HI302 - smar

FIRST IN FIELDBUS

USER'S MANUAL

HART / FOUNDATION FIELDBUS INTERFACE

JUN / 03

HI302

FOUNDATION

HI302ME

smar

ISO 9001 : 2000 OCS-0006 INMETRO

web: www.smar.com

Specifications and information are subject to change without notice.
For the latest updates, please visit the SMAR website above.

BRAZIL
Smar Equipamentos Ind. Ltda.
Rua Dr. Antonio Furlan Jr., 1028
Sertãozinho SP 14170-480
Tel.: +55 16 3946-3510
Fax: +55 16 3946-3554
e-mail: smarinfo@smar.com

ARGENTINA
Smar Argentina
Soldado de La Independencia, 1259
(1429) Capital Federal – Argentina
Telefax: 00 (5411) 4776 -1300 / 3131
e-mail: smarinfo@smarperifericos.com

CHINA
Smar China Corp.
3 Baishiqiao Road, Suite 30233
Beijing 100873, P.R.C.
Tel.: +86 10 6849-8643
Fax: +86-10-6894-0898
e-mail: info@smar.com.cn

FRANCE
Smar France S. A. R. L.
42, rue du Pavé des Gardes
F-92370 Chaville
Tel.: +33 1 41 15-0220
Fax: +33 1 41 15-0219
e-mail: smar.am@wanadoo.fr

GERMANY
Smar GmbH
Rheingaustrasse 9
55545 Bad Kreuznach
Germany
Tel: + 49 671-794680
Fax: + 49 671-7946829
e-mail: infoservice@smar.de

MEXICO
Smar México
Cerro de las Campanas #3 desp 119
Col. San Andrés Atenco
Tlalnepantla Edo. Del Méx - C.P. 54040
Tel.: +53 78 46 00 al 02
Fax: +53 78 46 03
e-mail: ventas@smar.com

SINGAPORE
Smar Singapore Pte. Ltd.
315 Outram Road
#06-07, Tan Boon Liat Building
Singapore 169074
Tel.: +65 6324-0182
Fax: +65 6324-0183
e-mail: info@smar.com.sg

USA
Smar International Corporation
6001 Stonington Street, Suite 100
Houston, TX 77040
Tel.: +1 713 849-2021
Fax: +1 713 849-2022
e-mail: sales@smar.com

Smar Laboratories Corporation
10960 Millridge North, Suite 107
Houston, TX 77070
Tel.: +1 281 807-1501
Fax: +1 281 807-1506
e-mail: smarlabs@swbell.net

Smar Research Corporation
4250 Veterans Memorial Hwy.
Suite 156
Holbrook , NY 11741
Tel: +1-631-737-3111
Fax: +1-631-737-3892
e-mail: sales@smarresearch.com

Introduction

The HI302 is an equipment integrated to System 302 which main function is to interface HART devices to Foundation Fieldbus Systems, allowing the user to perform maintenance, calibration, sensor status monitoring, device status, among others information. See below some HI302 features:

- Integral Part of System 302;
- Tight integration with different system manufactures due to the use of standard protocols such as Foundation Fieldbus and HART;
- 8 HART master channels;
- Optional Analog Conversion (4-20 mA / Foundation Fieldbus – HI302-I and Foundation Fieldbus / 4-20 mA – HI302-O);
- Totally integrated to AssetView;
- Uniform systems and tools, making it easy to operate and reducing maintenance costs;
- Non-multiplexed and independent HART channels;
- HART Configuration Commands located into the module and possibility to send HART messages through bypass parameters;
- Suitable for Asset Management Systems;
- Complete configuration of Smar devices enclosed in the HI302 module, thus no additional configuration is required.
- Configuration for third party equipments can also be embedded in the Flash memory or added through FF blocks.

Index

Chapter 1

Overview

This user manual contains instructions on how to install and configure the HI302. If the user already knows how to install the FF and HART devices and wants to work with the HI302 immediately, please, go to the chapter 7.

- This manual is compliant to firmware version 3.16 or above and DD 0301 or above. Check the parameter HCFG.FIRMWARE_VERSION.

General Characteristics

Figure 1.1 – HI302-I Module

Among the main characteristics, the following may be mentioned:

- The HI302 supports up to eight point-to-point HART devices or 32 HART devices in the multidrop mode (4 devices per channel);
- 8 HART Master communication ports that can be configured as Primary or Secondary;
- 1 Foundation Fieldbus H1 Channel;
- Fed via backplane (5Vdc @500 mA);
- Device supply should be from an external source;
- Input circuits 4-20 mA on HI302 – I (current conversion to Fieldbus);
- Output circuits 4-20 mA on HI302 – O (Fieldbus conversion to current);

There are three models for the HI302, according to the analog conversion needs:

- HI302 – N: only HART communication;
- HI302 – I : HART communication and conversion of eight 4-20 mA analog inputs to FF;
- HI302 – O: HART communication and FF conversion to eight 4-20 mA analog outputs;

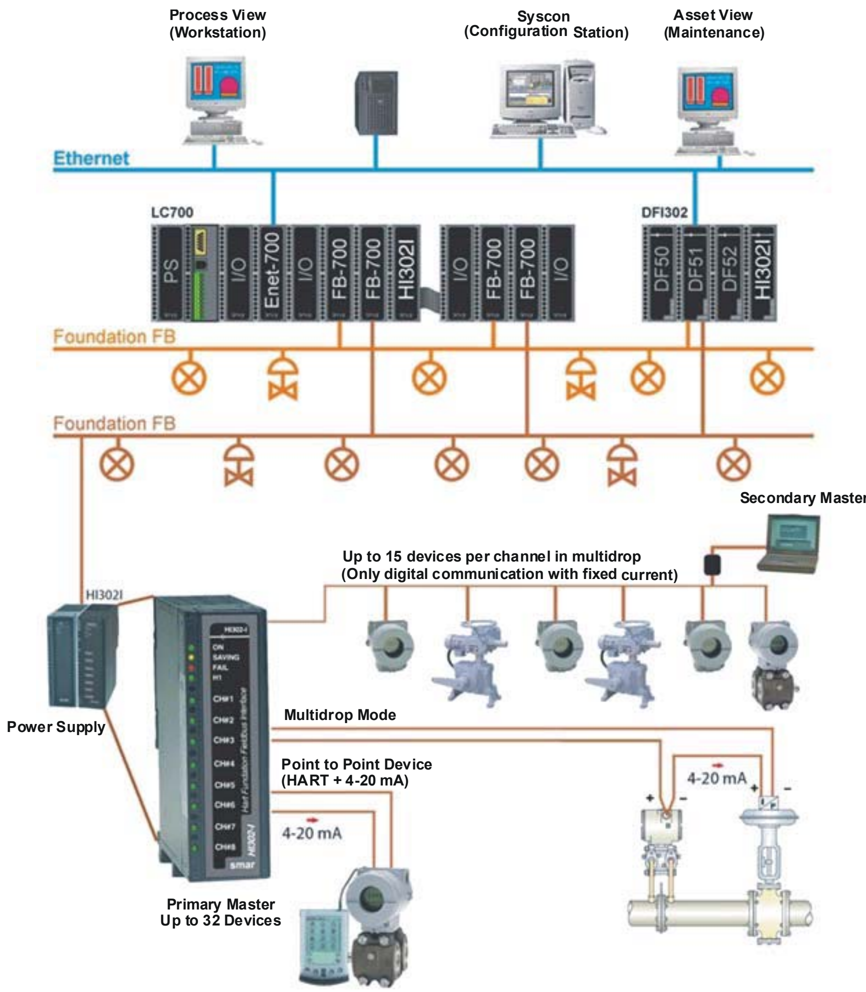

Figure 1.2 – Hart/4-20 mA

Function Blocks

Several blocks were implemented to give the module the required functionality.

HART Transducer Blocks

HCFG (HART Configuration and Diagnostic) – Concentrates general configuration parameters for module operation, in addition to parameters on HART Communication performance and diagnostic. It also concentrates parameters for analog circuit calibration.

HIRT (HART Information and Dynamic Data) – This block contains the main parameters, i.e., the most commonly used, besides dynamic variables. All parameters related to universal commands and some main “Common Practice” commands are found here. There should be one HIRT block for each HART device installed, up to 32 blocks in case of a multidrop connection. In normal operation, the HIRT block parameters show the HART device variables, since there are mechanisms to keep the HI302 database updated. See the Appendix A or the Function Blocks handbook for details. All of the HART dynamic variables should be accessed through this block.

HVT (HART Variable Template) – This block is a large collection of variables for general use. It is now possible to access any HART instrument parameter, specially associated to specific HART commands. To this effect, the module should get a configuration (HCD and HWPC blocks) to define the specific instrument to be accessed, and how these commands will relate to each parameter on the block. There is just one HVT block that should be shared among the devices when accessing them. This configuration is already in the HI302’s Flash memory, when it is also possible to include third party configurations according to the application’s needs.

HCD (HART Command Definition) – It contains the HART command description for each device type or version. This description stores information needed by the module to communicate and the data read on the HIRT or HVT blocks. The HCD blocks defining the universal and the common practice commands, as well as all commands specific to Smar instruments, are already stored in the equipment's memory and do not require any configuration from the user. See the Appendix B for details. Configuration of specific commands for third party devices can be made through this block.

HWPC (HART Write Parameter Configuration) - This block stores information about all parameters to be written in the instrument and mapped in the HVT block.

See table with detailed definitions on the Appendix A.

Analog Blocks

In order to support the analog circuits on HI302-I and HI302-O modules, use the MAI or MAO blocks, respectively, to convert the analog 4-20 mA standard to FF or FF to 4-20 mA. Inquire about our availability of AI or AO blocks.

MAI – Multiple Analog Input

The MAI block makes available to the fieldbus network 8 variables of the I/O subsystem through 8 output parameters, namely, OUT_1 to OUT_8. These parameters correspond to the current value, in percentages, on the 8 analog inputs. The current values read through this parameter may be linked to any other block, as part of the control strategy

MAO – Multiple Analog Output

The MAO block makes available to the I/O Subsystem 8 input parameters, IN_1 to IN_8. These parameters correspond to the current value on the 8 analog outputs. Through the MAO block, it is possible to control the current of each loop from another block's output that is part of the control strategy.

Technical Characteristics

Technical Specifications	
Dimensions	• 142 x 40 x 126 mm (max) • 5.6 x 1.6 x 5.0”
Environment Conditions	• Operation: 0 to 60ºC @ 20 to 90% non condensed relative humidity. • Storage: -20 to 80ºC @ 20 to 90% non condensed relative humidity.
HART	• 8 non multiplexed Master communication ports with galvanic isolation of 1000 Vrms, that is, an independent UART per HART port. • A green LED indicating te status of each port. • Maximum of 32 HART devices in multidrop or 8 devices (1 per port). • Supports HART devices version 5 and 6. • Accepts up to 4 different additional device configurations. • Resident configuration for Smar devices. • Allows configuration of specific HART commands.
Fieldbus	• FB3050 Smar dedicated controller • 1 independent H1 channel, with DMA, and 31.25 kbps baud-rate • Communication indicator green LED • Passive MAU (not supplied by the bus) with 500 Vrms galvanic isolation • Physical Layer: ISA-S50.02-1992
Power Supply	• 5 VDC ± 5% @ 400 mA (max), maximum ripple of 20mVpp, via rack. • Maximum consumption of:2W (25ºC). • Green LED indicating the device is powered.
Analog Input 4-20mA	• 250 ohms, 1/8W series resistor. • 16 bit AD converter, 0,15% accuracy @ 25ºC • Low-pass filter input (fc~10 Hz). • Isolation through optical couplers and DC-DC converters, 1000 Vrms. • Inputs are not isolated from each other.
Analog Output 4-20mA	• Passive circuit, only current control (sink). • Maximum voltage allowed on the output terminals: 36 V (zener protection). • Short Circuit and power surges protection, by TVS. • 12 bit DA converter, global accuracy greater than 0,2% of the span (25ºC). • Isolation through optical couplers and DC-DC converters, 1000 Vrms. • Protected by TVS and zener diode. • Supply of the HART devices through external power supply. • Outputs are not isolated from each other.
Memory	• 512 kBytes of Super FLASH for operational systems, applications and resident configuration. • 128 kBytes of SRAM. • 32 kBytes of EEPROM for user configuration.
Processor	• HC11 @ 16 Mhz • Red LED indicating fail. • Yellow LED indicating saving in the EEPROM.

Chapter 2

Installation

This chapter deals with the main physical installation features, namely: mechanical and electrical elements.

IMPORTANT
All comments or considerations made in this manual refer to HART communication using FSK modulation (Frequency Shift Keying).

Installation of the HI302 modules

Mechanical Installation

The HI302 modules are enclosed in the Smar standard plastic housing, like the LC700 and the DFI302. Therefore, they are fully interchangeable on the standard racks. The picture below shows a typical HI302 installation set:

Figure 2.1 – HI302 modules in racks

The HI302 requires **5V @ 400 mA** from the rack. Use Smar DF50 power supply modules. Besides providing a high quality feeding, they also provide a “Power Fail” signal to prevent power failure or AC problems. However, the user can use another power supply, provided it meets the minimum requirement of quality and safety (The PS-AC can also be used).

The other elements follow the same installation procedure as Foundation Fieldbus and HART devices. For more information on installation procedures, visit our www.smar.com site and download a free copy of the Smar HI302 manual.

Electrical Connections

The minimum electrical connections for the HI302 is the power supply, normally connected to the rack, to the connection with the H1 communication bus and to the connection with HART devices. See the following picture for details. Since the HI302 does not supply the devices, it is necessary to use a power supply for them. The DF50 can be used if the devices' consumption doesn't exceed 300 mA (about 12 devices), otherwise the DF52 should be used as shown in the picture.

IMPORTANT
Since the HI302 H1 channel is a passive channel, it is not necessary to use the bus power supply (DF53). For instance, if the DF51 channel is connected to the HI302 channel, they will communicate normally. However, the BT302 impedance should always be used.

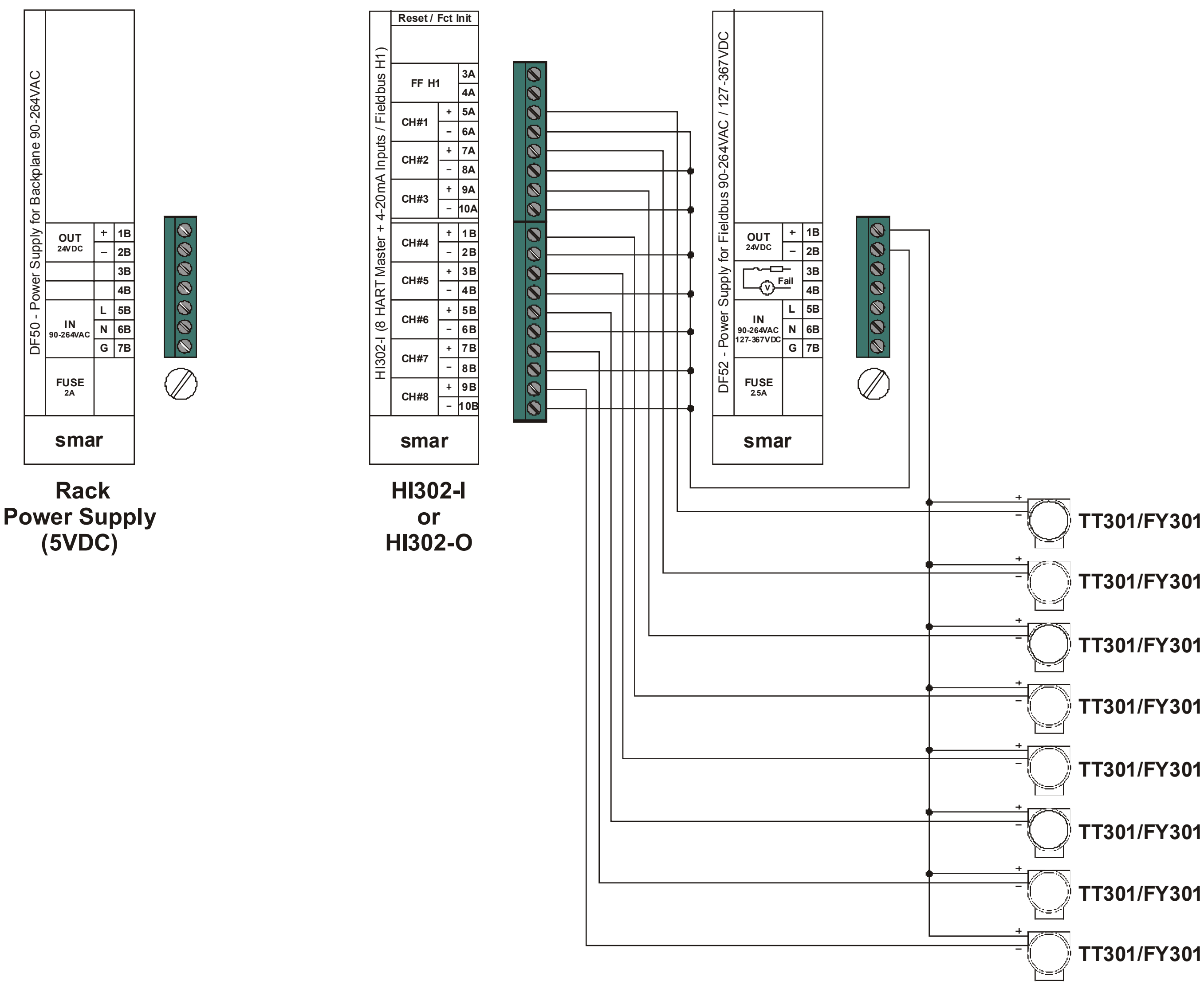

Figure 2.2 – Necessary connections for the HI302

IMPORTANT
The picture above shows the connection of devices supplied by the same power supply module. Remember that the HI302-I and HI302-O analog inputs and outputs are isolated from the field, not from each other, that is, their grounds are internally connected. The HI302-I or the HI302-O can only be connected to an I/O system with a common ground.

The following picture shows an example of an HI302-N connection focused on the HART communication. In this case, connect more devices (up to 32) in parallel, to perform a multidrop communication. In order to simplify the connection below, connect the HART channel in parallel to the device, instead of connecting it in parallel to the resistor. Doing so, a common ground should be used to decrease the wiring length. The connection below makes the channel independent from the device's power supply.

IMPORTANT
The HI302N channels are isolated between one another, thus they can be connected to different I/O subsystems regardless of the grounding or power supply used for the field devices.

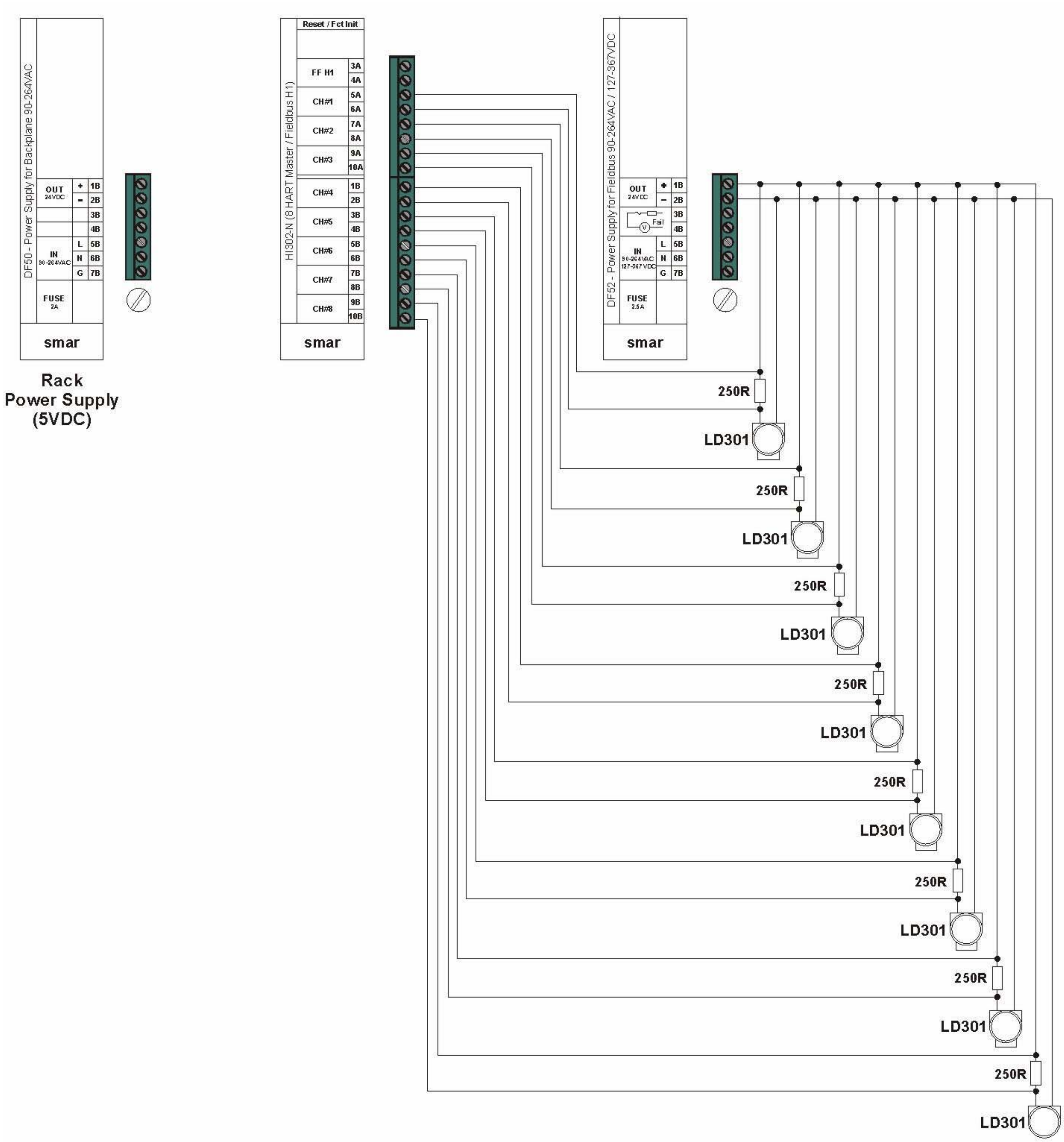

Figure 2.3 – Example of an HI302 connection

NOTE
The 250Ω resistor (see the previous picture) in series with each equipment is the analog input impedance, for example a PLC's input that reads the 4-20 mA signal. If the analog input impedance is less than 250Ω, connect a lower resistor, so that the association of both impedances is at least 250Ω. For instance, suppose that a TT301 is connected to a PLC's input with an impedance equal to 50Ω. So, connect a 200Ω resistor in series with the device's feeding in order to enable the HART communication. The user can also use a 250Ω instead of a 200Ω resistor.

