OPERATION MANUAL
HC-485 and GC-485 LVDTs
With digital output

Contents

1. Introduction

The HC-485 and GC-485 Series are heavy-duty LVDT type transducers. The GC-485 is spring loaded (Gage Head). These transducers are self-contained, ultra-precision, digital I/O devices for high-performance position/displacement measurements in environments containing moisture, dirt, and fluid contaminants. The HC-485 and GC-485 eliminate the need for expensive and error-prone analog to digital conversion by internally converting the analog LVDT signals into accurate engineering units (imperial or metric). The result is a fully calibrated and traceable measurement device, ready for installation, and 100% field interchangeable.

Position is normally given as a positive or negative number from a null position. The calibrated range of the transducer is stored internally, in non-volatile memory. If the position moves outside this range, the reading becomes non-linear, and eventually reaches hard limits. Although this can easily be calculated externally, the transducer Status register also provides under and over range indication.

For complete specifications and ordering information, please refer to the datasheet by using the following links.

HC-485: http://www.te.com/usa-en/product-CAT-LVDT0038.html

GC-485: http://www.te.com/usa-en/product-CAT-LVDT0050.html

2. Zero (Tare)

Normally the HC/GC-485 transducers give a zero reading with the core at the electrical center of the coil, (null position), reading positive as the HC-485 core is displaced toward the connector end (or GC-485 probe inserted) and negative when displaced toward the other end.

The Zero (also referred to as Tare) function allows an alternate zero reference to be used. This is particularly useful when an initial setup of the equipment is required to remove linkage or machine variability. The zero function may also be used when a unipolar output is desired, by performing a zero at end of the stroke.

Activating the zero function establishes an offset zero reference at the current position. A position measurement made immediately after this, assuming position was not changed, will indicate zero (within system resolution).

The Zero function can be removed at any time by writing 0 to the Zero register.

The Zero function can be made semi-permanent by saving the setup to non-volatile memory.

3. Filter

The transducer contains advanced digital filtering that provides optimum filtering for position calculations. The default is set to provide 200Hz+ bandwidth with a rapid roll off, at higher frequencies.

In certain situations, small, low frequency changes in indicated position or bit dithering may occur. This noise may obscure the intended measurement. To address this problem, other filtering options can be selected by adjusting the Filter value.

The use of this filter does not introduce any distortion or aliasing effects, and does not affect sampling rate or communications speed.

Increasing the filter value increases the filtering. A value of one indicates that the lowest filtering is employed (full bandwidth). In general terms the filter can be thought of as an additional first order filter whose time constant is (n-1) x Sample rate.

N	Time Constant	Equivalent analog frequency
1	-	200Hz+
2	1.5mS	100Hz
3	3mS	50Hz
100	150mS	1Hz

The filter count can be saved to non-volatile memory.

4. Units

The output of the transducer is always a single precision, floating point value in real position unit. The unit may be changed using the supplied setup utility or by programming to:

- meters
- centimeters
- millimeter
- inches
- thousandths of an inch (mils)
- micro-inches

This selection is simply a mathematic scaling of the position measurement. It does not affect calibration, accuracy, resolution, or any other performance parameters.

5. Minimum, Maximum, and Total Indicated Runout (TIR)

Some applications require recording minimum and maximum position readings throughout a measurement cycle, without continuously polling the transducer.

Minimum and maximum position values are computed and updated at 650 samples per second; they are available for reading upon request. The values may be reset to the current position using the Reset register.

Total Indicated Runout (TIR) is the arithmetic difference between min and max. It is often used to indicate runout on an eccentric shaft or other round parts.

6. Reset

When activated, the Reset register immediately clears the Min, Max and TIR values.

7. Velocity

Velocity is computed on a sample-by-sample basis and is calculated as the change in position, divided by the change in time. The active position units and filtering are both applied before velocity calculation.

Seeing extraordinary velocity figures when the system is stationary or hardly moving sometimes confuses people. However, the value is calculated over a very short period of time. Even a tiny change in position in 1/650th of a second has a high velocity. Filtering of the position signal alleviates this “noise” to a degree but it cannot remove the noise due to quantization error (about 1/40000).

