

# **RS-485 Transducer Digital Communication**

## **User's Manual**

**Firmware Version 218606C**

Stellar Technology Incorporated

# Copyright Notice

© 2004 Stellar Technology Incorporated. All rights reserved.

This manual, as well as the software described in it, is furnished under license and may be used or copied only in accordance with the terms of such license. The content of this manual is furnished for informational use only, is subject to change without notice, and should not be construed as a commitment by Stellar Technology Incorporated. Stellar Technology Incorporated assumes no responsibility or liability for any errors or inaccuracies that may appear in this book. Except as permitted by such license, no part of this publication may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, electronic, mechanical, recording, or otherwise, without the prior written permission of Stellar Technology Incorporated.

it2001 is a trademarks of Stellar Technology Incorporated. it2001 logo and the STI logo are trademarks of Stellar Technology Incorporated in the USA. Microsoft, Windows 9x/NT/2001/XP/CE, Microsoft Access, and Microsoft Excel are registered trademarks of Microsoft Corporation. All other products or name brands are the trademarks of their respective holders.

Part number: 221890 Rev A

# Contents

<b>Chapter 1: Installation</b> .....	<b>5</b>
System Connections.....	6
Communication Setup.....	7
<b>Chapter 2: Quick-Start Tutorial</b> .....	<b>9</b>
Basic Measurements .....	10
<b>Chapter 3: Using Commands</b> .....	<b>11</b>
Command and Query Structure .....	12
Command Entry.....	13
Argument Types.....	14
Command Usage Rules .....	16
Syntax Diagrams .....	16
Communication Timing Specifications.....	17
<b>Chapter 4: Command Reference</b> .....	<b>19</b>
Measure Subsystem .....	20
MEAS:PRES.....	20
MEAS:TEMP .....	20
MEAS:ALL .....	21
Instrument Subsystem.....	22
INST:SEL.....	22
INST:STAT .....	22
Test Commands.....	23
TEST:INP .....	23
System Commands .....	24
*IDN .....	24
*RST.....	24
OFFSET:SET .....	24
SPAN:SET.....	25
<b>Index</b> .....	<b>27</b>



# CHAPTER 1

# Installation

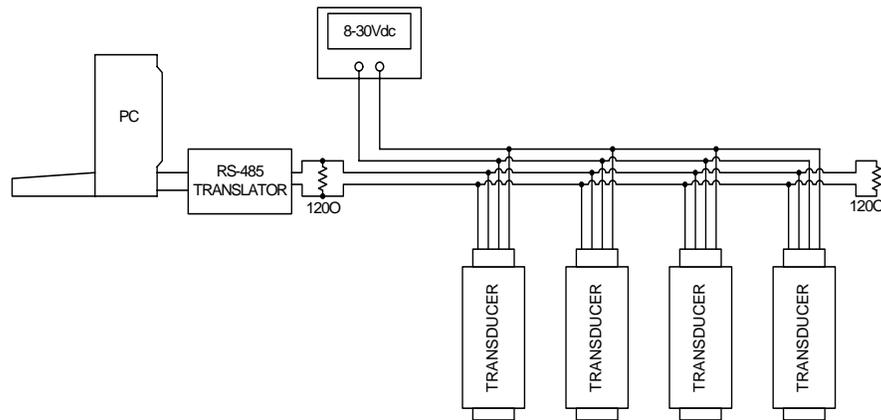
This chapter describes the steps to install the RS-485 Transducer.

## Contents

- System Connections .....5
- Communication Setup .....6

## System Connections

The RS-485 provides network addressing for up to 256 nodes. The following diagram illustrates the network of 4 RS-485 transducers.



## Communication Setup

The RS-485 Transducer uses an RS-485 interface as a means of communication with a controller. The default communication parameters are as follows:

Setting	Value
Baud rate	9600
Data bits	8
Stop bits	1
Parity	none
Flow control	none

**Table 1-1: RS-485 communication settings**

Commands are sent in ASCII text using the syntax described Chapter 2, “Using Commands”. All commands must be terminated with a *carriage-return / line-feed* combination or a *line-feed-only* character sequence.



## CHAPTER 2

# Quick-Start Tutorial

This chapter gives you a quick guide on how to connect RS-485 Transducer and start using it's basic functionality.