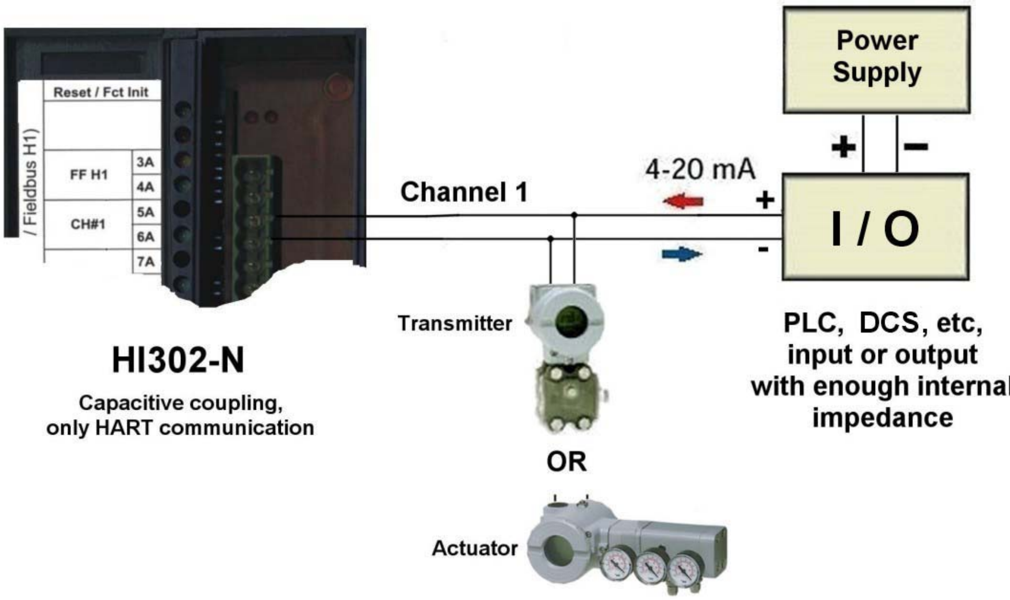

Figure 2.4 – Connections between the HI302-N and a HART equipment in an I/O system

HART Device Installation

Now we will describe the main communication features regarding the device installation. For more detailed information about the devices, please read the specific device manual. Concerning the HART communication, consider that the superimposition of a modulated signal on an analog current signal can deteriorate, if some precautions are not taken. It is important to mention that the HART communication **does not affect** the 4-20 mA analog signal, since the average value of a FSK modulated signal is zero. Thus, if the HART device is already installed, make sure that the minimum impedance (250 Ω) is used and connect the HI302 channel in parallel to the device.

Device Physical Types

Low Impedance Devices
Low impedance devices are typically signaling elements intended to receive current analog signals or serve as master for a multi-drop network. As an example of a low impedance device we can mention the FY301 or input analog cards, such as HI302-I.

High Impedance Devices
High impedance devices control current, either as a mean of analog signaling or at a fixed level in a multi-drop environment. As an example of a high impedance device we can mention the LD301, TT301 or analog output cards such as HI302-O.

These concepts are fundamental when connecting different devices. For example, in the connection shown below, it is not necessary to install a 250 Ω resistor for the HART communication. The transmitter's impedance should be enough. Although it may be necessary to install a resistor in series with the power supply, just to achieve minimum impedance requirement (250 Ω). Each case should be analysed individually according to the equipment's characteristics.

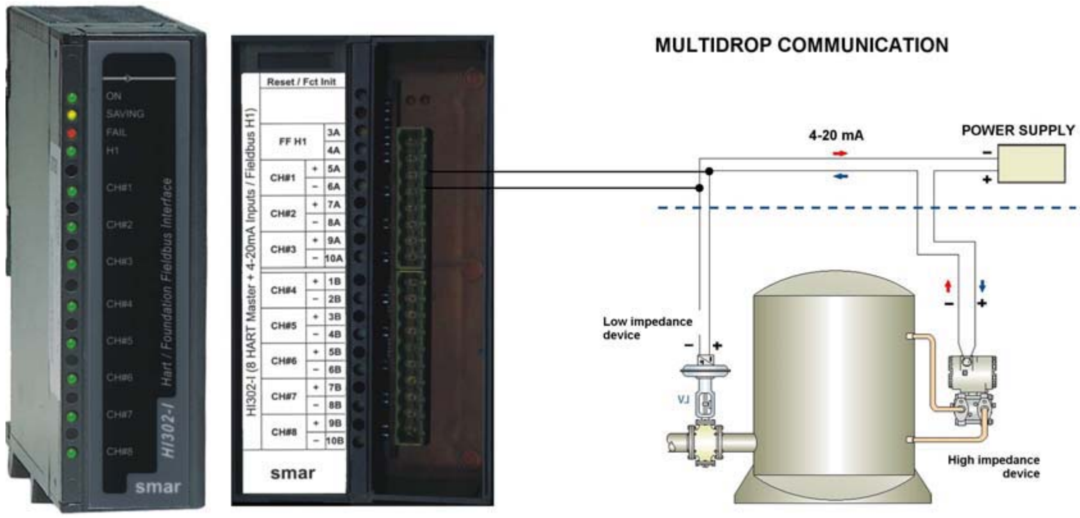

Figure 2.5 – Connection without the 250 Ohm resistor

IMPORTANT
Whatever the topology used, it is important to keep a 250 Ω impedance. In the previous picture, it is not necessary to connect an impedance in series with the power supply, IF the impedance read by the HART channel is at least 250 Ω. In case the impedance is lower than 250 Ω, improve its value to the minimum requirement. The connection above allows the communication between the two devices as long as the identification is done via TAG not via Polling Address.

HART Installation Topology

The HI302 complies with several applications, since the new ones to older installations, where it is necessary to increase the HART device's life span and preserve the investment with the gradual introduction of the Foundation Fieldbus technology. Below are some examples of connections. However, the applications are not limited to these examples and should be considered separately.

Supply Voltage vs. Total Loop Impedance

The total impedance of the devices connected to the pair of cables and the cable impedance should be kept between the operation limits complying with the loop supply voltage. Take a look in the graph below:

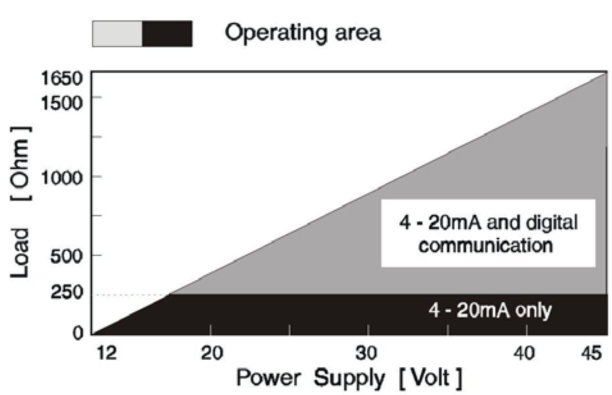

Figure 2.6 – Supply Voltage vs. Total Loop Impedance

Notice that it is very important to keep the minimum impedance (250 Ω) to allow HART communication. Sometimes the voltage supply must be increased to ensure that the system is in the operation area specially when associating devices on the same loop.

HI302-N (without Analog Conversion)

This HI302 option has only HART communication and no circuit for analog conversion. The HI302-N does not have an internal resistor, so it needs an external resistor or an active impedance (PSI301) if many devices are used. It is not necessary to use impedance or external resistor, if the loop has already enough impedance to guarantee communication.

- ***Typical Multidrop***

There are two ways to perform this connection. The resistor can be installed in series with the power supply or in parallel to the HART channel. The first way is shown below:

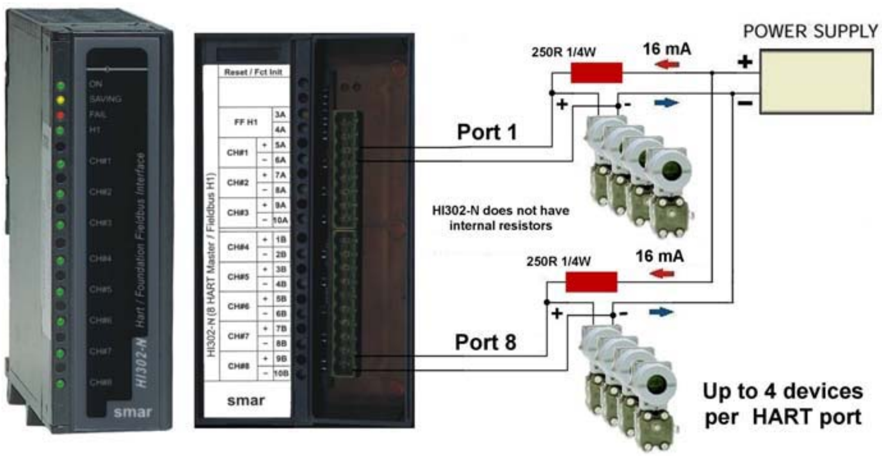

Figure 2.7 – Resistor in series with the power supply

- ***Multidrop with 4-20 mA enabled***

Be careful with this topology because some types of I/O devices do not accept the connections presented below, for example, the DCS's that feed the devices via internal power supply. The I/O device must receive external power supply, like PLC cards or field devices. Despite the complexity, this connection allows an improvement in the use of the HI302 channels and, due to the fact that the current that flows in the loop is in the order of mA, use an active impedance instead of a simple resistor. See the next picture:

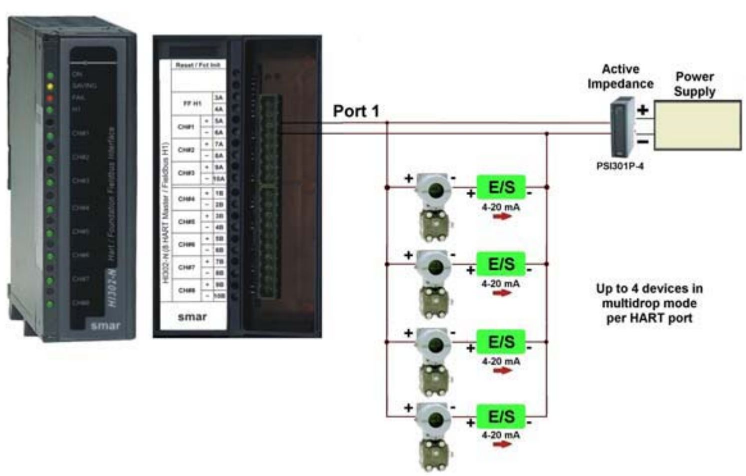

Figure 2.8 – Parallel impedance with the HART channel

HI302-I (4-20 mA to FF Conversion)

In this kind of topology, it is not necessary 250Ω external resistor connected in series with each device, since there is a 4-20 mA sampling resistor in the HI302 analog board serial to the loop. Be careful with a short-circuit in the loop, because the HI302's internal resistor may be damaged. The HI302 does not have internal protection against long run short-circuit.

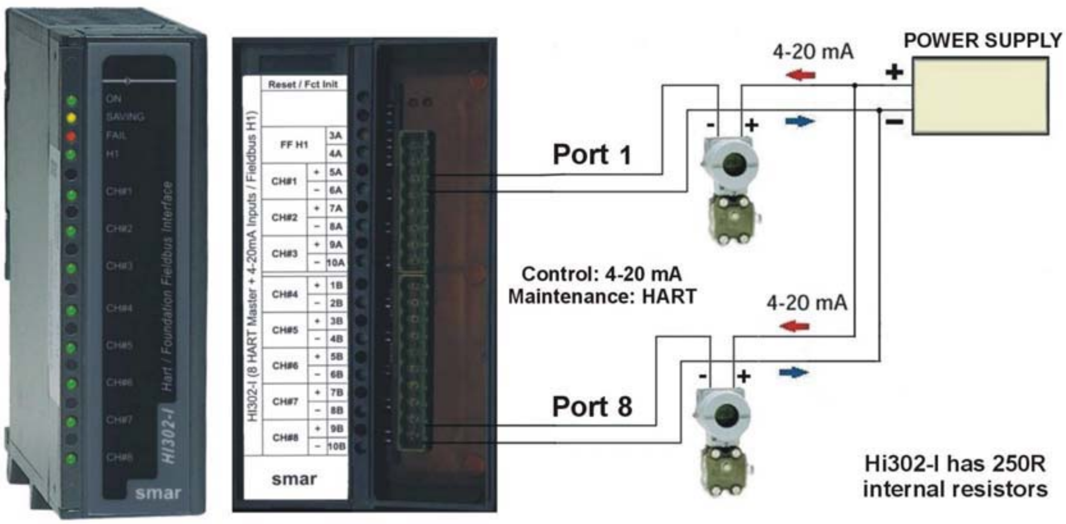

Figure 2.9 – 4-20 mA to FF conversion

HI302-O (FF to 4-20 mA Conversion)

In this topology, it is not necessary to use a resistor in series with the power supply, because the device's internal impedance and the actuator impedance ensures the minimum requirement for HART comunication. However, the user should watch the minimum supply voltage required for total impedance (including the wiring impedance).

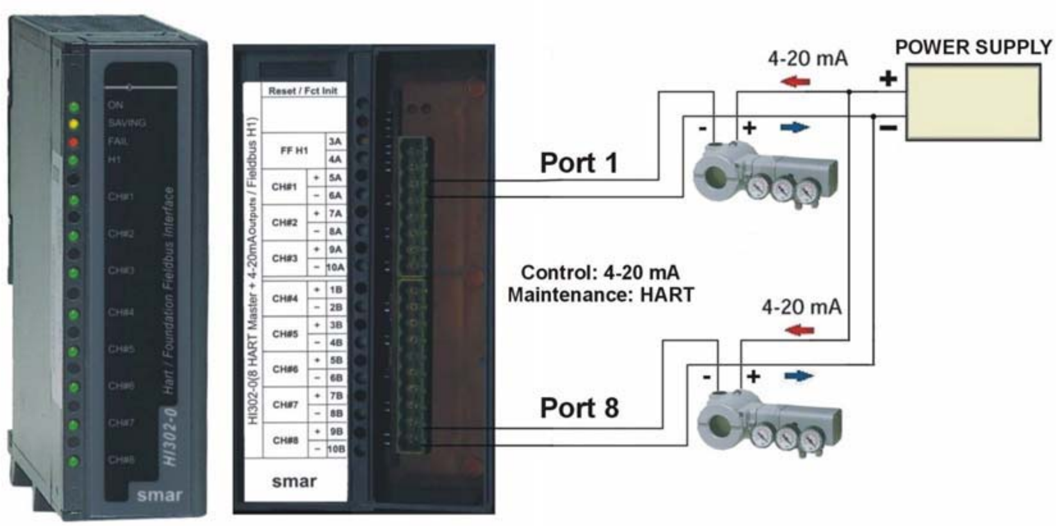

Figure 2.10 – FF to 4-20 mA conversion

Maximum Cable Length

The user may choose from a shielded pair of twisted cables, a multi-pair of cables with a single shield or a combination of these.

IMPORTANT
The shield can be overlooked if noise in the environment or any other interference does not affect the communication.

Use a 24 AWG (0,5 mm) cable for lengths up to 1500 meters. For lengths over 1500 meters, use at least a 20 AWG (0,8 mm) cable.

If a cable longer than 500 m is required, make a detailed analysis of the system to avoid operation failures. According to HFC (HART Communication Foundation), the maximum cable length depends on:

- The cable's resistance, capacitance and inductance,
- The device's resistance and capacitance on the HART channel, as well as the additional equipment.

Due to the complexity of the subject, users should read the *HART Foundation Communication* documentation, specially the *FSK Physical Layer* document.

Other devices in the Loop

The control loop may have additional devices, besides the HI302 and the HART devices. See some common types as follows.

Portable Configurator
As mentioned before, the HI302 operates as a master in most applications. So, there is no problem in using a portable configurator, such as a HPC301. Whatever the installation topology, make sure to install a 250 Ohms impedance serial to the power supply. If no active impedance or resistors are installed, the secondary master device will not communicate.

Indicators and Converters in general
Indicators and converters are very common in industrial installations. They usually have high impedance in the HART communication frequency (1200 to 2200 Hz). Sometimes the introduction of such elements in the loop can prevent communication. However, there is a simple and well known solution for this problem, by connecting a capacitor ranging from 0.1 to 1μF (100v at least) parallel to the device. This capacitor supplies an impedance of hundreds Ohms parallel to the device's impedance, allowing the HART communication.

Switching On the HI302

When the module is turned on, the system will check some important hardware and software components. If any errors are found, the module will not operate and the FAILURE LED (red LED) will light. The HART LEDs (green LEDs) will blink slowly around ¼ Hertz. This check takes approximately 1 minute, so you should wait before checking the HI302 in the Live List or watch for any value.

Updating the HI302 Firmware

To update the HI302 firmware you must use the FBTools program built in the System302 as an integral part of it. To do so, connect the RS232 ***(Code: 102A0927)*** serial cable to the firmware download connector located behind the module.

Before you begin the update, the HI302 module must be set in the boot loader mode. To do so, press the Reset button located at the upper right side in the front part of the module. By pressing it, the YELLOW SAVING LED AND THE GREEN LED will light. Then click the download button in the Serial Download program and wait for the end of the process.

After the completion of the update process, press the Reset button again, in order to put the module back in normal operation. As a standard procedure, the user should always do a Factory Init after updating the firmware. Just press the Fct Init button after the equipment returns to normal operation.

Figure 2.11 - Firmware Programming Connector

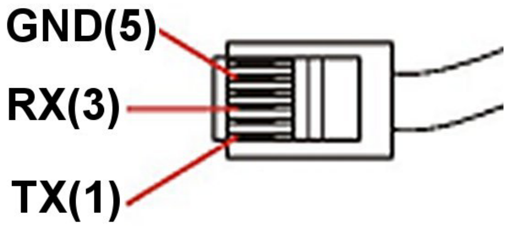

Figure 2.12 - Cable Connector for Firmware Download

Chapter 3

Basic Configuration

Instructions on HI302 Configuration

The minimum configuration to be applied in the Syscon consists of:

- 1 RESOURCE block;
- 1 HCFG block;
- 1HIRT block for each HART device;
- Just one HVT block, if necessary to use specific commands or a complete set of "Common Practice" commands. This block is shared with all of the devices installed.
- 1 HCD block and 1 HWPC block for each specific configuration (that is) not stored in the FLASH memory. These blocks are not necessary for Smar devices since their configuration is embedded in the FLASH memory. See the Appendix for more details.

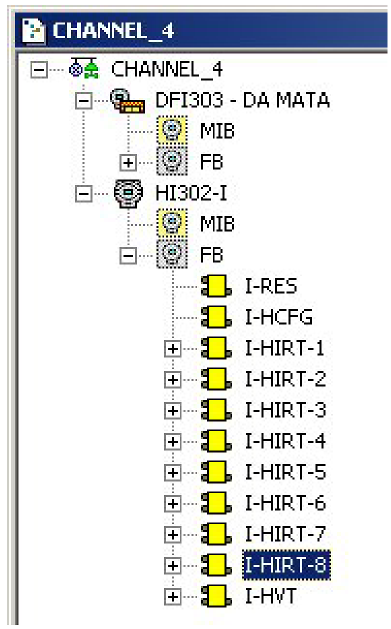

Figure 3.1 – Minimum configuration for the Syscon

The maximum block limit and its quantity in the factory configuration are shown below:

BLOCK	MAXIMUM	FCT INIT
RS	1	1
DIAG	1	1
MAO/MAI	1*	1*
HCFG	1	1
HIRT	32	8
HVT	1	1
HCD	4	0
HWPC	4	0

Except for the HI302-N = 0

IMPORTANT
Whenever a download is performed, wait for the yellow SAVING LED to turn off . Only then turn off or reset the equipment. If the equipment is turned off or reset during the data saving process, the configuration must be redone. See the chapter "HI302 Configuration Example" for more details.

Configuring the HCFG Block

The HCFG block has a series of parameters divided into two categories: operation parameters and diagnostic parameters.

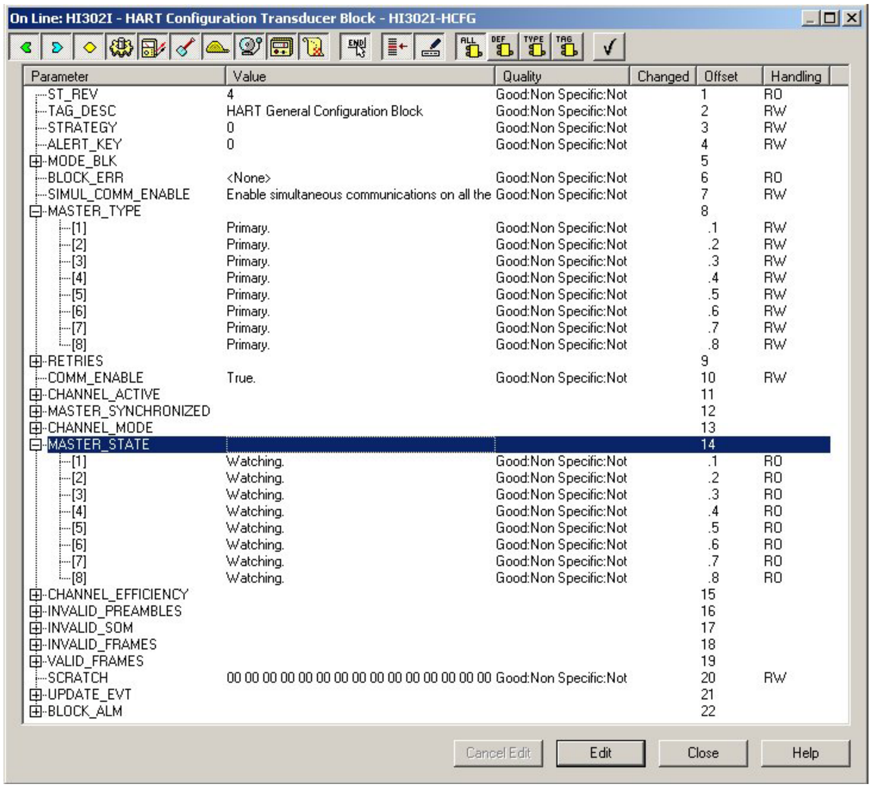

Figure 3.2 – Operation and diagnosis parameters

HART Communication Operation Parameters

- **FIRMWARE_VERSION:** A parameter indispensable to solve problems. If something is not working properly, confirm the equipment version before contacting the technical support.

- **COMM_BEHAVIOR:** Defines the HI302 behavior. It can operate in two ways: if the parameter is **Autonomous**, the HI302 communicates with the HART devices by using the previous configuration, i.e., in an independent way. The second way uses bypass parameters to send and receive HART messages. To do so, this parameter should be configured as **Bypass**.

- **COMM_ENABLE:** This parameter has two important functions:

  - To disable the whole HART communication for maintenance and configuration changes;

  - To validate the configuration loaded in the equipment and then begin the communication (afterwards). Its first value is **DISABLED**. Before making any change in the block configuration, (a must) set it on **ENABLED**, stopping the HART communication. After the configuration download, this parameter should be re-set to **ENABLED.**

ATTENTION
When changing the channel in the HIRT block or downloading new configurations, this parameter is automatically set on **DISABLED**, stopping, then, the HART communication in all channels. When the download finishes **ENABLED** must be set manually. If this procedure is not performed, the HI302 will not work properly.

