameters for DEC 4	01 OCT 2016
0.06	Correction if inconsistent mnemonics for Heater and Fan Drive Levels	13 MAY 2017

3 Introduction

This document describes the protocol and command set for the remote control and status monitoring of enabled Tempest DEC3 controllers.

The protocol is a half duplex ASCII syntax, based on a client-server model.

4 Terms and Conventions

All commands are initiated by a TEMP Controller and are responded to by a TEMP Responder.

Commands and Responses are collectively referred to as packets.

A single Command and associated Response is referred to as an exchange.

4.1 TEMP Controller

A TEMP controller is not to be confused with a DEC or 'DEC Controller'. The TEMP Controller, referred to here on in as a Controller will be the system of software that is a client of the Tempest DEC3s.

All serial packets sent by the Controller will be referred to as a command.

4.2 TEMP Responder

A TEMP Responder, for the purpose of this document, refers to a Tempest DEC3 and from here on, is referred to as a Responder.

All serial packets sent by a Responder will be referred to as a response.

5 Connectivity

5.1 Physical Connection

The connection to the Tempest Equipment is via an RS485 Buss with Controller Idle State bias, compliant with ANSI E1.20 section 2.4.1. at 115.2K Baud, 8 data bits, two stop bits.

5.2 DEC Configuration

For a DEC to be operable with TEMP, the DEC must be configured to be in 'BASIC' mode. For further details on configuring a DEC, refer to the accompanying user manual.

6 Responder Addressing

The responder address will be the electronic UID of the responder. A responder UID is comprised of a PLASA manufacturer ID and a unique manufacturer serial number. For Tempest equipment, the vendor ID or VID is \$544C. The remaining 8 characters are a hexadecimal extrapolation of the responder serial number, so for example, a serial number of 64035 would be represented by a hexadecimal value of \$0000FA23. These two strings combined together form the responder address \$544C0000FA23.

The responder serial number can be found on the Tempest electronics or by accessing the user display and scrolling the status information.

A GET command to address \$FFFFFFFF will solicit a response from all responders, which having more than one responder on a single buss will cause a data conflict.

A SET command to address \$FFFFFFFF will be acted on by a responder but not responded to.

7 Command Structure

Each packet is transmitted as an ASCII string, enclosed in the reserved characters '<' and '>'.

Any traffic before the < and after the > characters shall be disregarded.

On receipt of a < character before the final > character, all preceding data shall be disregarded.

Each packet is addressed to an associated Responder.

Each packet contains a reserved Command or Response Identifier character, being '?' or '!' respectively.

The responder address shall be sent immediately after the command or response Identifier.

There are two classes of exchange, **SET** to configure and **GET** to retrieve status data.

7.1 Get Command Example.

```
<?:$544C0000FA23;GET:STA>
```

7.2 Get Response Example

```
<!:$544C0000FA23;STA:ACV=#223,ACC=#5.3,ATM=#32,RHM=#62,RLY=AUT>
```

7.3 Set Command Example

```
<?:$544C0000FA23;SET:RLY=OFF>
```

7.4 Set Response Example

```
<!:$544C0000FA23;RLY=OFF>Packet Parsing
```

7.5 Order of Precedence

Packets are syntactically arranged for parsing in order of precedence, the precedence being a string literal, semi-colon, colon and comma.

7.6 Type Identifiers

- Decimal values are preceded by a hash character #
- Hexadecimal values are preceded by a dollar character \$
- Text strings are enclosed by the double quote character "
- All other data implicitly represents and enumerated constant

7.7 Reserved Characters

The following character are reserved for packet delineation and text encapsulation:

< > " \

Reserved characters used in strings will be preceded by the reserved escape character \ .

Character	Use	Description
\	\\	Backslash
<	\<	Start of packet
>	\>	Packet terminator
"	\"	String Literal Quote

8 Packet Parameters

Each Command packet will contain one parameter, optionally with an argument. Each Response packet will contain one or more parameters each with a parameter argument. Parameters with arguments will be in the format PPP=A, where PPP is the three letter Mnemonic or PID to identify the parameter and A being the argument value or data of the parameter. Parameters without arguments will be in the format PPP, where PPP is the three letter Mnemonic or PID to identify the parameter.

Get command packets may have one parameter argument.

Set command packets will have one parameter argument.

Response packet parameters are cardinal with no pre-determination of the order.

8.1 Command and Response PIDs

8.1.1 STA

8.1.1.1 GET Command

This is used to return the current status of the DEC, no parameter arguments are sent with this request.

8.1.1.2 GET Response

Responses to the request include the following values:

- ACC AC RMS Lamp current
- ACV AC RMS Line Voltage
- ATE Air Temperature (External) °C
- ATM Air temperature (Main) °C
- DLH Heater Drive Level %
- DLF Fan Drive Level %
- FS1 Fan Speed 1 RPM
- FS2 Fan Speed 2 RPM
- FS3 Fan Speed 3 RPM
- FS4 Fan Speed 4 RPM
- LHR Lamp hours
- PCT PCB temperature °C
- RHM Relative Humidity (Main) %
- RLY Relay Status (INI | OFF | SBY | RUN | TRP)

8.1.2 CFG

8.1.2.1 GET Command

This is used to return the current configuration of the DEC, no parameter arguments are sent with this request.

8.1.2.2 GET Response

Responses to the request include the following values:

- LOT Lower Temperature Setting °C
- UPT Upper Temperature Setting °C
- TRP Trip Out Temperature Setting °C
- RHM Relative Humidity Maximum Setting %
- RUN Lamp Current 'On' State Threshold AC Amps RMS

8.1.3 VER

8.1.3.1 GET Command

This is used to return version information associated with the DEC system, no parameter arguments are sent with this request.

8.1.3.2 GET Response

Responses to the request include the following values:

- HWR Hardware Revision Text Description
- FWR Firmware Revision Text Description

8.1.4 RLY

8.1.4.1 SET Command

This is used to change the operational mode of the relay in a DEC. valid arguments for this PID are:

- AUT Automatic On, Relay Closed Unless Over Temperature.
- OFF Manual Off, Relay Held Open

8.1.4.2 SET Response

Response to this parameter will be one of the following enumerated constants:

- INI Relay Open, System Initialisation
- OFF Manual Off state
- SBY Relay Closed, Lamp Load Off
- RUN Relay Closed, Lamp Load On
- TRP Relay Automatically Tripped Open, Over Temperature

8.1.5 LBL

Labels are 7 bit ASCII strings up to 32 characters long, enclosed in "" literal quotes. Reserved characters are escaped as detailed in section [Reserved Characters](#)

8.1.5.1 GET Command

This retrieves the current label string assigned to the DEC, no parameter arguments are sent with this request.

8.1.5.2 GET Response

- LBL Quote Enclosed DEC Label

8.1.5.3 SET Command

This writes a new label to the responder, the new label is sent as a parameter argument, formatted as a quote enclosed string.

8.1.5.4 SET Response

- LBL The Request String Echoed Back

9 Error Codes

In the case that a responder is not able to parse a Command packet, one of the following parameter and argument will be returned:

9.1 SET Response

- ERRHex Error Code Number

The table below describes the possible error codes

Code	Description
\$01	Packet Format Error
\$02	Unknown Command Class
\$03	Unknown Command Parameter
\$04	Parameter Argument Data Invalid
\$FF	Un-handled Error