### Contents

- Basic Measurements .....9

## Basic Measurements

To make basic digital pressure or temperature measurements you may use the standard Windows Terminal program with appropriate settings.

Steps required to make one digital pressure measurement using the Terminal program:

1. Run Terminal program
2. Set communication port settings
3. Type in the command: meas:pres? terminated with the command terminator (see Chapter 3, "Using Commands")
4. An example of a RS-485 Transducer response is shown below.

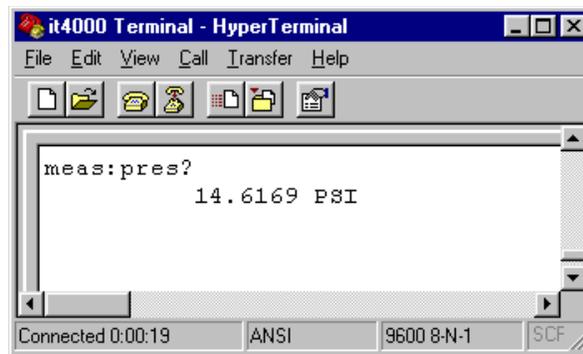


Figure 2-1: Windows Terminal program

For more commands see Chapter 4, "Command Reference".

# Using Commands

The RS-485 Transducer is controlled through the RS-485 interface using a large group of commands and queries. This chapter describes the syntax these commands and queries use and the conventions the transducer uses to process them. The commands and queries themselves are listed in Chapter 4, “Command Reference”.

## Contents

- Command and Query Structure .....12
- Command Entry .....13
- Argument Types .....14
- Command Usage Rules .....15
- Syntax Diagrams .....16
- Communication Timing Specifications .....17

Commands are transmitted to the transducer using the American Standard Code for Information Interchange (ASCII) character encoding.

This manual uses Backus-Naur Form (BNF) notation and syntax diagrams to describe commands and queries. The following BNF symbols:

Symbol	Meaning
< >	Defined element
::=	Is defined as
	Exclusive OR
{ }	Group; one element is required
[ ]	Optional; can be omitted
...	Previous element(s) may be repeated
( )	Comment

Table 3-1: BNF Symbols and Meanings

## Command and Query Structure

Commands consist of set commands and query commands (usually simply called commands and queries). Commands modify instrument settings or tell the transducer to perform a specific action. Queries cause the transducer to return data and information about its status.

Most commands have both a set form and a query form. The query form of the command is the same as the set form but with a question mark at the end. For example, the set command `INPut : GAIN` has a query form `INPut : GAIN?`. Not all commands have both a set and a query form; some commands are set only and some are query only.

A command message is a command or query name, followed by any information the transducer needs to execute the command or query. Command messages consist of five different element types, defined as follows:

Symbol	Meaning
<Header>	The basic command name. If the header ends with a question mark, the command is a query. The header may begin with a colon (:) character; if the command is concatenated with other commands the beginning colon is required. The beginning colon can never be used with a command beginning with an asterisk (*).
<Mnemonic>	A header sub-function. Some commands headers have only one mnemonic. If a command header has multiple mnemonics, they are always separated from each other by a colon (:) character.

Symbol	Meaning
<Argument>	A quantity, quality, restriction, or limit associated with the header. Not all commands have multiple arguments. Arguments are separated from the header by a <Space>. Arguments are separated from each other by a <Comma>.
<Comma>	A single comma between arguments of multiple-argument commands. It may optionally have white space characters before and after the comma.
<Space>	A white space character between command header and argument. It may optionally consist of multiple white space characters.

Table 3-2: Command Message Elements

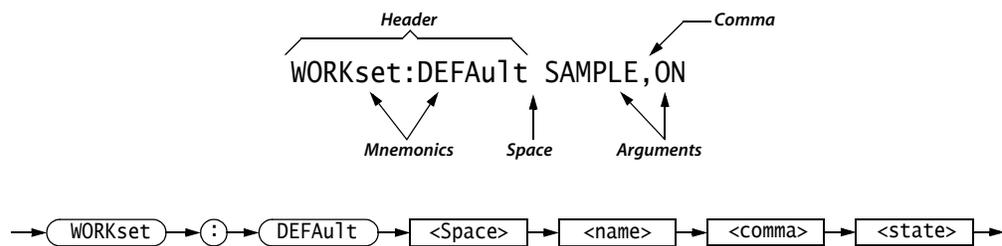


Figure 3-1: Command Message Elements

## Commands

Commands cause the transducer to perform a specific function or change one of its settings. Commands have the structure:

```
<Header> [<Space><Argument> [<Comma><Argument>] . . . ]
```

A command header is made up of one or more mnemonics arranged in a hierarchical or tree structure. The first mnemonic is the base or root of the tree and each subsequent mnemonic is a level or branch off the previous one. Commands at a higher level in the tree may affect those at a lower level. The leading colon (:) always returns you to the base of the command tree.