- **CHANNEL_ACTIVE:** This indicates how many HIRT blocks are instantiated for each HART channel. If no HIRT block is instantiated for the channel, the corresponding element of this parameter will display **NO** and the channel will be deactivated. As a result, there will be no device communication or scanning. The LED channel will blink at approximately ¼ Hz. This parameter is useful to check the configuration. The HVT block is also counted in this parameter.
- **MASTER_TYPE:** This parameter allows for adjusting the channel as a **Primary** master or as a **Secondary** master. Remember that each channel is an independent master. In normal conditions, the channel should be a primary master in order to permit the use of a portable configurator.
- **RETRIES:** This parameter adjusts the number of times the HI302 will try to communicate with a device, before detecting that the device doesn't respond. The standard value is 3 retries.

HART Communication Diagnostic Parameters

- ♦ **MASTER_SYNCHRONIZED**: Indicates if every Master channel has synchronized the communication layer and if each is ready to transmit the HART messages in normal operation or in Bypass mode.
- ♦ **CHANNEL_MODE**: Indicates if the channel is operating normally or there is any device in BURST_MODE.
- ♦ **MASTER_STATE**: Shows the status of the HART channel at every moment:.
  - **WATCHING**, indicates that the channel is only reading data that passes through the line and are crucial to keep the synchronism if there is another Master or any device in Burst mode.
  - **ENABLED**, the channel is free to send a HART message.
  - **USING,** indicates that a message was sent and a corresponding response is expected. The response has to be sent within the maximum number of retries configured in the RETRIES parameter.
- ♦ **COMM_ERRORS**: shows the percentage of detected errors in the communication of each HART channel. If the error percentage is lower than 0,5%, communication is in high quality.
- ♦ **REQUEST_COUNTER**: Totals the number of messages sent by each channel, including the retries.
- ♦ **RETRIES_COUNTER**: Totals the number of repetitions for each channel. A high value in this parameter (>0,5%) may indicate any installation problem or any command not supported by the device.
- ♦ **INVALID_SOM**: Totals the number of invalid SOM (Start of Message) detected in the channel. A high value in this parameter may indicate installation problems or any device with a problem.
- ♦ **INVALID_RX_FRAMES**: Totals the number of HART messages received but not considered because of any inconsistency in the message, for example, checksum error.
- ♦ **VALID_RX_FRAMES**: Totals the number of valid messages received and processed by the HI302, even if they are not addressed to it, for example, OACK, OBACK, STX etc.

ATTENTION
The counter parameters, used for diagnostic purposes, are always reset when the **COMM_ENABLE** parameter goes to **ENABLED**.

Configuring the HIRT Block

This block has a set of parameters that map all of the HART variables that can be accessed by the universal commands and by the some of the most usable "common practice" commands. Remember that there is a HIRT block for each HART device installed and that the configuration may vary according to the application mode and type. For more details about the supported commands, check Appendix A.

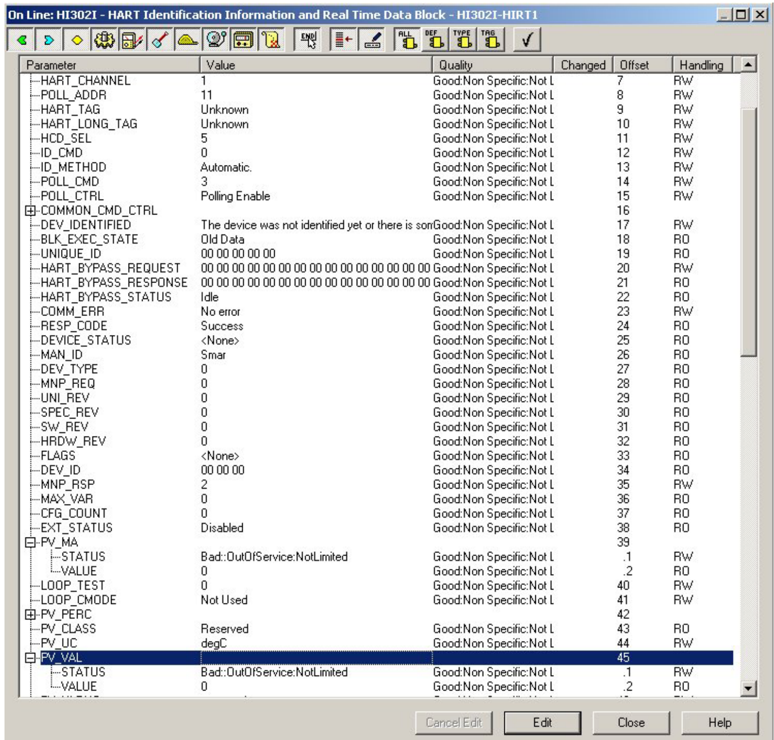

Figure 3.3 – Universal and common practice commands

A minimum set of parameters needs a configuration to allow the HI302 to work properly. Most of the parameters have standard values that are suitable for many operation cases. Thus, it is not necessary to download them. However, a comprehensive analysis must be done in order to determine the best profile for each device configuration. The HI302 offers several resources that must be understood. The parameters that require configuration to work are the following:

- **MODE_BLK**: Should be set on **AUTO**. If it is in OS, the communication with the respective device is interrupted. When the block is set on OS, it returns to the initial Identification state. When it is set on AUTO, all the update and identification processes are repeated.

- **HART_CHANNEL**: Indicates the channel on which the device is installed, from 1 to 8. In normal operation, any change in this parameter will stop the HART communication. See the HCFG.COMM_ENABLE parameter.

- **POLL_ADDR**: Indicates the polling address that has been configured in the device, from 0 to 15. This address is used to recognize the device if the command 0 has been selected in the ID_CMD parameter.

IMPORTANT
If the HART communication is enabled and this parameter has been written, the HI302 will accept the writing in the HART device and will generate a writing transaction. To change this value, without doing it in the device, write *DISABLED* in the HCFG.COMM_ENABLE parameter or set the block on OS.

- **HART_TAG**: the HART device's tag that can be configured by the HI302 module or by a portable configurator. It supports 8 characters and is used by the command 11 to identify the device. The same POLL_ADDR writing observation mentioned above, applies here.

♦ **ID_CMD**: This parameter indicates to the module which universal command identification (0, or 11) will be used to identify the device. The standard value is 0:

- The command 0 uses the "polling address" (POLL_ADDR) and is the most used command.
- The command 11 uses an 8-character tag and can be used provided the device has a tag configured in the block through the HART_TAG parameter. This option is indicated when there are equipments in multidrop.

IMPORTANT
The identification using TAG is very useful when the device is operating in the multidrop mode and the analog signal is enabled, ranging from 4 mA to 20 mA. In this case, the polling address for all devices in the channel should be 0 making the identification impossible through the command 0.

IMPORTANT
If the command 0 is selected, the HI302 will find the device by scanning the polling address from 0 to 15. This is a useful function if the user does not know the device's polling address and its Tag because it identifies the device automatically. This function must not be used in multidrop topologies for obvious reasons.

♦ **POLL_CTRL**: Indicates if the HI302 will keep polling the device.

IMPORTANT
The HI302 will poll the device only if the block has performed the initial procedure successfully and reached the UPDATE status. See the BLK_EXEC_STATE parameter.

♦ **COMMON_CMD_FILTER**: This parameter is a set of filters that prevents a HART command configured and not supported by the device, to be sent. Their elements should be filled in the HCD block with the definition index of the HART command. See the chapter "Example of Configuration" for more details.

♦ **VIEW_SELECTION**: Allows the user to choose the group of variables to be updated cyclically in the polling, according to the following table:

DYNAMIC VARIABLES POLLING MAP - VIEWS DESCRIPTION

View number	00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29
Needs Configuration?	N	N	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
Polling cycle ~ [s]	2	3	2	2	2	3	3	3	4	6	1	2	1	1	1	2	2	2	3	5										
Parameter name	**Updated Parameters**																													
COMM_ERR	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
DEVICE_STATUS	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
ADDITIONAL_STATUS	X	X	X	X	X	X	X	X	X	X																				
LOOP_CURRENT	X	X								X	X	X								X										
PV_PERC	X	X								X	X	X								X										
PV_UC		X								X		X								X										
PV_VAL		X								X		X								X										
SV_UC		X								X		X								X										
SV_VAL		X								X		X								X										
TV_UC		X								X		X								X										
TV_VAL		X								X		X								X										
QV_UC		X								X		X								X										
QV_VAL		X								X		X								X										
A1_UC			X			X		X	X	X			X			X		X	X	X										
A1_VAL			X			X		X	X	X			X			X		X	X	X										
A2_UC			X			X		X	X	X			X			X		X	X	X										
A2_VAL			X			X		X	X	X			X			X		X	X	X										
A3_UC			X			X		X	X	X			X			X		X	X	X										
A3_VAL			X			X		X	X	X			X			X		X	X	X										
A4_UC			X			X		X	X	X			X			X		X	X	X										
A4_VAL			X			X		X	X	X			X			X		X	X	X										
B1_UC				X		X	X		X	X				X		X	X		X	X										
B1_VAL				X		X	X		X	X				X		X	X		X	X										
B2_UC				X		X	X		X	X				X		X	X		X	X										
B2_VAL				X		X	X		X	X				X		X	X		X	X										
B3_UC				X		X	X		X	X				X		X	X		X	X										
B3_VAL				X		X	X		X	X				X		X	X		X	X										
B4_UC				X		X	X		X	X				X		X	X		X	X										
B4_VAL				X		X	X		X	X				X		X	X		X	X										
C1_UC					X		X	X	X	X					X		X	X	X	X										
C1_VAL					X		X	X	X	X					X		X	X	X	X										
C2_UC					X		X	X	X	X					X		X	X	X	X										
C2_VAL					X		X	X	X	X					X		X	X	X	X										
C3_UC					X		X	X	X	X					X		X	X	X	X										
C3_VAL					X		X	X	X	X					X		X	X	X	X										
C4_UC					X		X	X	X	X					X		X	X	X	X										
C4_VAL					X		X	X	X	X					X		X	X	X	X										
D1_UC					X		X	X	X	X					X		X	X	X	X										
D1_VAL					X		X	X	X	X					X		X	X	X	X										
D2_UC					X		X	X	X	X					X		X	X	X	X										
D2_VAL					X		X	X	X	X					X		X	X	X	X										
D3_UC					X		X	X	X	X					X		X	X	X	X										
D3_VAL					X		X	X	X	X					X		X	X	X	X										
D4_UC					X		X	X	X	X					X		X	X	X	X										
D4_VAL					X		X	X	X	X					X		X	X	X	X										

Configuration needed? : N means that no additional configuration must be done in order to read/write the VIEW-related parameters. Otherwise, if **Y**, the corresponding parameters **XX_CODE** must be set to a proper value in order to tell to HI302 which HART variables are associated which the parameters. This is necessary because HART command 33 is used to read those values and HI302 takes the values stored at XX_CODE parameters to assembly HART command 33. Check the device´s specific documentation to find out the variable codes associated with command 33.

Poling cycle ~ [s] : represents the approximate time that each polling cycle will take for that particular VIEW. This time is evaluated as 1 second for each HART transaction (issued command).

Configuring the HVT Block

The HVT block can be seen as a complement for the HIRT block. It includes all non-mapped variables in the HIRT block. Therefore, it is not necessary to have all of the parameters configured, except the following:

- **MODE_BLK**: Should be set on **AUTO**. If it is set on OS, the block operation will stop. When the block is set on OS, it returns to the initial Identification state. When the block is set on AUTO, the block identification and the update process is repeated if there is a valid TAG in the DEV_TAG_SEL parameter.

- **DEV_TAG_SEL**: This parameter doesn't need to be configured for the download. It connects the HIRT block to the HVT block temporarily, allowing the HVT block to communicate with the HART device. This connection is made through the HART_TAG parameter. If the supervisory software has to read the HVT parameters for any device, it should fill the device's HART_TAG. The HVT block then searches in every HIRT block for a written HART_TAG in order to make an association with it automatically. Then, it identifies the HART device and the specific command configuration to be used, as long as this configuration is stored in the Flash memory of the HI302 or in any HCD block. See the chapter about Functioning Theory for more details.

HI302-I - Configuring the MAI Block

All HI302-I inputs have an input circuit for the 4-20 mA acquisition. So, it is possible to instantiate a block with a multiple analog input or with 8-output parameters that provide the percentage value of the analog input related to the 16 mA span (4 mA = 0% and 20 mA = 100%). This block needs no configuration to operate.

- **MODE_BLK**: Set on AUTO.

See the Function Block manual for more details on configurations and options.

HI302-O – Configuring the MAO Block

The HI302-O has 8 HART channels in parallel to circuits that control the loop current and the actuators connected to them. The input parameter value should always be written in percentage. The following parameters should be configured:

- **MODE_BLK**: Set on AUTO.

There are other configuration options, for example, the output value in case of failure.
See the Function Block manual for more details on configurations and options.

Starting the HI302 Operation

After configuring the block, download it. Upon the download completion, set the HCFG.COMM_ENABLE parameter on ENABLED to enable the configuration and start the HART communication over all channels.

IMPORTANT
After the download is completed, the configuration should be saved in the non-volatile memory. Saving is automatic and starts with the download. It will take approximately 20 minutes, according to the configuration size. While the SAVING LED is lit, the HI302 cannot be turned off or reset, otherwise, the configuration will be lost..

If the user has devices whose configuration is already in the HI302 memory, e.g. Smar devices, go to the chapter about operation and functioning theory. In case the user has to set a specific configuration, see the chapter about advanced configuration, first. For more details on how to configure the HI302, visit our site www.smar.com.

Calibrating the HI302 analog boards

In order to achieve the most accurate operation of the HI302-I or HI302-O, it is important to calibrate the analog boards. During the factory tests, a preliminary calibration is done, being sufficient for most applications. However, it may be necessary to perform a new calibration in the field. To do so, follow the steps described in the Help of the HCFG.ANALOG_INPUT_TRIM or HCFG.ANALOG_OUTPUT_CAL parameters. See a summary below.

Important
When the calibration is done, it is valid for the pair GLL1193+(GLL1205 or GLL1194). If the analog board has been changed for any reason, a new calibration should be done because the calibration data is stored in the base board for that particular analog board (GLL1193).

HI302-I Calibration (GLL1205)

To calibrate a GLL1205, the user should use an accurate current source and follow the steps below:

1. Apply a 12 mA current (50% of the 16mA span) to each input. The calibration can be done in only one channel or in all of the 8 channels at once.
2. After stabilizing the current, write the number of the desired channel on the HCFG.ANALOG_INPUT_TRIM parameter or write All Channels to calibrate all of them at once.
3. Check, in the MAI block, if the value of the current is 50%. If the reading, in any channel, is incorrect, re-do the procedures.
4. If the reading in the 8 channels is 50%, write on the HCFG.ANALOG_INPUT_TRIM parameter the Trimmed and Checked value in order to save the calibration data.
5. Wait until the SAVING Led turns off and then, turn off the device.

The accuracy of the HI302-I analog input is approximately 0.15% of the span for all the operation temperature range (0 to 50^{o}C). During operation, for the temperatures around the calibration one (±2 ^{o}C), the accuracy can reach approximately 0.05%.

HI302-O Calibration (GLL1194)

The user should use an accurate multimeter to calibrate the GLL1194. Follow the steps below:

1. Write 50% on all of the input parameters of the MAO block.
2. Measure the current in each loop using the multimeter. Write each read value on the corresponding element of the HCFG.ANALOG_OUTPUT_CAL[channel] parameter. Write all of the numbers displayed on the multimeter to maximum accuracy.
3. Measure the current again and check if its value has changed, that is, approximately 12 mA.
4. In case the current has changed, write on the HCFG.ANALOG_INPUT_TRIM parameter the Trimmed and Checked value to save the calibration data.

The accuracy of the HI302-O analog output is approximately 0.5% of the span for all of the operation temperature range (0 to 50^{o}C). During operation, for the temperatures around the calibration one (±2 ^{o}C), the accuracy can reach approximately 0.1%.

Chapter 4

Advanced Configuration

Specific HART Command Configuration with HCD and HWPC Blocks

The HI302 allows the use of any HART device with a suitable configuration for reading and writing the necessary variables. As seen previously, all of the Universal and some Common Practice commands are available in the HIRT block, therefore no additional configuration is necessary. The whole group of Common Practice and Universal commands depends on extra configurations that can be stored in the FLASH memory of the HI302 or in the **HCD** and **HWPC** blocks. This chapter contains information on how to create a simple configuration by using the HCD and the HWPC blocks.

Basic Instructions on HART protocol

HART is a master-slave protocol and is loosely organized around the ISO/OSI 7-layer model for communication protocols (see the table below). The application layer is the topmost layer in the Open System Interconnect (OSI) model.

	OSI Layer	Function	HART
7	Application	Provides the user with network capable applications.	Command oriented. Predefined data types and application procedures.
6	Presentation	Converts application data between network and local machine formats.	
5	Session	Connection Management Services for applications.	
4	Transport	Provides network independent, transparent message transfer	
3	Network	End to end routing of packets. Resolving network addresses.	
2	Data Link	Establishes data packet structure, framing, error detection, bus arbitration.	A binary, byte oriented, Token passing, master/slave protocol.
1	Physical	Mechanical/electrical connection. Transmits raw bit stream.	Simultaneous analog & digital signaling. Normal 4-20 mA copper wiring.

The Application layer in HART defines the commands, responses, data types and status reporting supported by the protocol. In addition, there are certain conventions in HART (for example how to trim the loop current) that are also considered as part of the Application layer.

Types of HART Commands

The HART Communication Foundation (HFC) arranges the commands into three categories:

- **Universal Commands:** The Universal Commands must be implemented in the HART devices. These commands have a well-defined format, assuring the interoperability between different systems, for example, a Smar device and a third portable configurator. The HIRT block supports all of the Universal commands.

- **Common Commands (Common Practice):** These commands also have a well-defined structure and syntax. However, it is not mandatory to implement them, that is, the user is free to do it or not according to the device functionality. The configuration of the standard commands, used by the HIRT block, implements these commands, for example, the command 33. The device may not support some commands. As a result it is necessary to use the common command filter. See the Appendix B for more details.

- **Specific Commands:** These commands allow the user total operation flexibility. Thus, it is possible to know their syntax. To do so, the HI302 has blocks that allow the user to make a configuration to use any accessed variable (reading or writing variable) through the specific commands. See below an example of how to create a configuration.

Describing the HART commands

The structure of the HART command is similar regardless of the category. See below some examples of HART commands and how to create a reading and writing configuration for the variables. For a more detailed explanation, see the device manual.
We will use Universal Commands, although any HART command can be described the same way.

Important: Get the HART command documentation beforehand, as shown later in this manual. Get these documents from the HART manufacturer.

Since it is a transaction oriented protocol, every HART command has two sections: REQUEST data and RESPONSE data.

Each section is composed by fields divided into bytes. Each field has a specific size in bytes and a type, such as the 8 bit integer or a 32 bit number in a floating point. It is important to identify each field type to get the HART command description together.

Setting the definition of the HART Commands

Each HCD block parameter has the definition of one HART command, that is, 50 different commands can be described. This definition has information about the HART Command and about all the elements that compose this command. Each HART operation, related to a HART command, is composed of two messages: Request, a message from the master to the device, and Response, the answer for the master's request. The definition has the following HEXADECIMAL format:

Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8	Byte 9	…	Byte N

The first four bytes of the command definition are the header, and they mean:

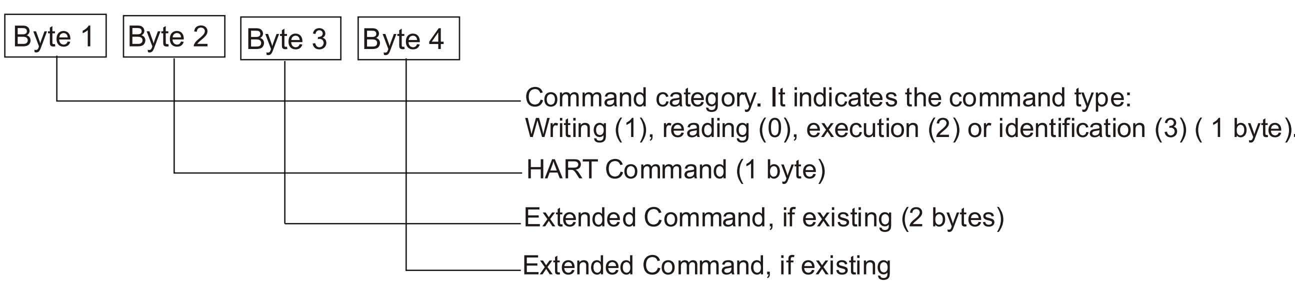

Figure 4.1 –Syntax for the description of the HART command

The command can have the following values:

- 0x00 – **READING**: command 3 for example.
- 0x01 – **WRITING**: command 18 for example.
- 0x02 – **EXECUTION**: commands that do not have arguments, neither from the REQUEST nor the RESPONSE, for example, the command 42.
- 0x03 – **IDENTIFICATION**: the commands used to identify the device, such as 0 or 11.

After the definition header, there is the command definition composed of fields with four bytes each. Each four byte-field defines a HART variable used in the command sending or received in the answer. In the HCD block, there are definition parameters with 10 and 25 variables. The first 40 parameters only support 10 variables, and the remaining 10 parameters support 25 variables.

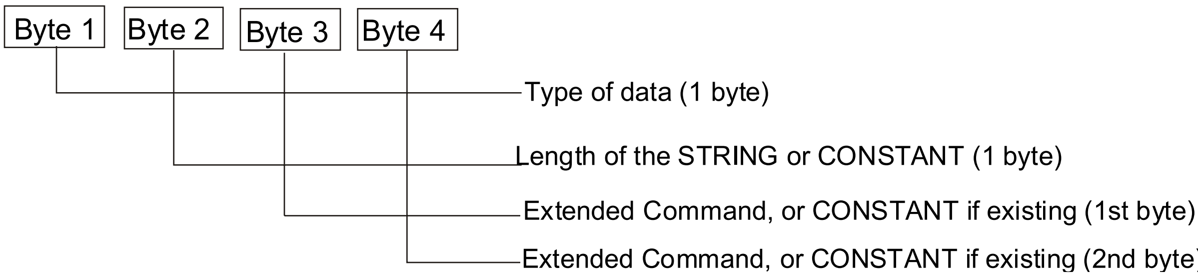

Figure 4.2 –Command definition

The data implemented are the following (always with reference to the HART protocol):

Type of Data	Request	Response
No data	0x00	0x00
Unsigned 8 (converted into unsigned 16)	0x01	0x81
Unsigned16	0x02	0x82
Unsigned24 (converted into unsigned32)	0x03	0x83
Unsigned32	0x04	0x84
IEEE754S, Float Single Precision	0x05	0x85
IEEE754D, Float Double Precision	0x06	0x86
Packed ASCII (converted to ASCII)	0x07	0x87
ASCII	0x08	0x88
Date	0x09	0x89
Constant	0x0A	0x8A

IMPORTANT: The first bit of each type defines whether the variable is present in the request or response. For example, an Unsigned 8 used in the request will be **0x01**. If used in the response, it will be **0x81**.