8. User ID

Four registers, 64 bits total, are available to the user. There is no restriction on what can be stored. This might be dates, serial numbers or location information to support configuration or traceability. These values may ONLY be changed using the HC/GC-485 configuration software.

9. Status

The Status register gives basic information on transducer operation. This includes the state of the Zero function as well as over and under range information. The format of the Status register is as follows:

Bit	Clear (0)	Set (1)	Bit	Clear (0)	Set (1)
0	No communication timeout	Communication timeout	8	Unused	
1	Unused		9	Unused	
2	No parity error	Parity error	10	Even parity	Odd parity
3	No under range	Under range	11	Parity disabled	Parity enabled
4	No over range	Over range	12	Communication echo disabled	Communication echo enabled
5	Unused		13	Modbus ASCII mode	Modbus RTU mode
6	Defined for factory test only		14	Fixed point output	Floating point output
7	Defined for factory test only		15	Modbus communication mode	iSeries communication mode

10. Changing Transducer Operation

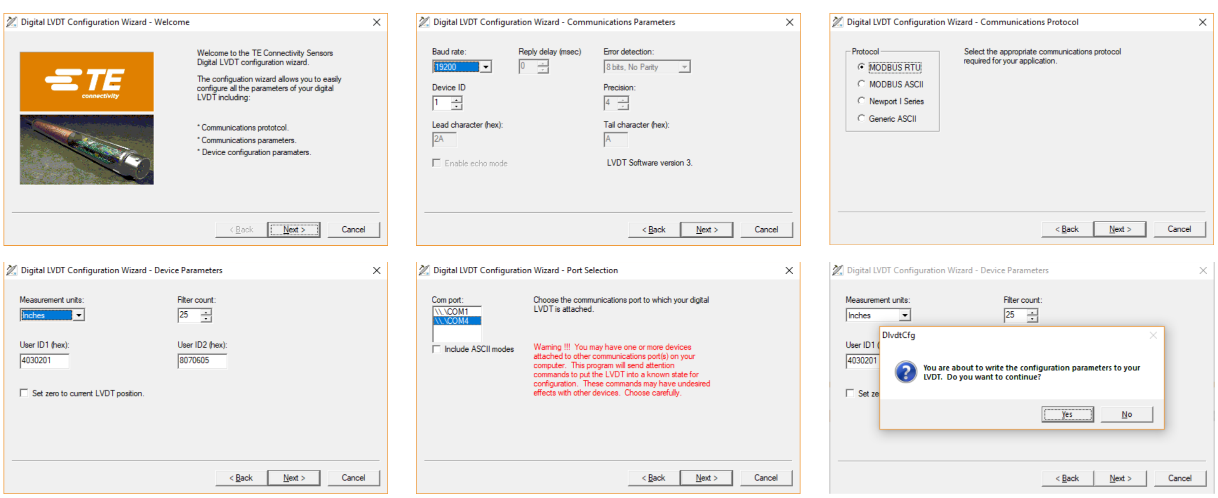
Variables below may be set to change the transducer operation. Some changes take immediate effect and others, mostly associated with communications protocol, will not take effect until saved to internal, nonvolatile memory and power is cycled. These functions are listed below, along with effectivity:

Parameter	Effect
Zero	Immediate
Units	Immediate
Filter	Immediate
Address	Save and restart
Baud rate	Save and restart
Parity	Save and restart
Modbus mode	Save and restart
Communication mode	Save and restart
Communication echo	Save and restart
Lead character	Save and restart
Tail character	Save and restart
Fixed/floating point	Save and restart

All of these values can be read back from the transducer. All of these values may be saved in non-volatile memory. Any values that are not changed will be saved in their original state.