## Queries

Queries cause the transducer to return information about its status or settings. Queries have the structure:

```
<Header>?
<Header>? [<Space><Argument> [<Comma><Argument>] . . . ]
```

## Command Entry

Follow these general rules when entering commands:

- Commands can be entered in upper or lower case.

- Any command can be preceded with white space characters. White space characters include any combination of the ASCII control characters 00 through 09 and 0B through 20 hexadecimal (0 through 9 and 11 through 32 decimal).
- The transducer ignores commands consisting of any combination of white space characters and line feeds.

## Suffixes

Some mnemonics have a plural form. The mnemonic that is expressed in plural form indicates that it represents more than one instance of a subsystem. This is illustrated as follows:

TEST: INP5?

All suffixes have a default value of one and is used when the suffix is not specified. Suffixes are enclosed in brackets in the command syntax descriptions to indicate their optional inclusion. The brackets are not to be included in actual usage.

## Command Termination

All commands are terminated with a *carriage-return / line-feed* combination. The SCPI specification also allows for the use of a *line-feed* character only. The ASCII codes for these command terminators are as follows:

ASCII Code	Key Stroke	Description
10	Control + j	Line feed
13	Control + m	Carriage return

Table 3-3: ASCII Codes for Command Termination

## Argument Usage

All arguments listed for a command are mandatory and must be specified by at least a placeholder. Multiple arguments must be separated by a comma. String arguments are case-sensitive unless otherwise noted. Those arguments that are actually mnemonics themselves follow the same abbreviation options as described above.

## Argument Types

The argument of a command may be in one of several forms. The individual descriptions of each command tell which argument types to use with that command.

### Numeric Arguments

Many transducer commands require numerical arguments. The syntax shows the format that the transducer returns in response to a query. This is also the preferred format when sending the command to the transducer though it will accept any of the formats. This manual represents these arguments as follows:

Symbol	Meaning
<NR1>	Signed integer value
<NR2>	Floating point value without an exponent
<NR3>	Floating point value with an exponent

Table 3-4: Numeric Argument Types

The transducer will automatically force most numeric arguments to a valid setting, either by rounding or truncating, when you input an invalid number unless otherwise noted in the command description.

### Quoted String Arguments

Some commands accept or return data in the form of a quoted string, which is simply a group of ASCII characters enclosed by a double quote ("). For example: "this is a quoted string"

Symbol	Meaning
<QString>	Quoted string value

Table 3-5: Quoted String Argument Type

Follow these rules when you use quoted strings:

- A quoted string can include any character defined in the 7-bit ASCII character set.
- Strings can have upper or lower case characters.
- A string cannot be terminated with the END message before the closing delimiter.
- The maximum length of a quoted string returned from a query is 256 characters.

### Block Arguments

Some transducer commands use a block argument form:

Symbol	Meaning
<NZDig>	a non-zero digit character, in the range 1-9
<Dig>	A digit character, in the range 0-9
<DChar>	A character with the hex equivalent of 00 through FF hexadecimal (0 through 255 decimal)
<Block>	A block of data bytes, defined as: <Block> ::= { #<NZDig><Dig>[<Dig>...][<DChar>...]   #0[<DChar>...]<terminator> }

Table 3-6: Block Argument Types

<NZDig> specifies the number of <Dig> elements that follow. Taken together, the <Dig> elements form a decimal integer that specifies how many <DChar> elements follow.