Configuring the HCD block

Each configuration must receive a code, in order for the configuration to be found, when the HI302 needs to run the HVT block configuration. This code is specific for each HART equipment and should be written in the **HCD_CODE** parameter. The code is formed by 5 bytes:

- MAN_ID: equipment's manufacturer ID
- DEV_TYPE: device type
- UNI_REV: version of the universal HART commands
- SPEC_REV: version of the specific command covered by the configuration
- SW_REV: resident software version in the HART equipment.

When operating, these 5 bytes are read from the HIRT block in order to find the configuration to be used by the HVT block. Look in the manufaturer's documentation or the reading by the HIRT block to obtain these values.

The HCD block has the parameter **DEVICE_INFO**, where the information should be placed for the identification of the configuration, such as the device's name, model, firmware version, etc. These informations, as well as the HCD_CODE, are shown in the HVT block when in operation.

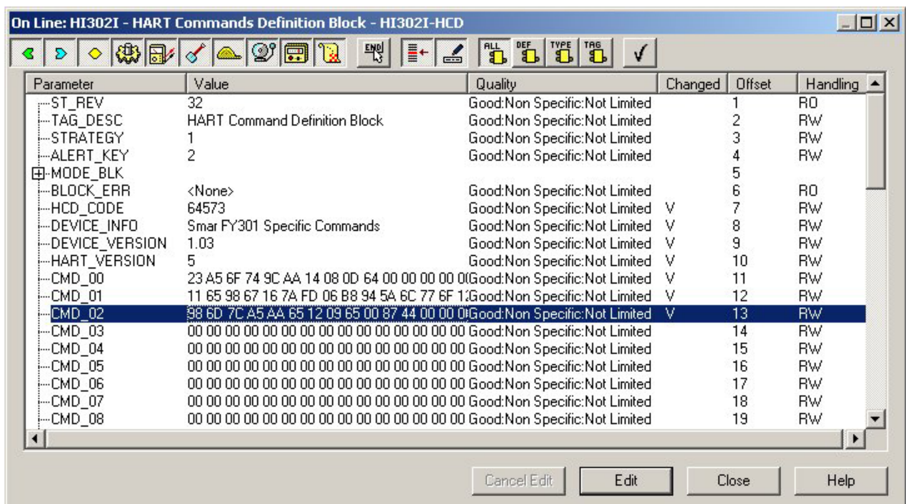

Figure 4.3 – HCD block configuration

Mapping the HART variable as FF parameters

To configure the HCD block, the user should have the HART documentation describing the commands to be used. Using the HART command description, the user should list the HART variables to be accessed, paying attention to their type (1 byte, 3 bytes integer, float, packed ASCII, ASCII, etc).

HVT Allocation Map and Command Description

It is possible to build a configuration used by the HIRT block. In most situations, it is recommendable to use the standard configuration (5 or 6), since the HCD and HWPC blocks will define specific configurationsto be used by the HVT block.

The HVT block has parameters of general use that are arrays with the following type of parameters: Unsigned 8, Unsigned 16, Unsigned 32, Float Single Precision and String. In order for each of those parameters to have a useful meaning, it is necessary that it is associated to a HART variable. See in the Appendix C, a model of the Allocation map.

Request Parameters

Each Request Parameter is described by two types of information: the **TYPE** (1 byte) and the **LENGTH** (1 byte), for STRINGS or CONSTANTS, besides the relative index and the sub-index corresponding to the HIRT or HVT block, where the parameter is stored. The parameters are described by their sending sequence in the HART frame. Even if the parameter is from a different type, the HI 302 converts automatically. Remember that the DATA TYPE should always appear in the documentation of the HART command and not in the parameter type of the FF block. The parameters are described in order in the HART frame.

Response Parameters

The Response Parameters are described the same way as the Request Parameters.

- The most significant bit should be "one" in the RESPONSE DATA TYPE field. There will be an implicit conversion for reading and storage data.
- When the type of data is Packed ASCII or ASCII, the definition will have a second byte (string length) that will indicate the length of the string that must be read and saved in characters, or the size of the constant, which could be 1 or 2 bytes.

What About the RESPONSE CODE?

It is up to the HI302 to read the RESPONSE CODE for each transaction concluded successfully and to update the corresponding parameters (COMM_ERR, RESP_CODE, DEVICE STATUS, parameters' status DS-65, etc.), so don't worry about the configuration for the response code, for the HI302 reads both bytes automatically.

The mapping and the estimated configuration could be done as it follows:

Command 0

- **Command 0: universal identifier using the polling address.**

Dados de requisição

Byte	Format	Description
None		

Dados de Resposta

Byte	Format	Description
0	Unsigned-8	“254”
1	Enum	Manufacturer’s ID code.
2	Enum	Instrument type.
3	Unsigned-8	Minimum Preambles number required by the Master to the Slave. This number includes both preambles used in Asynchronous Physical Layers (according to the Limit) to detect the beginning of the message.
4	Unsigned-8	Number of the main Review of the Universal Command implemented by the device.
5	Unsigned-8	Review Level of the device.
6	Unsigned-8	Review Level of the device’s Software. Levels 254 and 255 are reserved.

Header			
Category of the HART Command	**HART Command**	**Extended Command 1**	**Extended Command 2**
0x00	0x00	0x00	0x00

- The command 0 is originally an IDENTIFICATION command, not a READING command, used only as an example. Since there are no variables to be sent on the Request, we will go directly to the Response.

HART Variable	Type		HVT Parameter	Type	Code	Length	Index	Subindex
Constant "254"	Unsigned8	→	Descartado	Constant	0x8A	0x00	0x00	0x00
Manufacturer's ID code.	Enum	→	U8B_ARRAY_1[1]	Unsigned8	0x81	0x00	0x0B	0x01
Device's type code.	Enum	→	U8B_ARRAY_1[2]	Unsigned8	0x81	0x00	0x0B	0x02
Minimum number of Preambles.	Unsigned8	→	U8B_ARRAY_1[3]	Unsigned8	0x81	0x00	0x0B	0x03
Review of the Universal Commands.	Unsigned8	→	U8B_ARRAY_1[4]	Unsigned8	0x81	0x00	0x0B	0x04
Device Review.	Unsigned8	→	U8B_ARRAY_1[5]	Unsigned8	0x81	0x00	0x0B	0x05
Software Review of the Device.	Unsigned8	→	U8B_ARRAY_1[6]	Unsigned8	0x81	0x00	0x0B	0x06

- The first byte received on the response of the command 0 will always be 254, so it does not need to be mapped to any parameter and will be discarded when the response is processed, so the INDEX RELATIVE will receive the value 0.
- Even if the command has other variables in the sequence, the HI302 only processes the described variables. Suppose only the sixth variable is of interest. It is necessary to describe the first five, discarting the values until the sixth. The following variables don't need description.

Value that should be written in the **CMD_00 (Relative Index 11 - 0x0B)** parameter of the HCD block:

00 00 00 00 8A 00 00 00 **81 00 0B 01** 81 00 0B 02 **81 00 0B 03** 81 00 0B 04 **81 00 0B 05** 81 00 0B 06

Command 3

- **Command 3: reads the loop current and four manufacturer's pre-defined dinamic variables.**

Request data

Byte	Format	Description
None		

Response Data

Byte	Format	Description
0-3	Float	Loop Current of the Main Variable (milliamperes)
4	Enum	Main Variable Unit Code
5-8	Float	Main Variable
9	Enum	Secondary Variable Unit Code
10-13	Float	Secondary Variable
14	Enum	Tertiary Variable Unit Code
15-18	Float	Tertiary Variable
19	Enum	Quarternary Variable Unit Code
20-23	Float	Quarternary Variable

Header			
Category of the HART Command	**HART Command**	**Extended Command 1**	**Extended Command 2**
0x00	0x03	0x00	0x00

- This command does not have variables to be sent with the Request. So, we will go straight to the Response.

HART Variable	Type		HVT Parameter	Type	Code	Length	Index	Subindex
Loop Current of the Main Variable (mA)	Float	→	FLOAT_ARRAY_1[1]	Float	0x85	0x00	0x10	0x01
Main Variable Unit Code	Enum	→	U8B_ARRAY_1[7]	Unsigned8	0x81	0x00	0x0B	0x07
Main Variable	Float	→	FLOAT_ARRAY_1[2]	Float	0x85	0x00	0x10	0x02
Secondary Variable Unit Code	Enum	→	U8B_ARRAY_1[8]	Unsigned8	0x81	0x00	0x0B	0x08
Secondary Variable	Float	→	FLOAT_ARRAY_1[3]	Float	0x85	0x00	0x10	0x03
Tertiary Variable Unit Code	Enum	→	U8B_ARRAY_1[9]	Unsigned8	0x81	0x00	0x0B	0x09
Tertiary Variable	Float	→	FLOAT_ARRAY_1[4]	Float	0x85	0x00	0x10	0x04
Quarternary Variable Unit Code	Enum	→	U8B_ARRAY_1[10]	Unsigned8	0x81	0x00	0x0B	0x0A
Quarternary Variable	Float	→	FLOAT_ARRAY_1[5]	Float	0x85	0x00	0x10	0x05

- Observe that the Unsigned8 parameters are following those used to map the variables of the 0 command. That is not necessary. Any parameters of the U8B_ARRAY_X can be used.

Value that has to be written in the parameter **CMD_01 (Index Relative 12 - 0x0C)** from the HCD block:

00 03 00 00 85 00 10 01 **81 00 0B 07** 85 00 10 02 **81 00 0B 08** 85 00 10 03 **81 00 0B 09** 85 00 10 04 **81 00 0B 0A** 85 00 10 05

Command 13

- **Command 13: reads the TAG, Descriptor and Date.**

Request Data

Byte	Format	Description
None		

Response Data

Byte	Format	Description
0-5	Packed	Tag
6-17	Packed	Descriptor
18-20	Unsigned-24	Data Code

Header			
Category of the HART Command	**HART Command**	**Extended Command 1**	**Extended Command 2**
0x00	0x0D	0x00	0x00

- The command 13 reads the TAG variables, Descriptor and Date. The command 18 (described in the next section), writes these variables in the device. It is important to first describe the reading command and execute the reading before the writing in order to avoid that the parameters need to be initialized by the application before writing in one variable. For example, writing the tag without reading or initializating the descriptor and the Date would receive the value of the block parameter, which can be read from another device.

HART Variable	Type		HVT Parameter	Type	Code	Length	Index	Subindex
TAG	Packed ASCII	→	String_01	String	0x87	0x06	0x18	0x00
Descriptor	Packed ASCII	→	String_06	String	0x87	0x0C	0x1D	0x00
Date	Unsigned24	→	Descarted	Unsigned32	0x89	0x00	0x16	0x01

- Note that in this description the DATE type is mapped to an mapeado para um Unsigned32.

Value that should be written in the **CMD_02 (Index Relative 13 - 0x0D)** parameter from the HCD block:

00 0D 00 00 87 06 18 00 **87 0C 1D 00** 89 00 16 01

Command 18

Request Data

Byte	Format	Description
0-5	Packed	Tag
6-17	Packed	Descriptor used by the Master for a Register.
18-20	Unsigned-24	Date Code used by the Master for Register (for example, last or next calibration date).

Response Data

Byte	Format	Description
0-5	Packed	Tag
6-17	Packed	Descriptor
18-20	Unsigned-24	Date Code

Note: The values from the response data reflect the real values used by the field devices.

Note: Most devices only store the date, and some field devices may not detect an invalid date code.

Header			
Category of the HART Command	**HART Command**	**Extended Command 1**	**Extended Command 2**
0x01	0x12	0x00	0x00

- The command 18 has variables which should be sent on Request. The request variables should be defined, and then, the response variables. Observe that the Bit7 for the Request variables Type is **"0"**, while the Response variables has this bit equal to **"1".**

HART Variable	Type		HVT Parameter	Type	Code	Length	Index	Subindex
TAG	Packed ASCII	→	String_01	String	0x07	0x06	0x18	0x00
Descriptor	Packed ASCII	→	String_06	String	0x07	0x0C	0x1D	0x00
Date	Unsigned24	→	Discarted	Unsigned32	0x09	0x00	0x16	0x01
TAG	Packed ASCII	→	String_01	String	0x87	0x06	0x18	0x00
Descriptor	Packed ASCII	→	String_06	String	0x87	0x0C	0x1D	0x00
Date	Unsigned24	→	Discarted	Unsigned32	0x89	0x00	0x16	0x01

***IMPORTANT**: Besides the command description, it is necessary to create a complementary configuration on the HWPC block so that the HI302 may write on the mapped parameters. **Without the HWPC configuration, the variables may only be read**.*

Value that should be read in the parameter **CMD_03 (Index Relative 14 - 0x0E)** from the HCD block:

01 12 00 00 07 06 18 00 **07 0C 1D 00** 09 00 16 01 **87 06 18 00** 87 0C 1D 00 **89 00 16 01**

What about the parameter writing?

The HART variables that can be written on the device need special attention. Although the command definition used for reading is needed, information on how this variable relates with the block parameter where it has been mapped and with the HART writing command. This information is stored in the HWPC block.

Setting the HWPC block Configuration

The HWPC block completes the configuration done in the HCD block. For each HCD block with parameters that can be written, it is necessary to have a corresponding HWPC block with the same code in the definition library HCD_CODE. This value should be written in the **HWPC_CODE** parameter.

The HWPC block parameters are 20 position string. Each string allows 4 writing parameters, and each parameter has 5 bytes.

1. The parameter location in the block.

<Byte1>: Parameter relative index in the block.
<Byte2>: Parameter Sub-index.

2. Parameter Writing Command

<Byte3>: Parameter relative index of the HCD block that defines the command used to write in the HART device.
<Byte4>: Relative position of the HART variable in the HCD block parameter.

3. Parameter Reading Command

<Byte5>: Parameter relative index of the HCD block that defines the command used for the parameter reading.

This parameter should have information about HART commands that are used to write and read the corresponding parameters in the HVT and HIRT blocks. The HWPC block has 30 parameters of this type, allowing the writing in 30 different HART variables.

IMPORTANT
If the parameter is written and there is no corresponding configuration in the HWPC block, no HART command will be generated to change the variable in the device, even if there is a writing command in the HCD block.

Configuration of the HWPC for the given Example

The configuration for writing operation are for the following HVT parameters:

HVT Parameter	Relative Index	Mapped HART variable	Written HCD Index	Written HCD Index
String_01	24	TAG	14	13
String_06	29	Descriptor	14	13

With this information, just transpose these data for one of the HWPC parameters, as in the figure below:

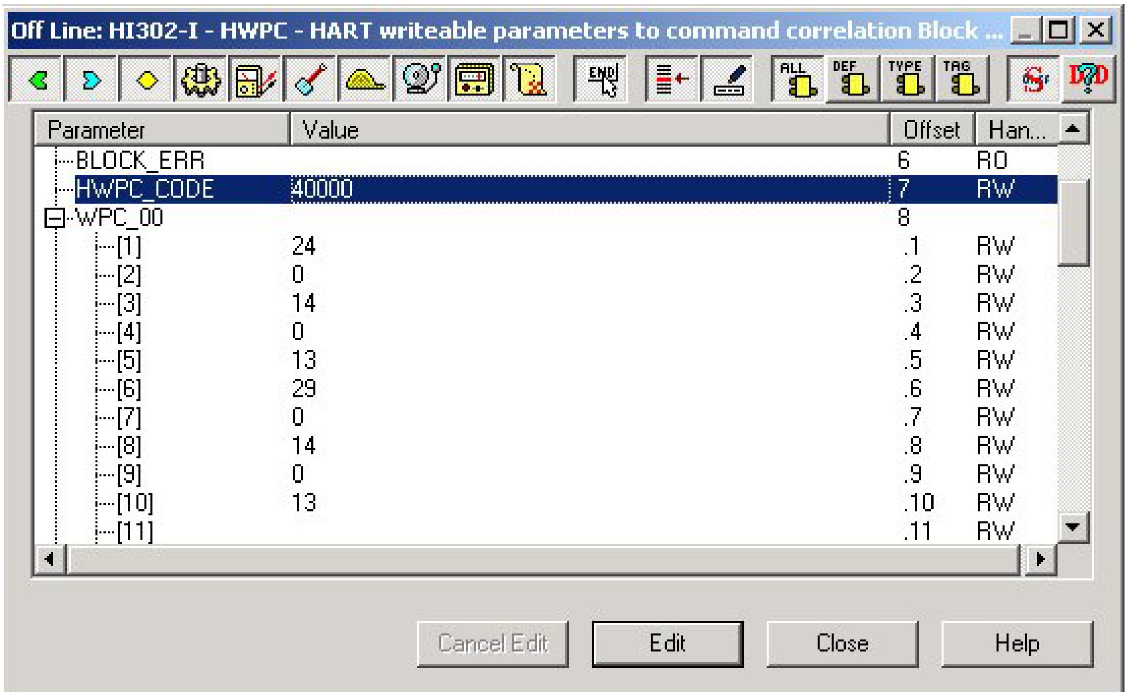

Figure 4.4 – HWPC Configuration

Chapter 5

Operation

An introduction on the module firmware, its overall performance, signal LEDs status and work in steady state is presented below. See the parameter HCFG.FIRMWARE_VERSION to know the firmware version.

Initialization

When initialized, the resident program performs a checkup on the module for its perfect working condition:

- The hardware's health, access to memories, access to UARTs, analog board's condition, etc;
- Start up on the Fieldbus communication;
- Search for a valid block configuration in the memory. If found, the HART communication is started.

After this stage, the front LEDs reflects the result of the initialization and the self-diagnostic.

LEDs Status

LED ON
This LED must remain lit, indicating that the module is supplying and that the HOT SWAP circuit is working normally. If this LED is off and the outside supplying is normal, it indicates that there is some problem with the HOT SWAP circuit, or that the short-circuit protection is working. Turn off the module, immediately, restart it a few minutes later and if the problem continues, send the device to maintenance.

LED FAIL
This LED has three statuses:

- **OFF**: there is no error to be reported, the minimum configuration has been set, correctly saved on the EEPROM and the HI302 should operate well.

- **Blinking**: the HI302 has not received the minimum configuration to operate. Send the minimum configuration via Syscon. Another reason is any fatal error that stops the HI302 operation. Turn it off and after a few seconds, restart it. If the problem persists, press the "FCT INIT" button. If these procedures were not effective, send the device to maintenance.

LED SAVING
The HI302 module has a serial EEPROM to store non-volatile data from its configuration. However, to avoid the premature wear off of the memory, a mirror of its contents in SRAM is used. While this LED remains lit, avoid the HI302 to be switched off or reset, as this will indicate that static data from the SRAM mirror is being saved on the EEPROM.

ATTENTION
After sending the configuration to the HI302, this LED may stay lit for a few minutes while data is being saved permanently on the EEPROM. If it is turned off or reset, this configuration will be lost, and must be sent again.

H1 LED

This LED indicates that an activity is going on in the Fieldbus Communication. At each information package sent by the HI302, the LED will blink. If the LED doesn't blink, it indicates that the HI302 is not communicating with the net. Check the H1 segment configuration (LAS). Then switch off the HI302 and restart it a few seconds later. If it doesn't return to the "LIVE LIST" after a few minutes, try a "FCT INIT". If this step fails, send the HI302 to maintenance.

CH#1 to CH#8 LEDs

These LEDs indicate the situation on every HART channel, under the following statuses:

- **Unlit**: the channel is active but there is no HART communication. The devices of the channel may have problems or their blocks are in OS mode. Check the devices and the block configuration.
- **Lit**: indicates the HART communication is deactivated (the COMM_ENABLE parameter in the HCFG block is at FALSE). This may happen in two situations:
    - "On line" configuration change of channels or devices;
    - Creation or exclusion of blocks.
- **Blinking regularly and fast (2Hz)**: indicates that the HI302 has not yet received the minimum configuration, that is, the HCFG block hasn't been found. Re-download the configuration or press the Fct Init to correct the problem.
- **Blinking regularly and slow (1/4 Hz)**: the channel is deactivated because there is no instantiated HIRT block using it. This information is useful to check if there has been a mistake in the configuration of the HIRT blocks. Check each HART channel configuration on the HIRT blocks.
- **Blinking irregularly**: In this status the LED blinks according to the HART rate of messages exchanged in the channel. If it remains unlit for too long, means the device may have problems or the configuration may be wrong. See the BLOCK_ERR and BLK_EXEC_STATE parameter of each HIRT block. After it is set to UPDATED, the HART communication is ended if polling is disabled on the HIRT block. Consequently, the LED will be off, if there is no other HIRT block using the channel.

Auxiliary Push Buttons

There are 2 auxiliary push buttons inside the front panel, on the topside, just above the connectors. The RESET button at left is used to reset and to put the equipment in the Bootstrap mode in order to update its firmware. The second push button, located at the right side is used to initialize the memory to its factory state. This second button must be carefully used.

Understanding the HART Communication

To follow the Communication operations performed by the HI302, the user should understand the diagnostic parameters from the HIRT block.

BLK_EXEC_STATE parameter

The BLK_EXEC_STATE may assume the following values:

- **0x00: IDENTIFICATION,** means the device is under an identification process or the block communication is stopped (MODE_BLK = OS).
- **0x01: OLD DATA,** only valid for HIRT blocks. It means the block has detected a HART device alteration. It sends the command HART 38 to reset the configuration changed flag in the device. If the HI302 has sent the writing request, the block goes straight to UPDATED. If the writing comes from an external source, the IDENTIFICATION and UPDATING process restarts. After this sequence, the ST_REV parameter is always increased.

- **0X02: UPDATING,** the block is being updated, that is, the HART commands are being sent and the information is not yet valid for reading. The supervisory software should wait until the block becomes UPDATED. In the HVT block case, this process may take a few minutes since it depends on how many HART commands are sent to the equipment. For instance, if 60 commands are sent (in the FY301), along with the HIRT block polling commands the time will be approximately 2 minutes.
- **0X03: UPDATED,** if the commands were sent successfully, the block assumes this status. To ensure that the polling and the automatic update work properly, the block should be at this status. If the block is at another status, check the device configuration, the HART commands used, etc. Once the parameters reached UPDATED, all the data is available for reading and writing.
- **0X04: PARTIALLY UPDATED**, If the block is at this status, one or more transactions were not done successfully during the UPDATING process. Make sure the device supports all of the commands sent by HIRT block, otherwise, filter the not supported commands in the COMMON_CMD_FILTER.
- **0X05: NOT RESPONDING**, The device does not respond anymore, even performing all retries.
- **0X06: BYPASS,** The block has no valid information because the communication is in BYPASS mode (see COMM_BEHAVIOR in the HFG block).
- **0X07: DEVICE NOT FOUND**, It is only valid for HIRT blocks. During the Identification process, the device wasn't found by the command selected in the ID_CMD parameter
- **0X08: HCD ERROR**, There may be any configuration error. If it is the HVT block, go to the HIRT block where the device is and whose HART_TAG is written in the HVT.DEV_TAG_SEL.
- **0X09: TAG NOT FOUND**, It is only valid for HVT blocks. It indicates that no HIRT block was found for the HART_TAG from the HVT.DEV_TAG_SEL parameter.
- **0X0A: WRITING**, The block accepted the recording request and has started the transaction. Set it on OLD DATA and, then change to UPDATE.