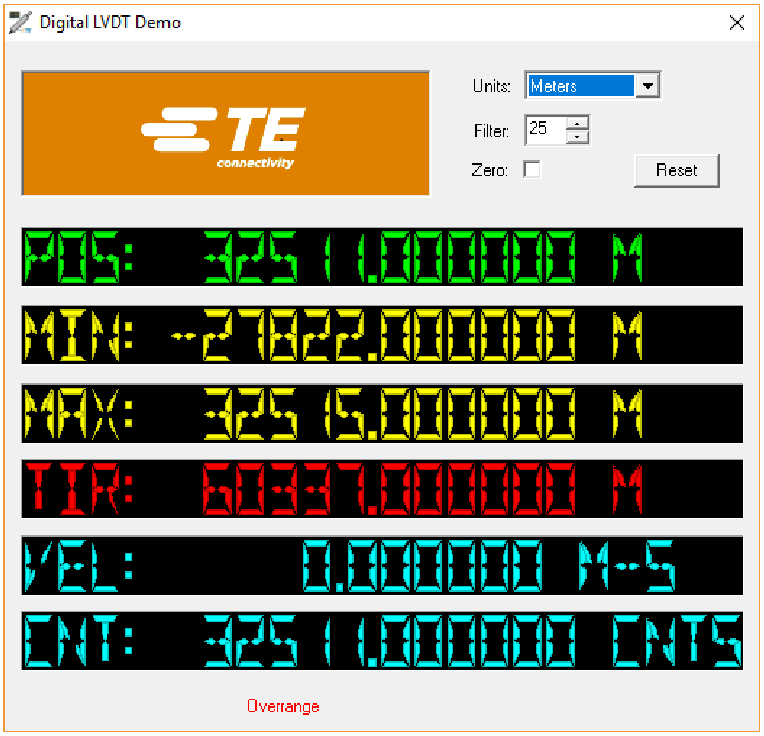
Note: It is possible to key protocol information over the standard protocols, however, it is highly recommended that the supplied configuration utility is used. This software has the ability to recover the transducer from erroneous protocol setup that the user may have inadvertently set.

- *SW-31180100-00 Configuration Utility*



. 2

- *SW-31730100-000* *Demo Software*



Note: The Demo software uses Com Port #1 as default. To use a different Com Port, pass the Com Port # as a parameter to the executable program, i.e. "DlvdtDemo.exe 4", to use Com Port #4.

11. Saving Setup

The Save Setup command is executed by specifically writing to one of the registers. This takes the current transducer setup data and stores it in non-volatile memory. It will take full effect on the next power on or restart.

The recommended sequence for changing the setup is:

- Power on the HC/GC-485
- Change required parameters
- Check everything is set correctly
- Activate “Save Setup”
- Verify it was saved correctly
- Power off and back on again

The supplied setup routine does all of these steps. We recommend to use it for transducer setup since it can also cope with any initial protocol setups.

Data integrity is verified by checking each value stored in EEPROM. In addition, an internal checksum is used to verify data restore. Hardware and software are specifically designed to guard against data loss or corruption.

Note: Min, Max and TIR values are strictly volatile and cannot be saved.

12. Communications

Serial communications of 7 and 8 bits are supported at baud rates up to 19200. Odd, even, mark, and space parity are supported in ASCII modes only. These parameters may be changed using the supplied setup utility or by programming. Programmed changes to communications or protocol settings do not take effect until the data is saved and the unit is restarted. The setup utility program handles this automatically.

13. Device Address

For multi-drop communication, every device must be assigned a different device address. The device address for multidrop communication may be changed from the factory default setting of 1, to any value from 1 to 247.

14. Protocols

Modbus RTU, Modbus ASCII, Generic ASCII, and iSeries ASCII are the standard protocols supported by the transducer. The Generic ASCII format may be used to emulate the Newport Instruments iSeries protocol and connect to RD-4 or RD-6 display modules and the i-Server Ethernet module.

15. MODBUS

Modbus is a widely used protocol for RS-485 and other digital networks. A detailed explanation of Modbus is beyond the scope of this document. The specifications are freely available at: http://www.modbus.org/.

The HC/GC-485 transducers support both the RTU and ASCII modes of the Modbus protocol. The RTU mode uses the full 8 bits of a character to send binary data. The ASCII mode only uses 7 bit printable ASCII characters and hexadecimal coding of binary data. Modbus RTU is roughly twice as fast as Modbus ASCII.

The transducer provides the following Modbus functions:

4 Read Input Registers
6 Preset Single (Holding) Register
8 Diagnostic

Function 4 permits multiple register reads. Multiple register writes are not supported.