## Command Usage Rules

It is important to keep the following rules in mind when using the commands in this reference guide:

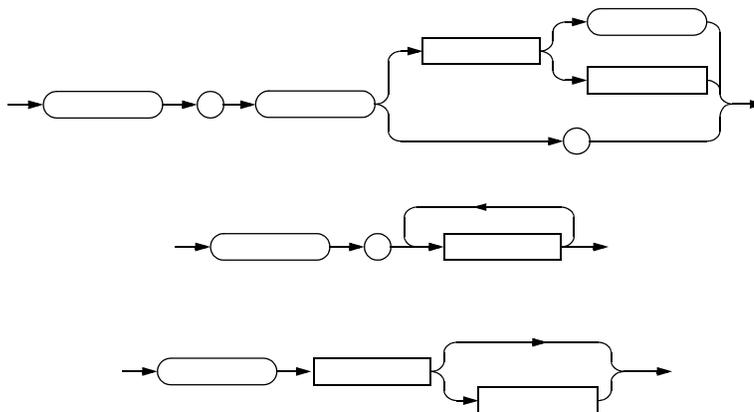
1. Commands are case-insensitive.
2. All commands are terminated by a *carriage-return / line-feed* combination or a *linefeed*.
3. All arguments are required.
4. Multiple arguments must be separated by a comma.
5. String arguments are case-sensitive unless they are a mnemonic.

## Syntax Diagrams

The syntax diagrams in this manual use the following symbols and notation:

- Circles and ovals contain literal elements. Most elements must be sent exactly as shown. The diagrams show command mnemonics in both upper and lower case to distinguish between complete and abbreviated spellings. These elements are not case sensitive and you can omit the lower case portion of the mnemonic.
- Boxes contain the defined elements described earlier in this section, such as <NR3> or <QString>.
- Elements are connected by arrows that show the allowed paths through the diagram, and thus the orders in which you can send the elements. Parallel paths show that you must take one and only one of the paths. A path around a group of elements shows that those elements are optional. Loops show elements that can be repeated.

Here are some examples of typical syntax diagrams:

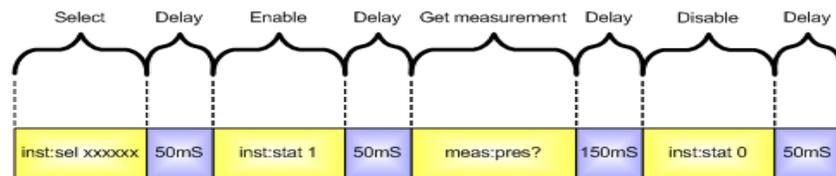


## Communication Timing Specifications

To ensure error-free communication with a network of RS-485 Transducer devices the following timing specifications must be taken into consideration:

1. Allow at least 50 mS between commands which don't return a value;
2. Allow at least 150 mS after query commands.

Refer to Chapter 5 for a complete listing of commands.





# Command Reference

This chapter describes each of the commands used to configure and control the RS-485 Transducer pressure transducer. The command reference is broken down into several groups of related functionality.

## Contents

- Measure Subsystem .....20
- Instrument Subsystem .....22
- Test Commands .....23
- System Commands .....24

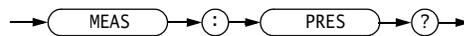
## Measure Subsystem

The measure subsystem includes commands for initiating pressure and temperature measurements.

### MEAS: PRES

Returns a pressure measurement.

**Syntax** MEAS: PRES?



**Remarks** The unit of measure is PSI.

**Example** meas:pres?  
14.1340

### MEAS: TEMP

Returns a temperature measurement. This is the temperature of the pressure sensing element which approximates that of the medium.

**Syntax** MEAS: TEMP [*channel*]?



**Suffix** *channel*

Parameter	Result
none	Temperature from the on-chip temperature sensor (default)
0	Temperature from the on-chip temperature sensor
1	Temperature from the RTD (optional)

**Remarks** The unit of measure is degrees Fahrenheit.

**Example** meas:temp?  
78.0910

## MEAS:ALL

Returns a pressure and temperature measurements.

**Syntax** MEAS:ALL?



**Remarks** First value - pressure measurement, second value - on-chip temperature measurement, third value - RTD temperature if there is one present.

**Example** meas:all?  
78.5000,123.2430

## Instrument Subsystem

The instrument subsystem includes commands for selecting and activating a device on an RS-485 network.

### INST:SEL

Selects an instrument.