BLK_ERROR and DEVICE_STATUS Parameters

The BLOCK_ERR is a parameter of fundamental importance for problem diagnosis both in the block operation and the associated HART device. For the HIRT and HVT blocks, according to the last transaction status with the device, the BLOCK_ERR may display the following messages:

STATUS HART bit	DEVICE_STATUS parameter	BLOCK_ERR parameter
2	Loop Current Saturated	Other
3	Loop Current Fixed	Local Override
1	Non-Primary Variable Out of Limits	Device Needs Maintenance Soon
0	Primary Variable Out of Limits	Process variable has BAD status
7	Device Malfunction	Device Needs Maintenance Now

Not all bits on the BLOCK_ERR parameter are associated with the HART device failure conditions and associated with the DEVICE_STATUS. Note the following:

- **OutOfService**: When the block is in OS mode, the BLOCK_ERR will present this bit already set, regardless of the last device or block operating condition.
- **ConfigurationError**: Indicates that there's possibly a problem in the configuration or in the HART commands of the HCD block selected on the HCD_SEL parameter. Check this parameter.
- **ReadBackFailed**: Indicates when the device stopped responding.

HIRT block Operation

Once the configuration is stored in the HI302, the user should write ENABLED in the COMM_ENABLE parameter in order to operate it. When the communication is enabled, the HI302 performs the following operations:

1. **BLK_EXEC_STATE = IDENTIFICATION**. HART device identification configured on each HIRT block. It is done by using the command configured in the ID_CMD parameter, which is located in the HIRT BLOCK. If after some RETRIES, the device does not respond, the BLK_EXEC_STATE parameter of the corresponding HIRT block goes to **DEVICE NOT FOUND**. Check the block configuration, the device channel, polling address, the TAG, the device, etc. If the ID_ CMD parameter is using command 0, a new identification attempt is performed after 30 seconds, polling the whole address space from 0 to 15.
2. **DEVICE_IDENTIFIED = TRUE and BLK_EXEC_STATE = UPDATING.** After the initial identification, the basic data are read and all corresponding parameters are filled.. The identification and the reading process last approximately 10 seconds per channel. Remember that, the 8 channels work normally in parallel, therefore, these initial processes last approximately 40 seconds for the 32 devices (4 devices per channel). See in the Appendix B, a list of all of the commands sent in this stage. During the UPDATING process, only the reading commands are sent.
3. **BLK_EXEC_STATE = UPDATED**. Once the data have been identified and updated successfully, the BLK_EXEC_STATE parameter goes to UPDATED, allowing access to the HIRT parameters. Thus, all devices are polled, the dynamic variable values and the device status are updated, if the HIRT.POLL_CTRL parameter is **Enabled**. The update frequency depends on the amount of HART devices in the channel and the number of HART commands sent on each polling cycle.

IMPORTANT
The POLL_CTRL parameter should be set on Enabled to allow for the HI302 to poll the device. In addition to updating the dynamic variables, the polling also updates the parameters that depend on the RESPONSE_CODE parameter, for example, DS-65 and DEVICE_STATUS. The polling is very important to detect device configuration changes made by other MASTERs like portable configurators, so, the HI302 can automatically update the data.

4. IF any error occurs during the updating process, the block should go to PARTIALLY UPDATED. See some reasons below:
   - The HART device does not support one of the sent commands. Check the commands using the device documentation (see the Appendixes). Since “Common Practice” commands are used, the user should filter them by using the COMMON_CMD_FILTER parameter.
   - There was an error during the communication. Check the HCFG diagnostic parameter to help you to solve the communication problem.
   - Set the block on OS and then, change it to AUTO for identification.
5. If the block is set on **UPDATED**, the user can associate the HIRT block with HVT block to read any device variable.
6. In normal operation, the alteration flag, located in the Response Code, is checked at each transaction. If this flag is set, the HI302 starts a reading sequence on the parameter of the HIRT block, that is, the BLK_EXEC_STATE parameter goes to IDENTIFICATION. The HVT block does not update automatically and its ST_REV parameter has no useful meaning. At the end of the reading process, the block should be updated and reflecting the device’s data. So, the ST_REV parameter of the corresponding HIRT block is incremented and will indicate that a parameter was changed.

HART Variable Writing and Reading

An operation model, based on reading cycles and writing requests, was designed to make the HI302 use as simple as possible.

HART variables Reading Cycles

The variable reading is always done in-group because of the HART command structure. Therefore, the HI302 variable readings are done through cycles, that is, the HIRT block always updates several parameters at the same time (except the writing). When the HVT block is executed, all of the reading commands from the device configuration are executed at the same time, following the configuration sequence. See the Appendixes.

When the HIRT block detects any change in the device (BLK_EXEC_STATE=OLD DATA), it starts an update process and sends again all the reading parameters of the configuration. The applicative checks if the ST_REV was changed and begins to execute the HVT block for the device.

Writing on parameters that map the HART variables

Unlike the reading, writing on parameters that map HART variables is done separately for each variable.

IMPORTANT
Not all parameters related to HART variables allow writing. Writing depends on the device's HART version. If the user has any doubt about writing on parameters, check the HIRT block documentation as well as the HART commands supported by the device.

Since some HART commands have variables for writing, the command will be composed of the parameter to be changed and the parameters described in the command definition (HCD block). As the parameter does not inform which HART command should be used to write (The writing and reading parameters are not the same), it is necessary an additional configuration for the parameters to be written. Such additional configuration is done in the HWPC block because it has the correlation between these parameters and the HART commands used for writing. The HWPC has the same identification code as the corresponding HCD, therefore, the entire configuration for a HART device should have a HCD + HWPC pair. A writing request starts a process with the following stages:

1. After the parameter had received the writing command, the HI302 checks the HWPC block to know whether that parameter can be written, that is, if it has a configuration to write. If the parameter has such a configuration, the HI302 stores the value received in a temporary variable and starts the writing process in the device. If the parameter does not have this configuration, the HI302 returns a standard response indicating that the parameter cannot be written.

IMPORTANT
In case the parameter does not accept writing, check if the HWPC configuration is correct.

2. If the parameter can be written, the writing request is started. The writing request has priority over HART transactions since the block is set on UPDATED. The block status, in this case, goes to WRITING.

3. As soon as possible, the writing command will be set according to the definition in the corresponding HCD block and will be sent to the device.

IMPORTANT
Despite the fact that the HI302 has responded and the writing was successful, the value read on the parameter continues to be the old one. The parameter will only be updated after the writing confirmation in the HART device.

4. In case the device does not respond, there will be attempts according to the schedule in the HCFG.RETRIES[CHANNEL] parameter. If there is any message saying that the writing was not performed, the HI302 will ignore the writing request. Even though the device does not respond to the writing command, the HIRT block will always return to UPDATE.

5. If the writing command response is positive, the block parameter is updated with the new value written in the device. Then, the monitoring will read the new value and finish the writing process successfully. The block status will go to OLD DATA and then, to UPDATED. Then, the ST_REV will be incremented.

6. Note that the time for a writing request can vary a lot:

   - If the channel is free and the writing request was immediately processed, the transaction will last approximately 3 s, provided there is no communication error due to the time to send the writing command, read the value and send the command to reset the device's alteration flag.

   - If the channel is busy, the request will wait until the next time to be transmitted. A long timeout is needed, if the transaction that is using the channel has several retries and each retry has many repetitions. So, the user should monitor the BLK_EXEC_STATE parameter.

IMPORTANT
This procedure indicates that the user's application should monitor the BLK_EXEC_STATE parameter. This parameter changes to WRITING, then to OLD DATA and finally it goes to UPDATED provided the writing was done successfully. At the same time, the parameter value (after the block has returned to UPDATED) should be compared to the written value. So, the operation will be confirmed or not.

Operating the HVT Block

The HVT block is an extension of the HIRT block, because it makes it possible to access all of the variables related to common or specific commands. After identifying the devices, the HART variables that aren't in the HIRT block may be accessed. To do this, one must know the allocation map on the HVT block that connects the HART variables on the device with the block parameters, whose names are generic ones. The devices can be fully defined using the HIRT+HVT blocks.

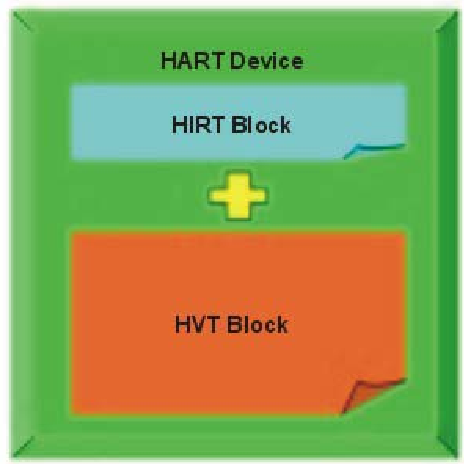

Figure 5.1 – Device definition using HIRT + HVT blocks

See below an example of a HVT allocation map for Smar FY301 intelligent positioner:

Index	Parameter Name	HART Variable Name	HART Command	Description
11.1	U8B_ARRAY_1[1]	EEPROM_CONTROL	39	Saves data from RAM to EEPROM or recover data from EEPROM to RAM
11.2	U8B_ARRAY_1[2]	DISPLAY_CONNECTED	128	Indicates if the display is connected or not
11.3	U8B_ARRAY_1[3]	AIR_TO	128,129	Indicates if the valve needs air to open or to close
11.4	U8B_ARRAY_1[4]	LOCAL_KEYS_MODE_CTL	132	Indicates if the local adjustment is enable or not
11.5	U8B_ARRAY_1[5]	COORD_GROUP_NUMBER	133	Select one coordinate to be read from the characterization curve
13.14	U8B_ARRAY_3[14]	MANIPULATE_VARIABLE_UNIT	216	MV unit
13.15	U8B_ARRAY_3[15]	SETPOINT_TRACKING	216	Not available in FY301
13.16	U8B_ARRAY_3[16]	SERVO_PID_MODE	216	Indicates if the servo-pid is enable or disable
16.7	FLOAT_ARRAY_1[7]	PRESSURE_LOW_LIMIT	244,245	Lower pressure limit for the input sensor
16.8	FLOAT_ARRAY_1[8]	POS_VALVE	168	Valve position
16.9	FLOAT_ARRAY_1[9]	TRAVEL_DEADBAND	189,190	Travel deadband value
16.10	FLOAT_ARRAY_1[10]	TRAVEL_LIMIT	189,190	It is the maximum distance covered by the valve before an alarm is generated
29	String_06	ACTUATOR_ID_NUMBER	183,184	Actuator identification number
30	String_07	VALVE_ID_NUMBER	185,186	Valve identification number
34	String_11	ORDERING_CODE	173,174	Factory device information

This way, if the user wants to read the AIR_TO variable, he must read the U8B_ARRAY_1[3] parameter. As there is only a single HVT block shared by all of the devices, it use must be triggered, that is, one must choose first the device that will use the HVT block, and then shoot the reading cycle or a writing operation.

Sequence for HVT Reading Cycle

After defining, executing and loading all of the configurations in the HI302, it is now possible to communicate with the devices using the HVT block. The respective HIRT block should have already been identified.

The HVT block operation is very simple and its use must comply with the following rules:

1. The user's application must check if the block is available by reading the BLK_EXEC_STATE parameter. If the parameter is set on UPDATING, the block is in operation and cannot be used for a new transaction.
2. Write on the DEV_TAG_SEL parameter the 8-character TAG of the HART device that one wishes to access. It must be the same as on the HIRT.HART_TAG parameter;
3. After writing on the HART_TAG, the block automatically checks if the command configuration desired is in the memory. Otherwise, the block goes to HCD ERROR.
4. If there is one, the HVT block checks if the written TAG belongs to a HIRT block that corresponds to a device installed in one HI302's channel. If the TAG was not found, the status will be TAG NOT FOUND and the process will be aborted.
5. If the HCD block and the HART_TAG were found, the parameter goes to UPDATING and the communication process with the HART device begins. Thus, all of the reading commands will be sent according to the sequence in the configuration.
6. If the device does not respond at the first command, the BLK_EXEC_STATE parameter changes to NOT RESPONDING.
7. If the device responds at the first command, the status will continue on UPDATING until the last reading command is sent. If there is an error during the communication, the BLOCK_ERR will report it and the process will not stop until the last command is sent. The block status will be PARTIALLY UPDATED and the user should identify the problem, which may be a command configuration error or a non-compatible version.
8. In case the process finishes with no error, the block status will be UPDATED. Thus, the applicative can read all of the block's parameters according to the allocation map and execute writing operations as well.

Writing Sequence in HVT Block Parameter

In order to write in a HART variable that is mapped as the HVT block parameter, the user should first perform an updating, that is, a comprehensive reading operation. When the HVT block is UPDATED, follow the same procedure used for writing on HIRT blocks.

IMPORTANT
All the HVT parameters accept reading and writing. However, only the parameters that have writing configuration will generate a HART transaction. The device should support the command configured, even if the HI302 has started a writing transaction. So, if any writing error occurs, check the HVT allocation map (Appendix C) to make sure that the parameter is mapped for a HART variable that can be written.

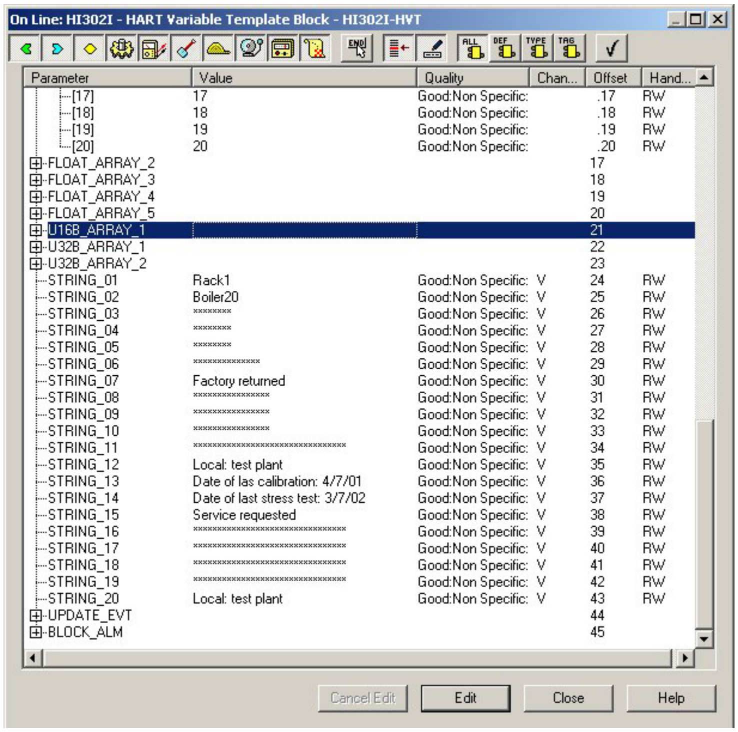

Figure 5.2 – Writing in HVT block parameters

HI302 versus Portable Programmers (Field Alterations)

The HI302 monitors every device that is working normally in a continuous way and whose HIRT block is in UPDATED with POLL_CTRL=ENABLED. Each HART master channel, when configured as PRIMARY by the HCFG.MASTER_TYPE [CHANNEL], permits the simultaneous use of a portable programmer, for example, the Smar HPC301. If some alteration happens on the device, it will report this to the HI302 in its next transaction, through the specific bit in the RESPONSE CODE. Then the HI302 will take the following actions:

- Will set the block status on OLD DATA (BLK_EXEC_STATE);
- Will send the HART command 38 to reset the flag that indicates that the device's configuration was changed.
- Will start reading all of the parameters on the HIRT block. After doing this, the HI302 changes to the UPDATED status and will increment the corresponding HIRT.ST_REV, indicating on the user's applicative that the data must be read again, possibly with the HVT execution for this device.

STATIC REVISION (ST_REV) Parameter

Usually, the ST_REV is only increased if a parameter classified as STATIC, is changed. Therefore, the HIRT and HVT blocks do not work that way. Other than the static parameters, any other parameter that maps the HART variable will increase the ST_REV of the corresponding HIRT block.

The ST_REV parameter from the HVT block do not mean anything for the moment; what happens while the HVT parameter is being written is:

- If the writing is accepted, a HART transaction is generated, making the HART device indicate the change through the RESPONSE CODE. The parameter ST_REV from the HIRT block associated to the HVT block will be increased.
- If Polling is enabled, the HIRT block associated to the HART device will notice this change and will automatically be set to OLD DATA, sending the command 38.

- After the command is sent, the HIRT block will start updating the parameters, being set to IDENTIFICATION and UPDATING. During the UPDATING process, the ST_REV from the HIRT block will be increased indicating that the data were updated and should be read again.

IMPORTANT
The polling commands do not increase the block's ST_REV.

Conclusions on the STATIC REVISION:

1. The client application should only monitor the parameters ST_REV from the HIRT blocks, ignoring the ST_REV from the HVT block..
2. If the application writes on the HVT block, it may ignore the increment done on the HVT block ST_REV, as well as the corresponding HIRT block increment.
3. If there is no writing on the HIRT block and the ST_REV was changed, the device information was updated and a new reading should be done in the HIRT and HVT blocks.

HART RESPONSE CODE Conversion to STATUS FF

The RESPONSE CODE of the last transaction with HART devices is mapped to the DEVICE_STATUS and COMM_ERR parameters of the HIRT block. In order to use the information about the device's general status, the DEVICE_STATUS is converted into a corresponding STATUS FF to feed the parameters that have STATUS. For example, the PV_MA can be separated into PV_MA.STATUS and PV_MA.VALUE. The PV_MA.STATUS represents the actual condition in which the device provides the PV_MA.VALUE parameter. See the table below:

STATUS HART bit	DEVICE_STATUS parameter	Status Foundation Fieldbus converted
7	Device Malfunction	Bad:DeviceFailure:NotLimited
0	Primary Variable Out of Limits	Bad:UncertainNonSpecific:iNotLimited
2	Loop Current Saturated	Uncertain:SensorConvNotAccurate:NotLimited
1	Non-Primary Variable Out of Limits	Uncertain:Subnormal:NotLimited
3	Loop Current Fixed	GoodNonCascade:GoodNCNonSpecific:Constant

- If the block is set at OS, all the parameter will assume the **Bad:OutOfService:NotLimited** status.
- If the device stops the communication and the block status goes to **NOT RESPONDING**, all the status go to **Bad:NotConnected:NotLimited.**
- If there is no error or alert condition to be informed, the status will assume the following status: **GoodNonCascade:GoodNCNonSpecific:NotLimited.**

"BYPASS" Mode

The BYPASS mode was created to make it possible for the HART messages to be sent to any device, through simple writing on a parameter called BYPASS_REQ_N (N is the channel number, from 1 to 8). Every parameter is a chain of 100 characters where the HART messages must be written to be sent to the device. The written content on the parameter is entirely sent to the channel configured on the HIRT block, as soon as the channel is free. Thus, the application must include in the message the preambles, the delimiter and so on. Only the RESOURCE and the HCFG blocks are necessary to operate the BAYPASS mode. Any HIRT or HVT block instantiated will stay on BYPASS status (BLK_EXEC_STATE parameter).

IMPORTANT
To use the BYPASS mode, the user should write BYPASS on the HCFG.COMM_BEHAVIOR parameter.

The HCFG.BYPASS_STATUS indicates the situation of the message and may assume the following values:

- **IDLE**, the channel may send a message using the BYPASS_REQ_N parameter or there is a message available to be read through the BYPASS_RES_N parameter. Even though, the channel is monitoring another master's communication or a device in burst mode, it will be available to send a message.
- **BUSY**, the applicative ordered a REQUEST that is being executing,
- **TIMEOUT**, after a predefined number of repetitions, it was not possible to receive a valid message for the request.
- **RESPONSE AVAILABLE,** after the programmed number of repeats, it was not possible to receive a valid response for the sent request .

The HI302 does not perform any integrity or content checkup on the message received. It transmits what it has received to the communication channel. The applicative must ensure the quality of the messages and the response interpretation.

Sequence for Sending a HART message through a BYPASS

- Check if the HCFG.BYPASS_STATUS[N] parameter is on IDLE or TIMEOUT. If positive, the message can be written on the HCFG.BYPASS_REQ_N parameter. The HI302 will check if the channel is available and transmit the content of the HCFG.BYPASS_REQ_N parameter.
- While the HCFG.BYPASS_STATUS[N] parameter is on BUSY, the HI302 is waiting the response or repeating the request up to the limit programmed in the HCFG.RETRIES[N].
- The HCFG.BYPASS_STATUS [N] parameter goes to IDLE if it has received a valid message. It goes to TIMEOUT, in case it hasn't received a valid response yet.

IMPORTANT
As the message has to be entirely supplied by the applicative writing on the HCFG.BYPASS_REQ_N, the applicative will guarantee that the message is in the right format, with the right address, etc.

Chapter 6

Basic Functioning Theory

This chapter presents basic concepts on HI302 hardware to help the user to solve field and maintenance problems.