Diagnostic sub-functions include:

0 Echo
1 Restart
2 Status
3 Change ASCII input delimiter (Tail)
4 Listen only mode

16. Register Definitions

Register	Read Function (4)	Format	Write Function (6)	Range
0	Position	IEEE Float	Read Only	
2	Minimum	IEEE Float	Read Only	
4	Maximum	IEEE Float	Read Only	
6	Velocity	IEEE Float	Read Only	
8	Runout	IEEE Float	Read Only	
10	Status		Read Only	
12	User ID1	32 bits	Read Only	
14	User ID2	32 bits	Read Only	
32	Unused		Reset	0
33	Unused		Zero (Tare)	0 or 1
34	Filter Count		Filter Count	0-100
35	Units		Units	0-5 (see Table 1)
36	Address		Address	1-247
37	Baud Rate		Baud Rate	0-3 (see Table 2)
38	Precision		Precision	1-8
39	Format		Format	0-254 (see Table 3)
40	Lead Character		Lead Character	0-255
41	Tail Character		Tail Character	0-255
42	Unused		Save Setup	0xAA

Units	Value
Inches	0
Thousandths (mils)	1
Micro-inches	2
Meters	3
Centimeters	4
Millimeters	5

Table 1

Baud Rate	Value
19200	0
9600	1
4800	2
2400	3
38400	4

Table 2

Bit	Set (1)	Clear (0)
5	Odd Parity	Even Parity
4	Parity Enable	Parity Disable
3	Echo Enable	Echo Disable
2	Modbus ASCII	Modbus RTU
1	ASCII Fixed Point	SCII Floating Point
0	Modbus	ASCII - iSeries

Table 3

17. Errors/Exceptions

Errors that may indicate a corrupt message will not generate a response from the transducer. Exception responses will occur for the following reasons:

01 ILLEGAL FUNCTION — The message function is not implemented.
02 LLEGAL DATAADDRESS — The address (register) does not exist.
03 ILLEGAL DATA VALUE — The value written is not allowable.
04 FAILURE IN ASSOCIATED DEVICE — The device has failed.

18. Modbus Data Format

Modbus was designed based on 16 bit long “word” registers. These registers were called holding registers, input registers and coils, reflecting their PLC roots. The transducer uses the definition “holding registers” to relate to the transducer read and write parameters.

Note: Modbus documentation numbers registers from 1, but the message format numbers them from zero. This can be confusing.

The HC/GC-485 transducers have to work with 8, 16 and 32 bit values. All 16 and 8 bit words use normal Modbus 16-bit registers. For 8 bit, the unused bits are set to zero. For 32 bit values (i.e. floating point values), two adjacent registers have to be used. The order of these words is critical to interpreting the number correctly. The lowest address is the LEAST significant.

Floating-point values are encoded according to IEEE754; This is a common format used by many computers and software including Microsoft Visual C++ and Visual Basic.

IEEE 4-byte floating-point format:

Most Significant		Least Significant
1 bit	8 bits	23 bits
Sign	Exponent	Mantissa

The value of the number is: **(-1)S * 2(Exponent-127) * 1.Mantissa**

Zero is represented by 4 bytes of zeros. The precision is approximately 7 decimal places

Note: This works for little endian processors such as the Intel X86 architecture. For big endian processors the most and least significant bytes of each register must be swapped.

19. Generic/iSeries ASCII Protocol

The generic ASCII and iSeries protocols can send transducer data out as plain ASCII numbers. For example, a position of 1.054321 comes out as “1.054321”. While this seems logical, it takes a lot more characters to send this out than in the RTU mode, and it can require a lot more effort to read and process into a computer or a PLC.

However, ASCII can often be used to provide a simple method for display or datalogging setups where the system merely displays or prints the transducer response. Indeed, the Newport Instruments RD4 display can be connected directly to the transducer and display position without requiring any computer or complex programming.

It should be noted that the generic protocols are not designed to be a replacement for the more robust, higher performance Modbus. However, it is possible to achieve the same functionality.

20. Message Format

The message is composed of three parts:

- Lead character
- Body
- Tail character

The lead character has to be recognized as the start of message. It is generally a unique character that is not used anywhere else in the messages (i.e. “*“).

The body of the message contains the device address, function and any data. The first two hexadecimal characters indicate the device address. The next character is the function, followed by any associated data in hexadecimal:

Prrvvvv — Enter or write value to (16 bit) register

“rr” is two hexadecimal characters indicating register number

“vvvv” is four hexadecimal characters indicating value to be written

Grr — Get or read value from (16 bit) register

“rr” is two hexadecimal characters indicating register number

Z02 — Restart

Xrr — Send floating point value as ASCII string

“rr” is two hexadecimal characters indicating register number

The tail character (i.e. <cr>) terminates all messages, which marks the end of message.