**Syntax** INST:SEL *instrument*



**Parameters** *instrument*  
Six digit serial number.

**Remarks** The instrument's state may be changed only if the instrument is selected.

**Example** inst:sel 123456

### INST:STAT

Changes the state of the selected instrument.

**Syntax** INST:STAT *state*



**Parameters** *state*  
1 - on  
0 - off.

**Remarks** All communications on the RS-485 network will be addressed only to the instrument with the state set to 1 (on).

**Example** inst:stat 1

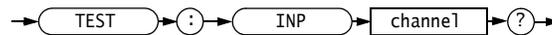
## Test Commands

Test commands allow to calibrate digital output of the RS-485 Transducer.

### TEST:INP

Reads digital counts from selected channels.

**Syntax** TEST:INP [*channel*]?



**Suffix** *channel*  
Possible values: 5

Parameter	Result
5	Digital pressure and temperature counts, Board temperature

**Example** test:inp5?  
11775507,49985,67.332

## System Commands

System commands includes those relating to identification and resetting the unit.

### \*IDN

Returns the transducer part number, serial number and a revision.

**Syntax** \*IDN?



**Example** \*idn?

STELLAR TECHNOLOGY INC,IT2001-15A-101,007713,0

### \*RST

Resets the it2001 to power-up status. All parameters return to their default state.

**Syntax** \*RST



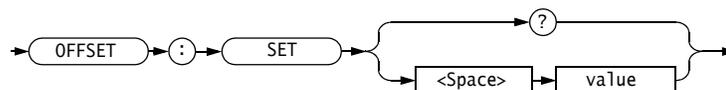
**Remarks** Executing this command is equivalent to executing a power-up sequence.

**Example** \*rst

## OFFSET:SET

Sets or returns the value for the input offset.

**Syntax** OFFSET:SET *offset*  
OFFSET:SET?



**Parameters** *offset*

Real numeric value.

This signed value will be added to the digital output prior to displaying it.  
Input offset will also affect the analog output.

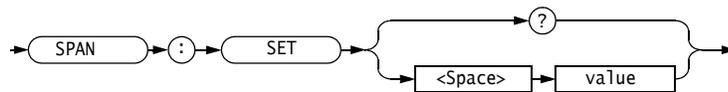
**Remarks** Input offset value is always in PSI.

**Example** `offset:set 3.4`  
`offset:set?`  
3.40

**SPAN:SET**

Sets or returns the value for the span.

**Syntax** `SPAN:SET span`  
`SPAN:SET?`

**Parameters** *span*

Real numeric value from (0..150] interval.

The pressure transducer's span will be set to the span% of the original value.

**Remarks** Default value is 100.

**Example** `span:set 50`  
`span:set?`  
50.000



# Index

## Symbols

\* (asterisk) 12  
: (colon) 12, 13

## A

analog output 25  
argument 13, 14, 15  
Argument Types 14  
    Block Arguments 15  
    Numeric Arguments 14  
    Quoted String Arguments 15  
ASCII (American Standard Code for Information Interchange) 7, 12, 14, 15

## B

Backus-Naur Form (BNF) 12  
Basic Measurements 10

## C

carriage-return 7, 14, 16  
case  
    lower 13, 15, 16  
    upper 13, 15, 16  
channel 23  
Command Entry 13  
    Argument Usage 14  
    Command Termination 14  
    Suffixes 14  
Command Reference 19  
    Input Subsystem  
        INPut  
        OFFSet  
        VALUE 24  
    Measure Subsystem 20  
        MEASure  
            ALL 21  
            PRESsure 20  
            TEMPerature 20  
        UNIT  
            PRESsure 21  
    System Commands 24  
        \*IDN 24  
        \*RST 24  
    Test Commands 23  
        TEST  
            INPut 23  
Command Usage Rules 16  
Commands and Query Structure 12  
    Commands 13

    Queries 13  
    communication port 10  
Communication Setup 7

## D

digital  
    count 23  
    output 23, 25  
    pressure 10  
    temperature 10

## G

gain 12

## L

line-feed 7, 14, 16

## M

message 12, 15  
mnemonic 12, 13, 14, 16

## O

offset  
    input 24, 25

## R

RS-232 7, 11

## S

Standard Commands for Programmable Instruments (SCPI) 14  
    suffix 23  
Syntax Diagrams 16

## W

white space 13, 14