The HI302 Block Diagram

The picture below shows the block diagram for the HI302:

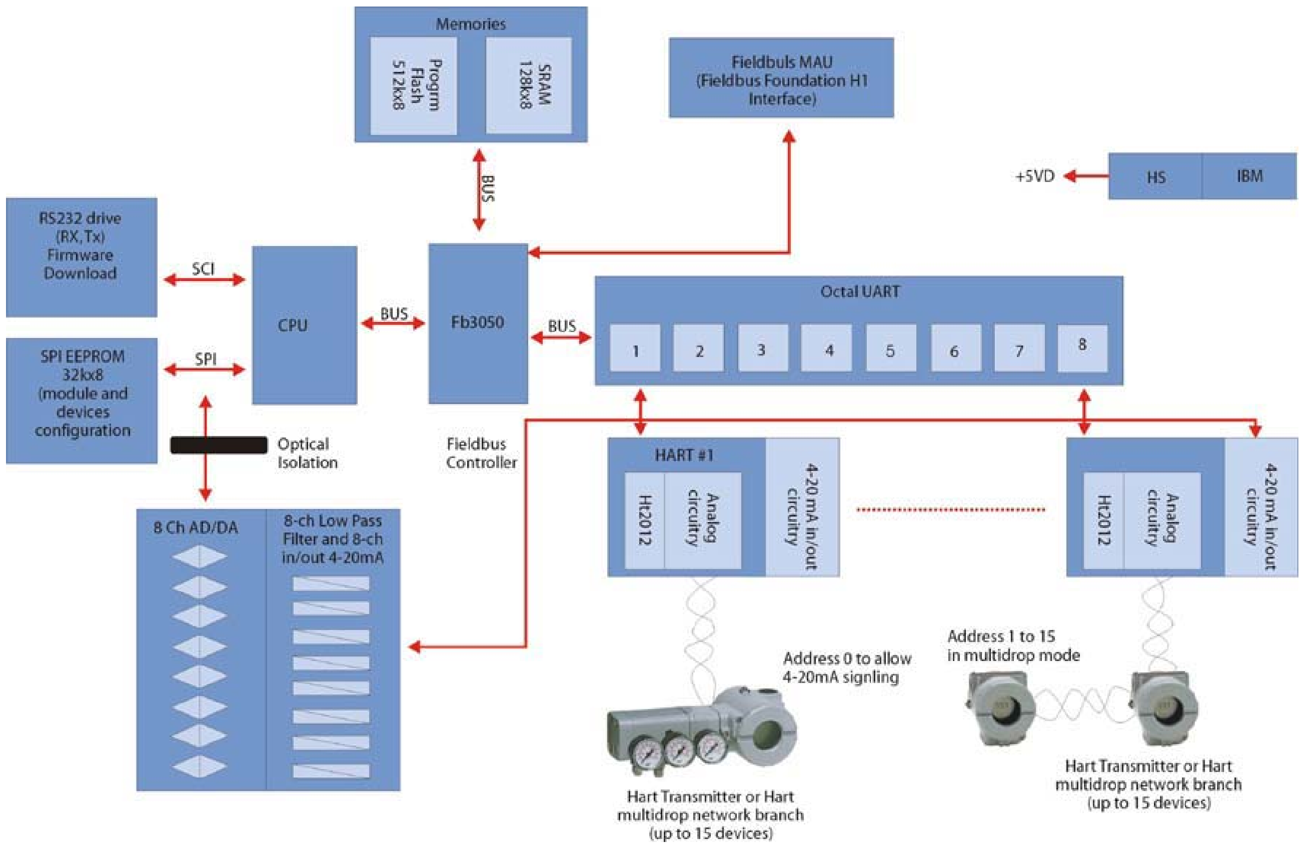

Figure 6.1 – HI302 block diagram

Hardware

This chapter presents a summarized description of the main functional blocks that make up the HI302 module, as a support for electricians and/or maintenance technicians. See below the GLL1193 main board pictures.

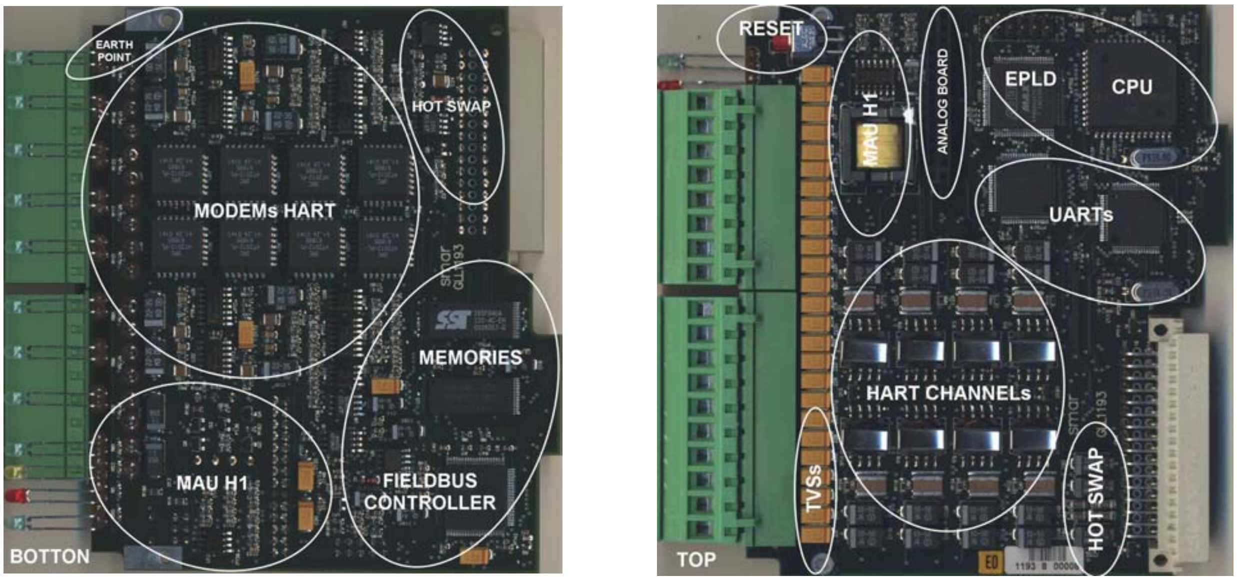

Figure 6.2 – GLL1193 Main boards' picture

Power Supply, Operation Voltage and Protection

HOT SWAP

The HI302 is equipped with a U22 HOT SWAP controller that allows the insertion or removal of the module with the power supply on. This is important when the HI302 is being supplied by a shared source, e.g., a rack with other HI302 or DF51 modules. In addition to a 5V source, the HI302 should receive a failure signal from the source (/PFAILR), working at a low level. That signal indicates to the HI302 that the feeding has been stopped and the starts a safety procedure for failing feed. The Smar PS-AC and DF50 sources also have this signal. The HOT SWAP controller also monitors the power consumption and, if a problem arises, the feeding is switched off to avoid further damages to the HI302 and the power supply.

ATTENTION
If the HI302 is supplied by a source without the failure signal, the pin should be left open.

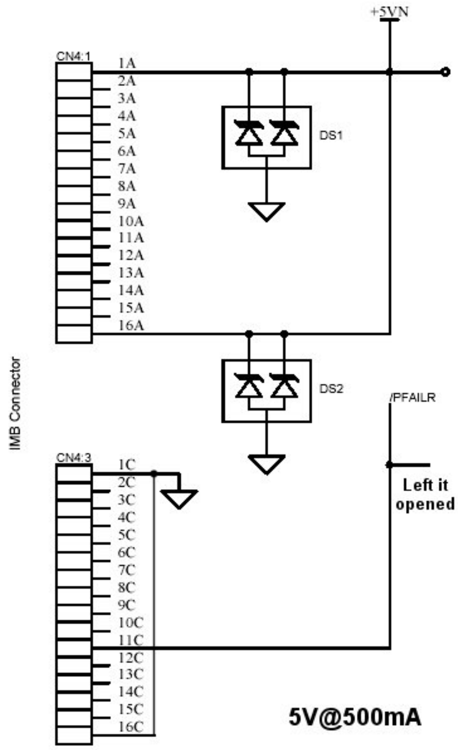

Figure 6.3 – HOT SWAP controller

Regulators

The HOT SWAP controller receives +5VN voltage from the rack (or external source) and, through the MOS channel-N transistor, delivers +5VD to all digital circuits. A 3V3 (U11) regulator feeds the EPLD (U10). And there is yet a precision 2V5 (U9) power source of reference for the virtual ground cable of the operational amplifiers in the Fieldbus and HART communication circuits.

In the HI302-I and HI302-O modules, there are also a DC-DC (U1) converter and the optical couplers that completely isolate the ground cables of the analog and digital circuits. The DC-DC converter generates, from the +5VD, a non-regulated voltage called +VA (+9V) that is used by the analog inputs (HI302-I), and through a linear regulator generates a +5VA voltage for the other circuits (ADC and DAC). As a voltage reference for converters there is another 4V096 (U8) precision reference.

Protection

The HI302 has advanced specific components for protection against ESD and voltage surges. However, keep in mind that good ground wiring is fundamental.

Figure 6.4 – ESD protection

Electrostatic Discharge (ESD)

The power input (DS1 and DS2) and all the communication channels are protected against electrostatic discharges by specific components (zener diodes) with high response speed and good power absorbing capacity (24W @ 1ms @ 25 ^{o}C). In addition to this protection, there are sparklers built on the printed circuit board, on every LEDs and on all (CN5 and CN6) front connector pins. They are designed to sparkle with 200-300V voltages, depending on the air humidity.

Surges, High Voltage and Grounding

Voltage surges, whether inducted or conducted, occur often in industrial environments due to electric motors, frequency inverters semiconductors switching and many other factors. Besides, there is always the danger from atmospheric discharges that may permanently damage the equipment. HI302 modules have components highly efficient against these surges, with high absorption capacity (400W @ 1 ms @ 25^{o}C) and a (typically < 1 ns) good response speed, in addition to the PCI built-in sparklers. However, this protection is practically useless without a good wire grounding.

ATTENTION
Indispensably, a good quality grounding (< 5Ω) must be connected to the system, either on the DIN rail, on the power source, or directly to the module grounding terminal. Without this, all protection for the HI302 module, as well as for other equipment, WILL NOT WORK. Note that grounding resistance must be measured every 2 years.

Processing Core

The core of the HI302 module is based on the 68HC11 (U1) microcontroller running at 16 MHz (Y1): one of the most reliable and well accepted components worldwide. Around the microcontroller, there is the Fieldbus Smar FB3050 (U2) controller. Also there are the 128x8 (U3) SRAM memories and the 512kx8 (U4) FLASH. And to store the module configuration for the functional blocks and the HART commands there is a 32kx8 (U6) synchronous serial memory. A (U5) reset circuit/voltage supervisor is also part of the equipment.

Firmware programming

Programming the firmware in the FLASH memory is done through a RS232 serial interface (U2, on the backup board) by using the bootloader mode on the HC11. The connector is accessed in the module's lower part, and it can be done by user, if the programming cable and the FBTools, basic to the System302, are available.

Figure 6.5 – Firmware programming connector

To reduce the amount of components, increase reliability and make the hardware more flexible, a EPLD, U10 programmable logic device has been used, including several functions with combinational and sequential logic, such as frequency divider for HART modems, address decoders, etc. This component is supplied with 3V3 and has 5V-compatible inlets and outlets. One of its most important functions is to generate the clock for the HART modems, stemming from the UART clock.

EPLD Programming

Update the EPLD programming in the circuit is also possible, due to the continuous improvement in the equipment, though it should only be done in the factory.

Manual RESET key and FACTORY INIT

In the front of the HI302 module there are two push button keys. One of them is the Hand RESET button, connected to the U5, that covers the following components: HC11 (U1), FB3050 (U2), EPLD (U10) and UARTs (U7 and U8). The other key, interrupts the HC11, which initializes the memory and brings the whole configuration back to the factory standard. This option is very useful in case of malfunctioning due to configuration errors, as it reinstates the configuration memory in its basic functional status.

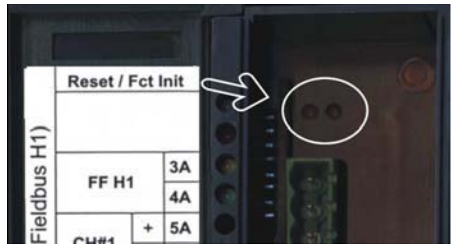

Figure 6.6 – Fct Init

ATTENTION
Do not press the RESET button when the HI302 while the factory init process is taking place. Otherwise all the configuration will be lost. Also, pay close attention to not push the Fct Init button instead of the RESET button.

HI302 Module Resetting

The HI302 module has a very sophisticated resetting system, consisting of a circuit that supervises the U5 supplying voltage and of several built-in mechanisms on the HC11 microcontroller, such as the WATCH DOG clock monitor. In addition, the RESET is used by the U10 EPLD to adjust the firmware programming (resident program). When starting up the module, the EPLD begins the device's normal operation. If the RESET button is pressed, the module enters in the BOOTLOADER mode, and is ready to update the resident program on the FBTools. When the RESET button is pressed again, the module gets back to normal operation.

IMPORTANT
From what was explained above, press the RESET button twice to get the HI302 RESET.

H1 Fieldbus Communication

The Fieldbus communication is done through the FB3050, that access directly the memory via DMA and shares the 10 MHz clock generated by the HC11 (U1). The analog interface is done through the Media Access Unit (MAU). This is a passive interface, i.e., it is not fed by the net, with capacitive discoupling and galvanic isolation. It is protected against electrostatic discharges and voltage surges. All circuits are fed with +5VD, and some operational use the 2V5 reference.

The communication is indicated by a LED fed by the (U10) EPLD, as the LED blinks at every message transmitted.

HART Communication

The HI302 modules have 8 channels for HART master communication, with capacitive and galvanic isolation (TF3). All channels also are protected against ESD (D14) and surges (Z2, Z3 and Z5). The initial stage on each receptor is a low pass filter (U23A) centralized in approximately 1700 Hz, as the range of HART communication frequencies is from 1200 to 2200 Hz (fundamentals). Next, a comparator (U27A) changes the signal received and amplified in a square wave, for it to be decodified by the Smar HT2012 HART modem (U35). The transmission is made by switching a 22uF electrolytic capacitor that works integrating the signal modulated by the HT2012.

Every HART channel has a LED (L1), controlled by the UART, indicating the several statuses of the channel's work. Two quadruple UARTs (U7 and U8) totaling 8 independent UARTs concentrate the 8-channel communication.

These UARTs generate a signal of 14.7456 MHz (Y2) used by them and by a divider on the EPLD (U10), that generates the clock for the modems, 460,80 KHz strong.

4-20mA To Foundation Fieldbus Analog Conversion (HI302-I)

The HI302-I model has an additional circuit (GLL1205 backup board) that converts the transmitter's analog signal by means of a signal conditioning circuit and a low pass filter with a cut frequency of approximately 10Hz (U9A), an AD converter (U7) and a referential voltage source of 4V096 (U8). These circuits are protected against high voltage by zener diodes, against overheating and inverted polarity. This backup board is connected by the CN2 and allows the module to indicate the current's measured value by a MAI (or AI) block, to be utilized on any control mesh. All circuits forming the analog acquisition are isolated from the others by a DC-DEC (U1) converter and by (U3, U4...) optical couplers. The reading update is approximately 200 ms.

Figure 6.7 – Signal conditioning and low-pass filter circuit

Foundation Fieldbus to 4-20mA Analog Conversion (HI302-O)

In a similar way, a board with analog output circuits is capable of controlling a current used by actuators, for example. These circuits are protected against high voltage by zener diodes, against overheating and against inverted polarity. The output module uses circuits required by current control (UX), in addition to a (UXX) DA converter and a referential 4VO96 (U8) voltage source. These circuits make up 8 independent channels for current control working in parallel to HART digital communication. This backup board is connected via CN2 and allows for the module to control the current's value by means of a MAO (or AO) block, utilized on any control loop. All circuits forming the analog acquisition are isolated from the others by means of a DC-DC (U1) converter and (U3, U4...) optical couplers. The output backup board informs if the current loop is open or not, through the circuit formed by UX. The output update period is approximately each 200 ms.

Chapter 7

An example of HI302 usage

There follows an example of how the HI302 can access the variables of the Smar LD301 intelligent pressure transmitter. From this example, users may create more complex configurations. For further details, see the other chapters of this manual.

Installation

Below see how the transmitter was connected to the HI302, by using the resistor inside the HI302-I:

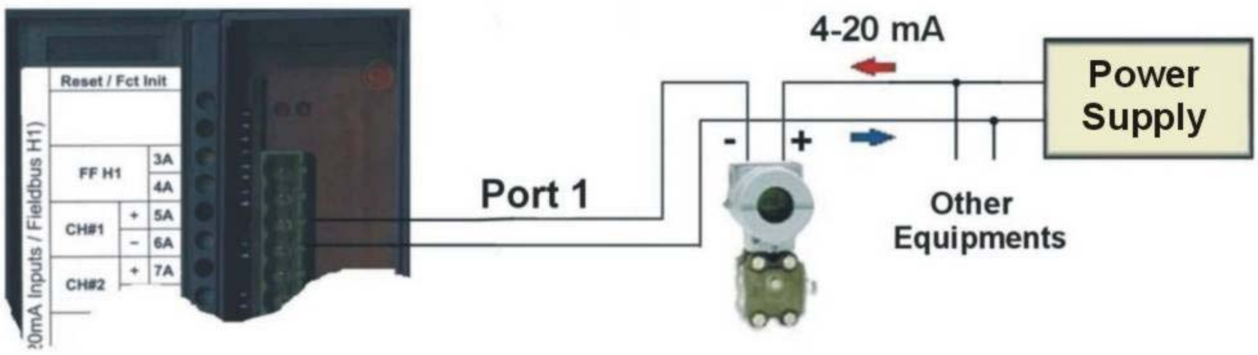

Figure 7.1 – HI302-I application example

Step by Step Configuration

1. When the HI302 is fed by the rack and is switched on, press the Fct Init button to initialize the factory configuration. The device will reset and then light the yellow LED SAVING and all LEDs on the HART channels. HI302´s current configuration will be lost!

2. Next, if the HI302 is connected to the Foundation Fieldbus wiring, the green H1 LED will start blinking quickly, indicating that the communication started. If it doesn't blink, it may possibly be some problem with the FF bus. The first time the HI302 is ever connected to the linking device (DFI302), wait 2 minutes until the HI302 is on the Live List.

3. Open the FF configurator (Syscon in the example) and notice that the HI302 is on the Live List, according to the screen below:

4. Create a configuration that includes one RESOURCE block, one HCFG block, one HIRT block and one HVT block. Then, assign a Tag to the HI302, so that it assumes the same TAG of our configuration test, as follows:

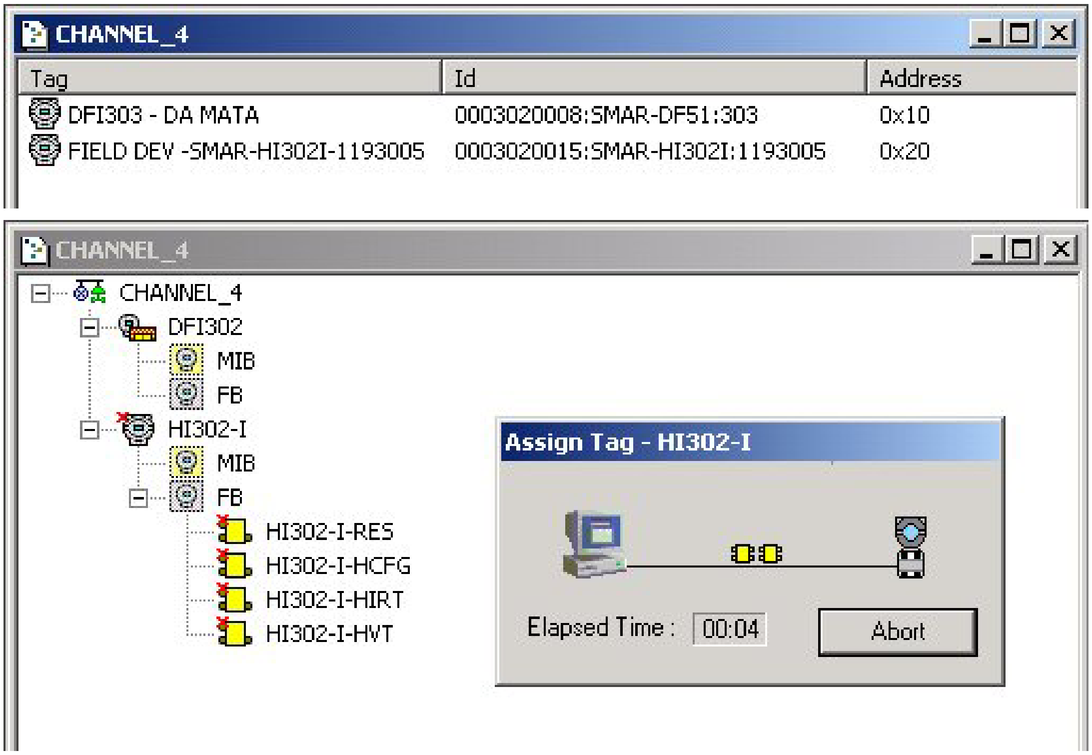

Figure7.2 – HI302 configuration

5. Let's now fill up the basic parameters on the HIRT block so that the HI302 communicates with the HART device. This minimum configuration makes it possible for the HI302 find the device and start the communication. In fact, let's see:

- **MODE_BLK.TARGET = AUTO**, so that the block starts right away executing the mode. Configure this parameter on the HVT block, too.
- **HART_CHANNEL = 1**, must be the same physical channel where the device is installed.
- **POLL_ADDR = 1**, assuming this is the device's address, previously configured.
- **ID_CMD = 0 – Polling address**, as the polling address is to be used.