Examples:

*02P010034	Write 0034h to register 1 in device address 2
*09G10	Get register 16 in device address 9 as hexadecimal string (e.g. 90CD5F34)
*03Z02	Restart device with device address 3
*02X01	Get register 1 in device address 2 as ASCII string (e.g. 12.345)

21. Message Reply

The response from the transducer may optionally echo back the transmitted command. The echo bit of the format register determines if echo is active. If set to 1, the command message is echoed back from the transducer, followed by the command response:

P For write operations, there is no response.

G The response is hexadecimal representation of 4 bytes.

X Sends data in fixed point (e.g. "0.234") or floating point decimal (e.g. 1.234e1).

Z No response

The tail character (always sent) marks the end of message.

Example:

Get register 16 in device address 9 as hexadecimal string. Echo enabled and register contains 0x90CD5F34. Head and tail characters are respectively set to "*" and <cr>.

- Master Sends: *09G10<cr>
- Transducer Sends: *09G1090CD5F34 <cr>

22. Message Timing

The maximum time between characters of a message to the transducer must not be longer than one second.

Changing Protocol characteristics (Some of the generic protocol message characteristics can be changed to use in other systems):

Lead Character

Changes the character that marks start of message. Can be set to any 7-bit character except 0 (null)

Tail Character

Changes the character that marks start of message. Can be set to any 7-bit character except 0 (null)

Float/Fixed and Precision

These settings affect the "X" command only. The X command outputs a floating-point number in ASCII. The number of digits given is set by the Precision value. The format may be either fixed (e.g. 20.122) or scientific (e.g. 2.0122e-1).

Command Echo On/Off

To echo the command as part of the command response. Should always be set (On) when used with the Newport Instruments iServer, RD4 or RD6 displays.

iSeries

The following settings allow use of the iSeries RD4 display with the transducer.

- Head="*"
- Tail=<cr> (0x0D, carriage return)
- Fixed point
- Precision=4

- Echo On

The RD4 communications parameters, should be set to match the transducer as follows:

- Communications Standard = 485
- Baud Rate = Transducer baud rate
- Data Format = Transducer Parity
- Mode = Host
- Device address = transducer device address
- Interface device - doesn't matter

The display will then show position value in selected unit.

23. Reading and Writing Values

Unlike Modbus, read and write operations for the same variable (such as units) are two DIFFERENT addresses.

All write operations (W commands) are 16 bit hex values. No 32 bit values can be written. Registers are numbered by word address, starting at 1.

All read operations (R commands) are 32 bit values (8 hex digits). All registers addresses are a 32-bit starting at 1.

24. Errors

Errors in the format of the messages sent to the HC/GC-485 will not result in an error message coming back from the transducer. This may seem unusual but this is done to prevent a bad message from causing in a flood of error messages on the RS-485 bus from every device at the same time.

Messages with the correct format but with other errors (i.e. trying to read a register that doesn't exist) will result in an error message.

The iSeries/Generic ASCII protocol uses two error messages from the transducer. These are:

?43	Communications error	Unimplemented function or bad value (write)
?46	Format Error	Bad hex number or incomplete message

These error messages will only be reported if the first part of the message (Head and Address) is valid.

NORTH AMERICA
Measurement Specialties, Inc.,
a TE Connectivity Company
1000 Lucas Way
Hampton, VA 23666
United States
Phone: +1-800-745-8008
Fax: +1-757-766-4297
Email: customercare.hmpt@te.com

EUROPE
MEAS Deutschland GmbH
a TE Connectivity Company
Hauert 13
D-44227 Dortmund
Germany
Phone: +49-(0)231-9740-0
Fax: +49-(0)231-9740-20
Email: customercare.dtmd@te.com

ASIA
Measurement Specialties China Ltd.,
a TE Connectivity Company
No. 26, Langshan Road
High-tech Park (North)
Nanshan District, Shenzhen 518057
China
Phone: +86-755-33305088
Fax: +86-755-33305099
Email: customercare.shzn@te.com