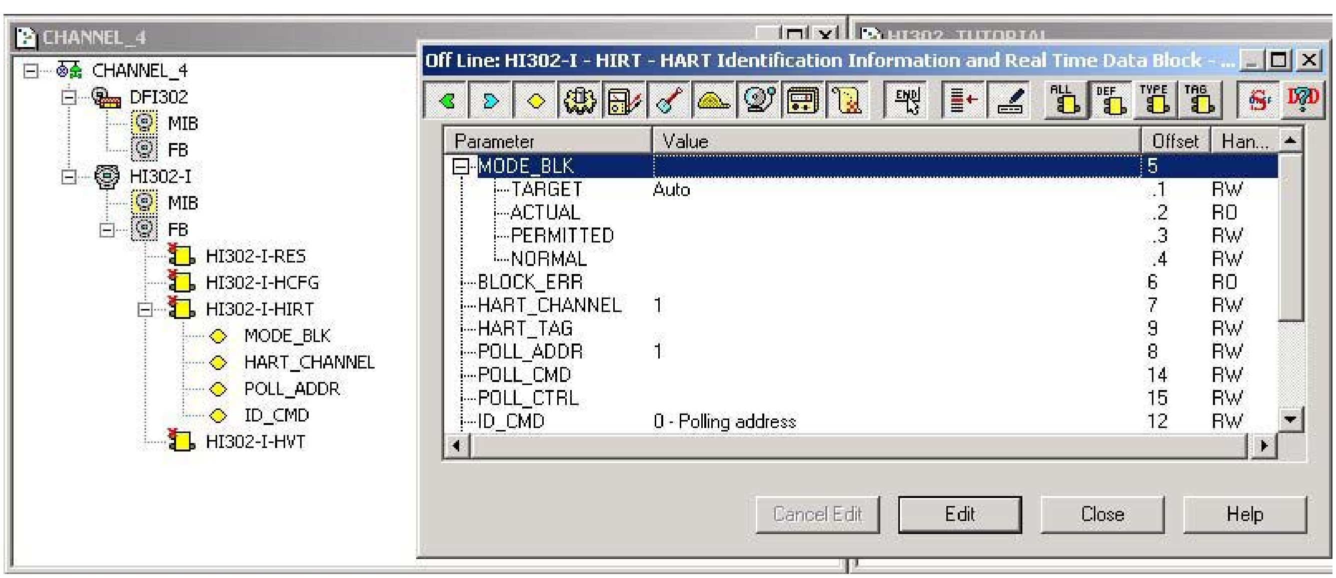

Figure 7.3 – HIRT block parameters

6. After filling up the essential parameters, just Download the configuration. See the screen below:

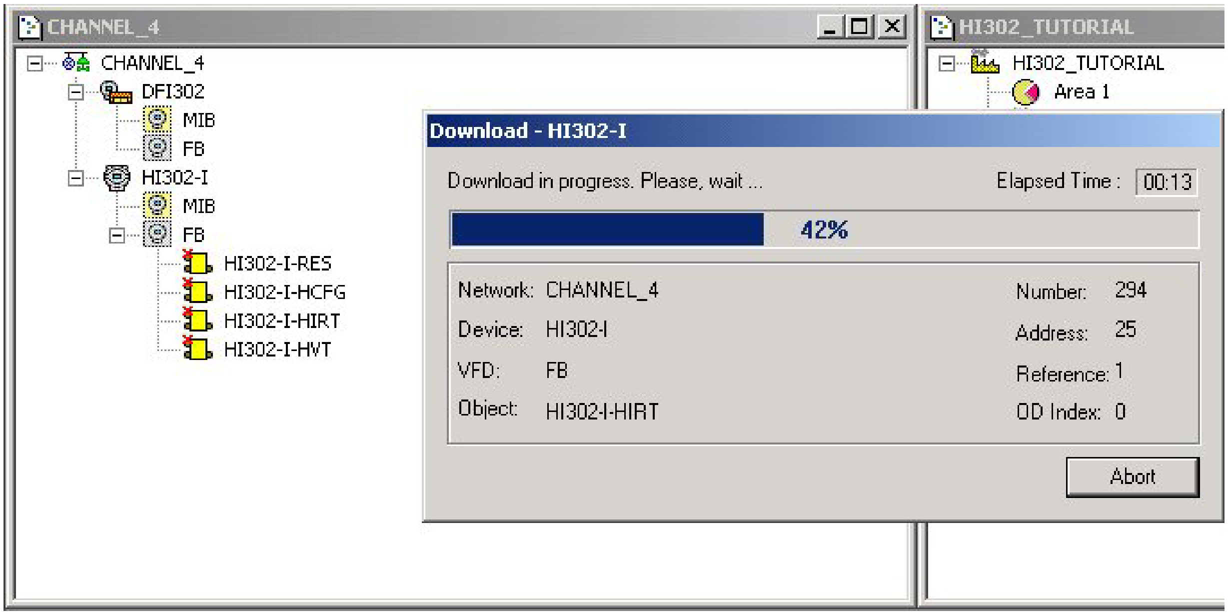

Figure 7.4 – Downloading the configuration

Step by Step Operation

7. After executing the download, the HART communication must be started manually by writing **ENABLED** on the COMM_ENABLE parameter in the HCFG block

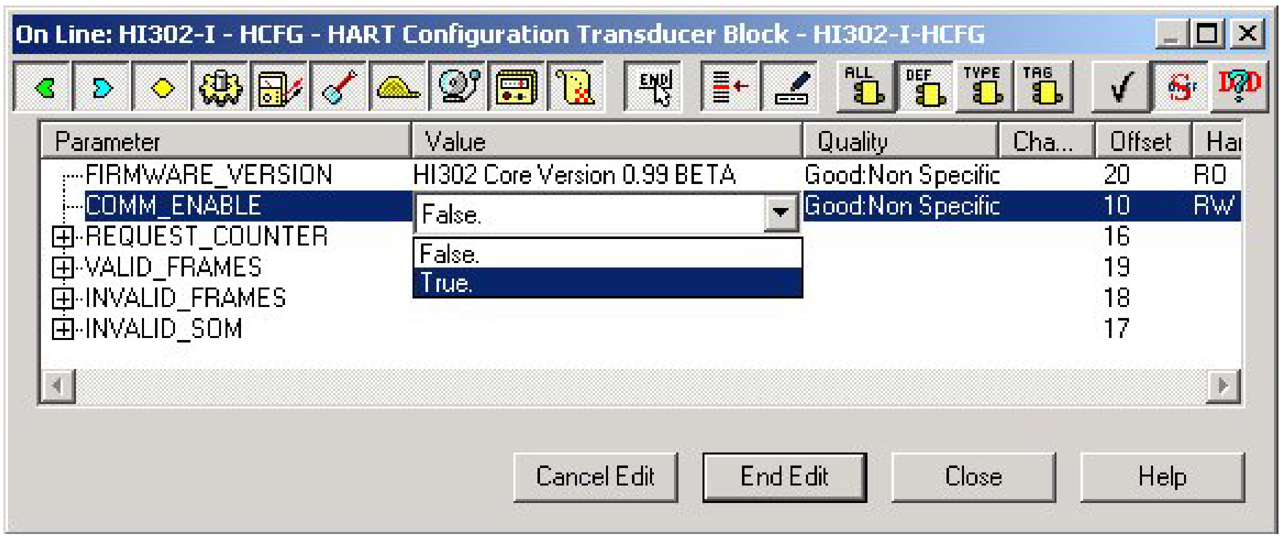

Figure 7.5 – HCFG block

8. The HART communication should start immediately after the parameter changes to ENABLED. This may be easily checked by the irregular blinking in the green CH#1 LED, according to the HART transactions. Note that the CH#2 and CH#8 LEDs blink at the ¼ Hz frequency, to indicate that there is no HIRT block using the channel.

9. See that the BLK_EXEC_STATE parameter, from IDENTIFICATION, moves to UPDATING. If not, check if the block is being executed. The MODE_BLK.ACTUAL parameter should be AUTO.

10. After identifying the device and filling up the HIRT block, if the polling is enabled in the parameter POLL_CTRL, the HI302 will continuously send the HART commands according to the view selected on VIEW_SELECTION parameter,. See that the BLK_EXEC_STATE parameter must be on UPDATED to that matter. If not, the initial sequence of HART commands wasn't executed successfully and there must be some configuration or device problem. Watch carefully the parameter values and check if they correspond to what they are supposed to be, according to the device installed. Then, all the values available in the block may be read.

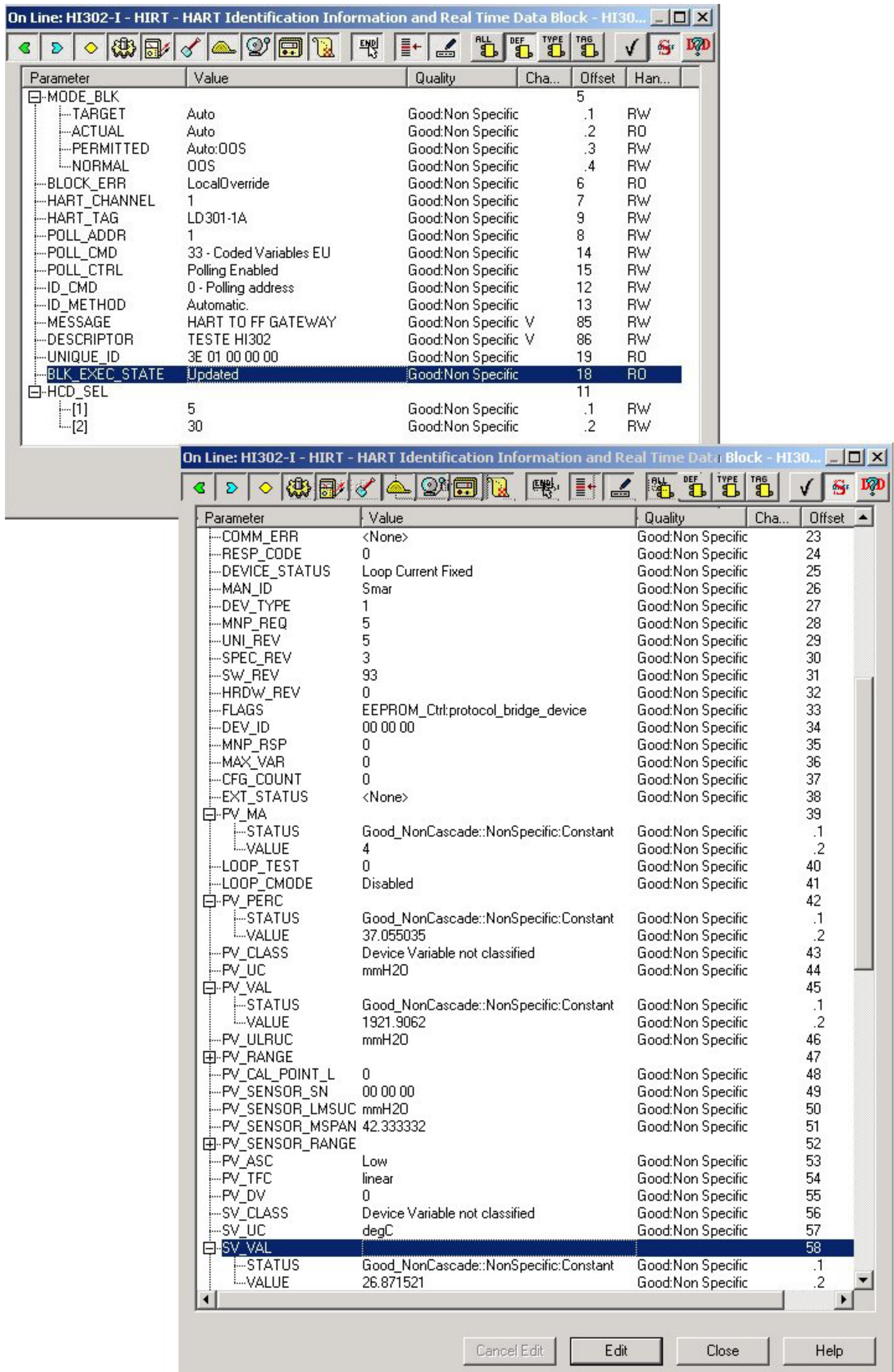

Figure 7.6 – HVT blocks

11. When on UPDATED, just proceed as if with any common parameter, in order to write it in any parameter that accepts writing, as for example, MESSAGE. Note that the BLK_EXEC_STATE parameter passes by three states: WRITING, OLD DATA and UPDATED. In normal conditions, the block must always return to UPDATED, to indicate that the data in the block are the same ones as the values in the HART device.

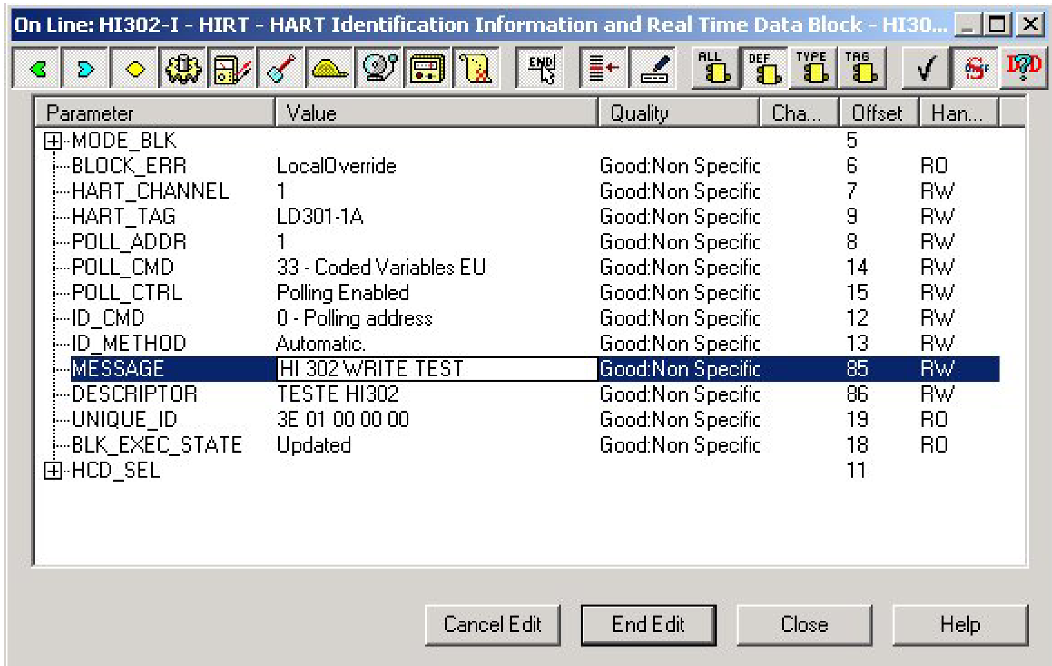

Figure 7.7 – HIRT block

12. After the device related to the HIRT block has been identified, it is also possible to read and write on the specific variables, **by using the HVT block**. The HVT block must also have both the MODE_BLK.TARGET and MODE_BLK.ACTUAL = AUTO. Then, just write on the HART TAG, in the DEV_TAG_SEL (**DEV_TAG_SEL = LD301-1A**) parameter, to start reading the specific variables in the HVT parameters. Refer to the allocation map from the HVT block to the device, so that you can relate the HVT parameters to the variables on the HART device. The allocation maps for all equipments with the configuration built-in the HI302 are shown in the Appendixes section.

13. If the written HART_TAG is correct and the same one read by the HIRT block, the HVT block starts the update, by sending all the reading commands in the configuration, to the LD301. See that the BLK_EXEC_STATE parameter changes to UPDATING and then goes to UPDATED at the end of the process, which lasts about 2 minutes. Moreover, it shows on the **HCD_DEVICE_INFO** e **HCD_SELECTED** parameters information about the selected configuration.

14. To make a writing, the process is the same as for the HIRT. The next figure shows the HVT being updated.

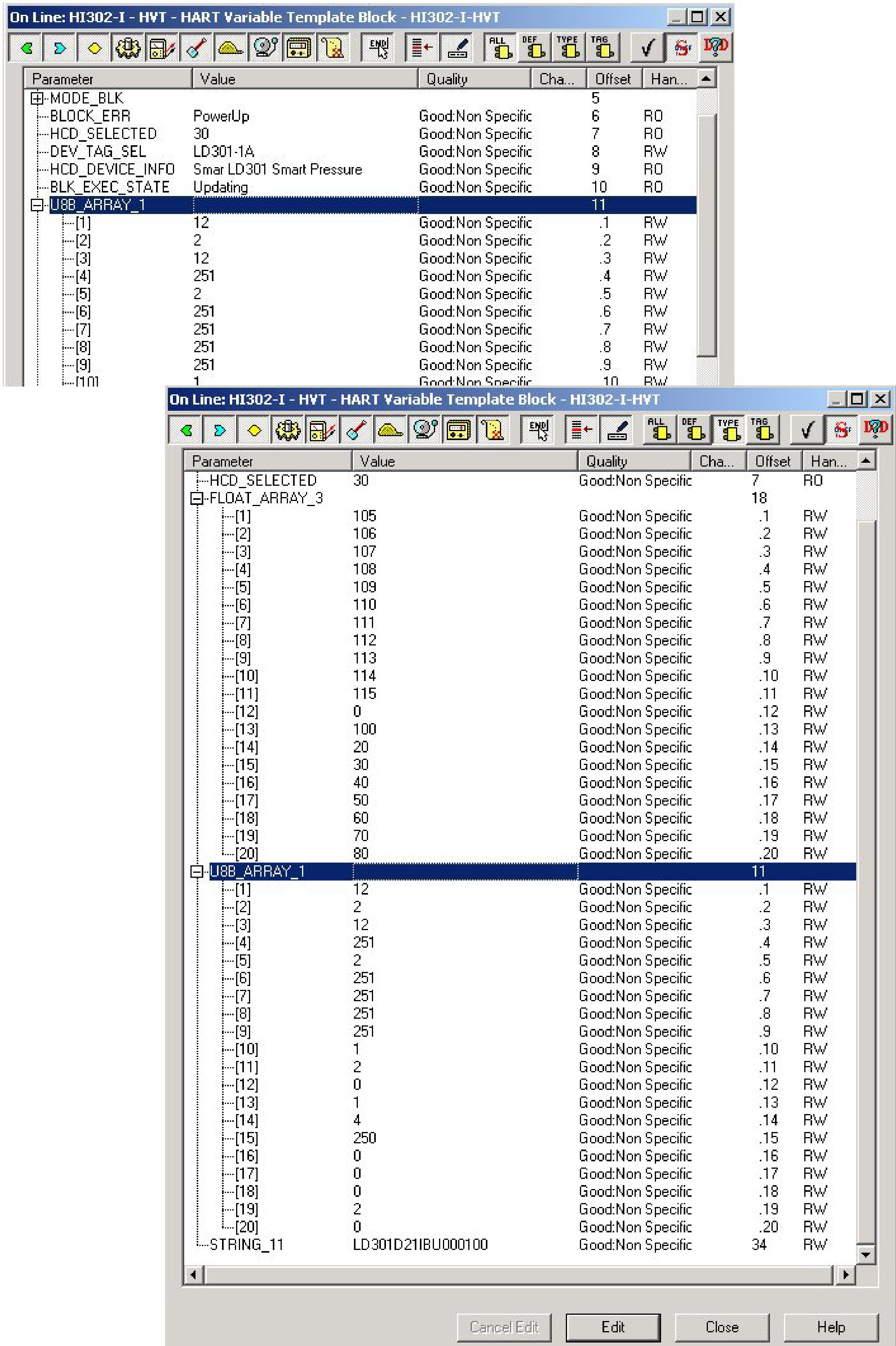

Figure 7.8 – Updating the HVT

Chapter 8

Troubleshooting

Installation

Problem	Solution
When to use the 250Ω resistor in series with the loop.	• Increase the impedance measured from the HI302 terminals if it is less than 250Ω. Maybe, it is necessary to increase the supply voltage. • See the "Impedance x Minimum Supply Voltage" graphic in the Chapter 2.
When to use the active impedance PSI301P.	• The active impedance should be used when the total loop current prevents the use of a resistor. This happens when using a multidrop connection with the device's analog output enabled (varying from 4 to 20 mA) or the number of devices is usually more than 4 even in fixed current mode.See an example in the Chapter 2.
The instrument is correctly installed; it works normally, communicates with a portable programmer, but doesn't communicate with the HI302.	• Place the terminals of the portable programmer on the corresponding channel of the HI302 and check whether it communicates correctly. If it works, it may be a problem with the HI302. If not, check the equipment and the installation. • This problem may happen especially with actuators (a low impedance device). Probably a high impedance device, such as a transmitter or an indicator, in the loop is preventing the communication. The solution lies in connecting a capacitor (0.1 to 1μF x 200V) in parallel to the high impedance device that is preventing the communication. • If it doesn't solve the problem, check the wiring and the configuration in the corresponding HIRT block.

Configuration

Problem	Solution
I can't instantiate the MAI/MAO block.	• Check the firmware version in the HCFG.FIRMWARE_VERSION parameter. Only the HI302-I supports the MAI block. The HI302-O supports the MAO block. The HI302-N supports neither the MAI nor the MAO blocks. • Check if the correct DD is installed in the configurator and is being used.

After sending the configuration to the HI302, the HART communication doesn't work.	• Check if the HART channel LEDs are lit. The HCFG.COMM_ENABLE parameter must be on DISABLED. Write the ENABLED value manually to start operating the HI302.
	• Check if the HIRT blocks are set as follow:
	o MODE_BLK.ACTUAL = AUTO;
	o BLOCK_ERR = PowerUp;
	o BLK_EXEC_STATE = IDENTIFICATION.

What's the correct procedure to remove a device for maintenance?	• Set the block to OS (MODE_BLK = OS). Thus, the block won't be occupied trying to communicate with devices that are not being used.

Operation

Problem	Solution
The HIRT block does not change from the IDENTIFICATION status.	• Check if the block is working properly: MODE_BLK.ACTUAL = AUTO and BLOCK_ERR = PowerUp. • Check if the HART communication was enabled in the HCFG.COMM_ENABLE parameter.
The HIRT block does not change from DEVICE NOT FOUND status.	• The device was not found by using the command selected in the HIRT.ID_CMD parameter. • The POLL_ADDR or the HART_TAG may be wrong. Check them. • Check if the device is On and communicating with another HART Master, for example, the Smar HPC301 or another portable programmer. If so, check the HI302 connections. Tip: Use the command 0 (polling address) to perform the identification. Thus, the HI302 will poll the addresses every 30 seconds, until it finds the device.
Once set at UPDATED, the HI302 does not start to poll the device or the variables haven't been updated.	• Check if the Polling is enabled in the HIRT.POLL_CTRL parameter. • Check if the VIEW selected in the HIRT.VIEW_SELECTION parameter corresponds to the right VIEW to update the desired dynamic variables. Each VIEW reads different HART variables and updates a different group of parameter under a different polling cycle. • Check if the dynamic variables are from the A to D group, check the configuration code of these variables

Problem	Solution
The HVT block does not start UPDATING when writing the TAG on the DEV_TAG_SEL parameter.	• Check if the block is working: MODE_BLK.ACTUAL = AUTO and BLOCK_ERR = PowerUp. • Check if the TAG that has been written on the DEV_TAG_SEL parameter is the HART_TAG and not the HIRT block TAG. • Check if the HI302 has the configuration for the desired HART device The HI302 uses the MAN_ID, DEV_TYPE, UNI_REV, SPEC_REV and SW_REV parameters to find the configuration in the memory.
The read value does not correspond to the HART device variable.	• Check if BLK_EXEC_STATE = UPDATED. The parameters related to the HART variable are valid if the BLK_EXEC_STATE = UPDATED. • For the HVT block, make sure the read parameter corresponds to the desired variable. See the HVT block allocation map.
I can't write in the parameter.	• Does the parameter correspond to a HART variable that can be written? Is this reflected in the current device´s configuration? Check them. • Even for the HIRT block, check if the device supports the HART command used for writing. See on the Appendix section a detailed explanation about the commands used by the HIRT block for writing.
When attempting to program a new firmware (a new version of the resident program), the FBTools displays a new error message.	• Check if the HI302 is in Bootloader mode. In this mode, only the ON (green), SAVING (yellow), and H1 (green) LEDs must be lit. To access the Bootloader mode, press the RESET button one time, with the module started up. By pressing again the RESET button after the programming is done, the module will start executing the new resident program.

Next see some of the most common questions about the HI302:

1. **Does the HART Communication interfere with the 4-20mA Control Signal?**
No. The HART Communication is imposed by a signal modulated in FSK (Frequency Shift Keying) over the current signal generated or received by the device, in a frequency usually ranging from 500 Hz to 4 KHz. According to the theory of communications, a signal modulated in FSK has an average value equal to 0, and therefore no disturbance is caused on the control signal. Thus, the HI302 module may be connected to any existing installation. The wiring used in the HI302's installation may be a noise source and cause interference in the control loop. To avoid such problem, the user should use a shielded cable with the minimum length and with reinforced electrical connection.

2. **Does the HI302 work with third-party Devices, that is, those not made by Smar?**
Yes. HI302 was designed to work with any devices that comply with HFC (HART Communication Foundation) norms. The most used universal and common practice commands are supported by the HI302 through the HIRT block, with no need for a special command configuration. If the user wants to use the device's complete set of commands (Common Practice and Specific Commands), a specific configuration must be loaded, based on the device documentation. For Smar devices, this configuration is already built in the HI302's memory.

3. **Can I link the PV_VAL parameter in my the FF strategy?**
No. It is not possible to link any HART-related parameter in this version. <u>The HART variables are available only for supervision. If your application calls this specific feature, check if the latest release supports it.</u>

4. **Are the HI302 channels isolated among themselves?**
The HI302-N model has channels isolated among themselves and they can be connected to different I/O systems with different power supplies with independent grounding. The HI302-I and HI302-O analog boards short-circuit the grounding of the eight channels, making the HART channels not isolated among themselves.

5. **Which dynamic variables of the HART device can the HI302 read and how the user can access them?**
The HI302 has a group of parameters that can be updated during the polling, according to the VIEW selected in the HIRT.VIEW_SELECTION parameter. The VIEWs use the HART 33 command to access the variables listed in the HART device. Actually, any HART variable, accessible by the 1, 2, 3 or 33 commands, can be read by the HI302. The variables accessible by the command 33 require the configuration of the HIRT block XX_CODE parameters. For example, the user wants to read the TEMP_MAX variable, which the manufacturing code is 12. Then, configure the A1_CODE parameter with 12 and select the VIEW_02. Thus, the variable TEMP_MAX will be read in the A1_VAL parameter, and in turn its unit on the A1_UC parameter.

6. **How many HI302 modules can be installed on the same H1 segmentl?**
From the H1Channel's perspective, up to sixteen modules can be connected to it, as it is a passive-MAU device. It must be remembered, that the HI302 module works as a HART device concentrator, supporting up to 32 HART devices in continuous communication. The total number of blocks (or TAGs) to be supervised, as well as the supervision periodicity, must be taken into consideration for this dimensioning. The suggestion is for no more than 32 HART devices per H1 channel, namely,

    - one HI302 in the multidrop mode, with 8 channels x 4 HART instruments per channel or,
    - 4 point-to-point HI302, with 8 channels x 1 HART device per HI302 channel.

7. **Can other types of devices be installed on the same H1 segment?**
Yes, as long as all limitations and restrictions are considered regarding the time for blocks and parameters supervising, publication of links, etc. The HI302 macrocycle is very large (~1s) because the HIRT block execution takes a lot of time (~50 ms).

8. **How many HART devices can be connected to a HI302?**
Up to a total of 32 instruments (transmitters or actuators), being at most 15 (addressed from 1 to 15) for each channel. It is strongly recommended that at most 4 devices per channel be installed, so that the communication in the multidrop mode is most efficient.

9. **What is the difference between the HI302-N, HI302-I and HI302-O?**
The three work on the same basic hardware and firmware platform, the difference being on the analog conversion board installed over the main board.

    - On the HI302-N model there's no analog conversion, only HART communication.
    - On the HI302-I, the circuit converts 4-20mA to FF, through a MAI block.
    - On the HI302-O, the installed circuit converts from FF to 4-20mA, by using a MAO block.

10. **What is the Update Periodicity of the HART variables on the HI302 blocks?**
This period of time depends on a series of factors such as: number of devices per channel, communication errors, presence of another Host in the channel, a device in Burst Mode, number of the selected VIEW, etc. Generally speaking, a HART transaction lasts about 800 ms to be finished. See in the VIEW table, the average time to update the dynamic variables.

11. **Does the HI302 let Portable Programmers to be used, such as the HPC301?**
Yes. Since the HPC301 is a Secondary Master, it is possible to have it connected to a HI302 channel without any problems, provided this channel is configured as a Primary Master.

Appendix A

HCFG - HART CONFIGURATION & DIAGNOSTIC

Index	Parameter	Data Type	Valid Range / Options	Default Value	Units	Store / Mode	Description
0	BLOCK_STRUCTURE	DS-64			NA	S	
1	ST_REV	Unsigned16		0	None	S / RO	
2	TAG_DESC	OctString(32)		Spaces	NA	S	
3	STRATEGY	Unsigned16		0	None	S	
4	ALERT_KEY	Unsigned8	1 to 255	0	None	S	
5	MODE_BLK	DS-69		O/S	NA	S	See Mode Parameter
6	BLK_ERR	Bitstring(2)			None	D / RO	
7	FIRMWARE_VERSION	VisibleString(32)		0	NA	D / RO	Useful for information and diagnostic purposes.
8	COMM_BEHAVIOR	Enumerated	**0x00:** Autonomous **0x01:** Bypass	Autonomous	NA	S	Autonomous means the normal behavior where the database is built automatically. If set to Bypass the device depends on external applications to send HART commands and the block HBC must be used.
9	COMM_ENABLE	Enumerated	**0x01** Enabled **0x00:** Disabled	Disabled	NA	S	After a download or if a channel has been changed in a HIRT block, this parameter will be set to Disabled automatically. Set it to Enabled to start HART communication. **Caution: this parameter must not be saved OFF LINE and downloaded. It must always be written in ON LINE mode only!**
10	CHANNEL_ACTIVE	Enumerated[8]	**0x00:** No **0x01:** 1 **0x02:** 2 **0x0F:** 15	No	NA	D / RO	This parameter shows how many HIRT blocks have been configured to use the corresponding channel. If no block is using the channel, it will remain deactivated.
11	COMM_ERRORS	Float[8]		0	%	D / RO	Shows the percentual of communication errors. Up to 0.5% is acceptable for more than 10k requests.
12	MASTER_TYPE	Enumerated[8]	**0x01**: Primary **0x00**: Secondary	Primary	NA	S	HART Master Type, normally Primary.
13	RETRIES	Unsigned8[8]	3 to 10	3	None	S	Number of retries if slave does not respond before slave timeout or if any error is received. Increase the number to make the communication more reliable in noisy environments.
14	MASTER_SYNCHRONIZED	Boolean[8]	**0x01:** Synchronized **0x00:** Not synchronized	Not synchronized	NA	D / RO	**FACTORY USE** - Synchronized means normal operation.
15	CHANNEL_MODE	Enumerated[8]	**0x00:** Normal **0x01:** Burst Mode	Normal	None	D / RO	**FACTORY USE -** This parameter shows if any burst mode device was detected on the respective channel.
16	MASTER_STATE	Enumerated[8]	**0x00:** Watching **0x01:** Enabled **0x02:** Using	Watching	None	D / RO	**FACTORY USE -** It's the Master State Machine behavior at each moment.
17	REQUEST_COUNTER	Unsigned32[8]		0	None	D / RO	**FACTORY USE -** Counts the number of requests made to all devices on that channel.

Index	Parameter	Data Type	Valid Range / Options	Default Value	Units	Store / Mode	Description
18	RETRIES_COUNTER	Unsigned32[8]		0	None	D / RO	**FACTORY USE -** Counts the number of retries to all devices on each channel.
19	INVALID_SOM	Unsigned32[8]		0	None	D / RO	**FACTORY USE -** Counts the number of invalid Start Of Messages captured in that channel.
20	INVALID_RX_FRAMES	Unsigned32[8]		0	None	D / RO	**FACTORY USE -** It totalizes the number of INVALID frames received by each channel, whichever the error.
21	VALID_RX_FRAMES	Unsigned32[8]		0	None	D / RO	**FACTORY USE -** It totalizes the number of VALID frames received by each channel.
22	ANALOG_INPUT_TRIM	Enumerated	**0x00:** Channel 1, **0x01:** Channel 2, **0x02:** Channel 3, **0x03:** Channel 4, **0x04:** Channel 5, **0x05:** Channel 6, **0x06:** Channel 7, **0x07:** Channel 8, **0x08:** All Channels, **0x09:** Not Trimmed, **0x0A:** Trimmed and Checked	Not Trimmed	NA	S	**FACTORY USE** - Used to calibrate the analog inputs when applicable. Apply a stable signal of 12 mA (+/- 0.005 mA) to the channel (or to all channels at once) and write to this parameter accordingly to the channel you want to calibrate (or All Channels if you want to calibrate all at once). **After calibrate and test write this parameter to Trimmed and Checked to save the data.**
23	ANALOG_OUTPUT_CAL	Float[8]		1.00	mA	S	**FACTORY USE** - This array is used to calibrate the analog outputs when applicable. Using a precision miliampmeter (+/- 1uA) put all outputs in 50% by actuating in MAO block. Using the value the meter is reading, write it in the correspondig element of this array, always with at least 2 decimal, in mA (should be ~ 12 mA). After written the value look at the meter again and confirm the reading now is 12.0 mA. **After calibrate and test write ANALOG_INPUT_TRIM parameter to Trimmed and Checked to save the calibration data.**
24	UPDATE_EVT	DS-73			NA	D	This alert is generated by any change to the static data.
25	BLK_ALM	DS-72			NA	D	The block alarm is used for all configuration, hardware, connection failure or system problems in the block. The cause of the alert is entered in the subcode field. The first alert to become active will set the Active status in the Status attribute. As soon as the Unreported status is cleared by the alert reporting task, another block alert may be reported without clearing the Active status, if the subcode has changed.

HIRT - HART INFORMATION & DYNAMIC DATA

Index	Parameter	Data Type	Valid Range / Options	Default/ Value	Units	Store /Mode	Description	HART Read	Hart Write
0	BLOCK_STRUCTURE	DS-64			NA	S			
1	ST_REV	Unsigned16		0	None	S / RO			
2	TAG_DESC	OctString(32)		Spaces	NA	S			
3	STRATEGY	Unsigned16		0	None	S			
4	ALERT_KEY	Unsigned8	1 to 255	0	None	S			
5	MODE_BLK	DS-69		O/S	NA	S	See Mode Parameter		
6	BLK_ERR	Bitstring(2)			None	D / RO			
7	HART_CHANNEL	Unsigned8	1 to 8	1	None	S	HART channel where the device is attached to.		
8	ID_CMD	Enumerated	**0:** 0 - Polling Address **11:** 11 - HART Tag	0 - Polling Address	None	S	Selects the HART Universal Command used to identify the device associated with this block.		
9	LAST_HART_WRITE_STATUS	Enumerated	**0x00: No Command-Specific Errors** **0x05: Too Few Data Bytes Received** **0x07: In Write Protection Mode** **0x08: Warning: Update Failure** **0x20: Busy** **0x40: Command Not Implemented** **0x81: Undefined Comm Error** **0x82: Buffer Overflow** **0x84: Reserved Comm Error** **0x88: Longitudinal Parity Error** **0x90: Framing Error** **0xA0: Overrun Error** **0xC0: Vertical Parity Error**	0x00: No Command-Specific Errors	None	D / RO	This parameter always reflects the first byte of the Responde Code (Command Response Summary) of THE LAST WRITE HART COMMAND ISSUED and must be checked to ensure the write operation has been succeeded.		
10	POLL_ADDR	Unsigned8	0 to 15	0	None	S	Device's Polling (short) Address).	**7**	**6**
11	HART_TAG	VisibleString(8)		Spaces	NA	S	Device's TAG.	**13**	**18**
12	POLL_CTRL	Boolean	**0x01:** Polling Enabled **0x00:** Polling Disabled	Polling Enabled	NA	S	Real Time Data Polling Control.		
13	VIEW_SELECTION	Enumerated	**0x00:** VIEW_00, **0x01:** VIEW_01 **0x02:** VIEW_02, **0x03:** VIEW_03 **0x04:** VIEW_04, **0x05**: VIEW_05 **0x06:** VIEW_06, **0x07:** VIEW_07 **0x08:** VIEW_08, **0x09:** VIEW_09 **0x0A:** VIEW_10, **0x0B:** VIEW_11 **0x0C:** VIEW_12, **0x0D:** VIEW_13 **0x0E:** VIEW_14, **0x0F:** VIEW_15 **0x10:** VIEW_16, **0x11:** VIEW_17 **0x12:** VIEW_18, **0x13:** VIEW_19 **0x14**: VIEW_20, **0x15:** VIEW_21 **0x16:** VIEW_22, **0x17:** VIEW_23 **0x18:** VIEW_24, **0x19:** VIEW_25 **0x1A:** VIEW_26, **0x1B:** VIEW_27 **0x1C:** VIEW_28, **0x1D:** VIEW_29	VIEW_00	None	S	Selects the set of parameters to be updated at each polling cycle. See HI302's user manual for further information about each VIEW.		

Index	Parameter	Data Type	Valid Range / Options	Default/ Value	Units	Store /Mode	Description	HART Read	Hart Write
14	COMMON_CMD_FILTER	Unsigned8[5]		0	None	S	This parameter has 5 positions to ignore commands present in configuration but not supported by the device. See HI302´s User Manual for a complete list of that commands.		
15	BLK_EXEC_STATE	Enumerated	**0x00:** Identification **0x01:** Old Data **0x02:** Updating **0x03:** Updated **0x04:** Partially Updated **0x05:** Not Responding **0x06:** Bypass **0x07:** Device Not Found **0x08:** HCD Error **0x09**: TAG Not Found **0x0A:** Writing	Identification	None	D / RO	Status of parameter update information and HART communication. Normal condition is UPDATED.		
16	COMM_ERR	BitString(8)	**0x00: No Command-Specific Errors** **0x05: Too Few Data Bytes Received** **0x07: In Write Protection Mode** **0x08: Warning: Update Failure** **0x20: Busy** **0x40: Command Not Implemented** **0x81: Undefined Comm Error** **0x82: Buffer Overflow** **0x84: Reserved Comm Error** **0x88: Longitudinal Parity Error** **0x90: Framing Error** **0xA0: Overrun Error** **0xC0: Vertical Parity Error**	**0x00: No Command-Specific Errors**	None	D / RO	First byte of the last transaction´s Response Code if communication error has occurred (Bit7 = 1).	**All**	**All**
17	DEVICE_STATUS	BitString(8)	**0x01:** Primary Variable Out of Limits **0x02:** Non-Primary Variable Out of Limits **0x04:** Loop Current Saturated **0x08:** Loop Current Fixed **0x10:** More Status Available **0x20:** Cold Start **0x40:** Configuration Changed **0x80:** Device Malfunction	0	NA	D / RO	Second byte of the last transaction´s Response Code. See HI302's User Manual or HART specification for further details.	**All**	**All**
18	ADDITIONAL_STATUS	OctString(6)		0	NA	D / RO	Cyclic read depends on chosen VIEW. Device specific Additional Status. See device´s specific documentation for details.	**48**	
19	MAN_ID	Enumerated	HC TABLE 8	0	None	D / RO	Manufacturer ID Code. Used to select specific configuration for HVT block.	**(0, 11)**	
20	DEV_TYPE	Unsigned8		0	None	D / RO	Manufacturer Device Type Code. Used to select specific configuration for HVT block.	**(0, 11)**	
21	UNI_REV	Unsigned8		0	None	D / RO	Revision Level of the HART Universal Commands. Used to select specific configuration for HVT block.	**(0, 11)**	

Index	Parameter	Data Type	Valid Range / Options	Default/ Value	Units	Store /Mode	Description	HART Read	Hart Write
22	SPEC_REV	Unsigned8		0	None	D / RO	Revision Level of the Device Specific. Used to select specific configuration for HVT block.	**(0, 11)**	
23	SW_REV	Unsigned8		0	None	D / RO	Software Revision Level. Used to select specific configuration for HVT block.	**(0, 11)**	
24	HRDW_REV	Unsigned8		0	None	D / RO	Hardware Revision Level.	**(0, 11)**	
25	FLAGS	BitString(8)	HC TABLE 11	0	NA	D / RO	Flags (manufacturer specific).	**(0, 11)**	
26	DEV_ID	OctString(3)		0	NA	D / RO	Device ID Number.	**(0, 11)**	
27	LOOP_CURRENT	DS-65		0	mA	D	Cyclic read depends on chosen VIEW. Loop Current Value (milliamps). **LOOP_CURRENT.VALUE:** loop mA last value, **LOOP_CURRENT.STATUS**: HART Response Code converted to FF status. See HI302´s User Manual for further details.	**2**	
28	LOOP_TEST	Float		0	mA	D	Write the desired current value in mA to enter fixed current mode. Write 0 to exit fixed current mode.		**40**
29	LOOP_CMODE	Enumerated	HC TABLE 16	0	None	D	Loop Current Mode. See device´s specific documentation.	**7**	**6**
30	MESSAGE	VisibleString(32)		Spaces	NA	D	Message for general purpose.	**12**	**17**
31	DESCRIPTOR	VisibleString(16)		Spaces	NA	D	Descriptor, text for general purpose.	**13**	**18**
32	DATE_INFO	Date		1/1/2001 00:00:00:000 0	NA	D	Date (Only Day/Month/Year are considered).	**13**	**18**
33	WRITE_PCODE	Enumerated	HC TABLE 7	0	None	D / RO	Write Protect Code. See device´s specific documentation.	**15**	
34	PLDC	Enumerated	HC TABLE 8	0	None	D / RO	Private Label Distribuitor Code.	**15**	
35	PV_ACF	Enumerated	HC TABLE 26	0	None	D / RO	PV Analog Channel Flags. See device´s specific documentation.	**15**	
36	FAN	OctString(3)		0	NA	D	Final Assembly Number.	**16**	**19**
37	DEV_TEST	Enumerated	**0x00:** Invoke Self Test	0	None	D	Write to perform a device self test.		**41**
38	DEV_RESET	Unsigned8	**0x00:** Reset Device	0	None	D	Write to perfom a device Master RESET.		**42**
39	BURST_MODE	Enumerated	HC TABLE 9	0	None	D	It allows to control device's Burst Mode		**109**
40	PV_ULRUC	Enumerated	HC TABLE 2	0	None	D	PV Upper & Lower Range Value Units Code	**15**	**35**
41	PV_RANGE	DS-68		{0, 0, 0, 0}	XD_SCALE	D	**PV_RANGE.EU_100**: HART PV Upper Range Value, **PV_RANGE.EU_0**: HART PV Lower Range Value, **PV_RANGE_UNITS_INDEX**: HART PV Range (Upper & Lower) Value Units Code translated to Fieldbus table, **PV_RANGE.DECIMAL**: no meam.	**15**	**35, 44**
42	PV_CAL_POINT_L	Enumerated	**0x00:** Set PV Zero	0	None	D	Write to this parameter to set PV Zero (invoke HART Command 43).		**43**

Index	Parameter	Data Type	Valid Range / Options	Default/ Value	Units	Store /Mode	Description	HART Read	Hart Write
43	PV_SENSOR_SN	OctString(3)		0	NA	D	PV Sensor Serial Number.	**14**	**49**
44	PV_SENSOR_LMSUC	Enumerated	HC_TABLE 2	0	None	D / RO	PV Sensor Limits and Minimun Span Units Code.	**14**	
45	PV_SENSOR_MSPAN	Float		0	49	D / RO	PV Minimun Span.	**14**	
46	PV_SENSOR_RANGE	DS-68		{0, 0, 0, 0}	XD_SCALE	D / RO	**PV_SENSOR_RANGE.EU_100**: PV Upper Sensor Limit, **PV_SENSOR_RANGE.EU_0**: PV Lower Sensor Limit. **PV_SENSOR_RANGE.UNITS_INDEX**: PV Sensor limits and Minimun Span Units Code translated to Fieldbus table. **PV_SENSOR_RANGE.DECIMAL:** no mean.	**14**	
47	PV_ASC	Enumerated	HC TABLE 6	0	None	D / RO	PV Alarm Select code.	**15**	
48	PV_TFC	Enumerated	HC TABLE 3	0	None	D	PV Transfer Function Code.	**15**	**47**
49	PV_DV	Float		0	s	D	PV Damping Value.	**15**	**34**
50	PV_PERC	DS-65		0	%	D / RO	Cyclic read depends on chosen VIEW. PV Percent of Range. **PV_PERC.VALUE**: the percentual Pv value, **PV_PERC.Status**: HART Response Code converted to FF status.	**2**	
51	PV_UC	Enumerated	HC TABLE 2	0	None	D	PV Units Code.	**15**	**44**
52	PV_VAL	DS-65		0	PV_UC	D / RO	**Cyclic read depends on chosen VIEW.** **PV_VAL.Value** : actual PV value in engineering units, **PV_VAL.Status**: HART Response Code converted to FF status.	**3**	
53	SV_UC	Enumerated	HC TABLE 2	0	None	D / RO	SV Units Code.	**3**	
54	SV_VAL	DS-65		0	SV_UC	D / RO	**Cyclic read depends on chosen VIEW.** **SV_VAL.Value**: SV actual value, **SV_VAL.Status**:HART Response Code converted to FF status.	**3**	
55	TV_UC	Enumerated	HC TABLE 2	0	None	D / RO	TV Units Code.	**3**	
56	TV_VAL	DS-65		0	TV_UC	D / RO	**Cyclic read depends on chosen VIEW.** **TV_VAL.Value**: TV actual value, **TV_VAL.Status**: HART Response Code converted to FF status.	**3**	
57	QV_UC	Enumerated	HC TABLE 2	0	None	D / RO	QV Units Code.	**3**	
58	QV_VAL	DS-65		0	QV_UC	D / RO	**Cyclic read depends on chosen VIEW.** **QV_VAL.Value**: TV actual value, **QV_VAL.Status**: HART Response Code converted to FF status.	**3**	
59	A1_CODE	Unsigned8		0	None	S	Selects variable to appear on A1_VAL	**33**	
60	A1_UC	Enumerated	HC TABLE 2	0	None	D / RO	A1 variable Units Code.	**33**	

Index	Parameter	Data Type	Valid Range / Options	Default/ Value	Units	Store /Mode	Description	HART Read	Hart Write
61	A1_VAL	DS-65		0	S0_UC	D / RO	**OUTPUT PARAMETER. Cyclic read depends on chosen VIEW and A1_CODE.**	**33**	
62	A2_CODE	Unsigned8		0	None	S	Selects variable to appear on A2_VAL	**33**	
63	A2_UC	Enumerated	HC TABLE 2	0	None	D / RO	A2 variable Units Code.	**33**	
64	A2_VAL	DS-65		0	S1_UC	D / RO	**Cyclic read depends on chosen VIEW and A2_CODE.**	**33**	
65	A3_CODE	Unsigned8		0	None	S	Selects variable to appear on A3_VAL	**33**	
66	A3_UC	Enumerated	HC TABLE 2	0	None	D / RO	A3 variable Units Code.	**33**	
67	A3_VAL	DS-65		0	S2_UC	D / RO	**Cyclic read depends on chosen VIEW and A3_CODE.**	**33**	
68	A4_CODE	Unsigned8		0	None	S	Selects variable to appear on A4_VAL	**33**	
69	A4_UC	Enumerated	HC TABLE 2	0	None	D / RO	A4 variable Units Code.	**33**	
70	A4_VAL	DS-65		0	S3_UC	D / RO	**Cyclic read depends on chosen VIEW and A4_CODE.**	**33**	
71	B1_CODE	Unsigned8		0	None	S	Selects variable to appear on B1_VAL	**33**	
72	B1_UC	Enumerated	HC TABLE 2	0	None	D / RO	B1 variable Units Code.	**33**	
73	B1_VAL	DS-65		0	S0_UC	D / RO	**Cyclic read depends on chosen VIEW and B1_CODE.**	**33**	
74	B2_CODE	Unsigned8		0	None	S	Selects variable to appear on B2_VAL	**33**	
75	B2_UC	Enumerated	HC TABLE 2	0	None	D / RO	B2 variable Units Code.	**33**	
76	B2_VAL	DS-65		0	S1_UC	D / RO	**Cyclic read depends on chosen VIEW and B2_CODE.**	**33**	
77	B3_CODE	Unsigned8		0	None	S	Selects variable to appear on B3_VAL	**33**	
78	B3_UC	Enumerated	HC TABLE 2	0	None	D	B3 variable Units Code.	**33**	
79	B3_VAL	DS-65		0	S2_UC	D / RO	**Cyclic read depends on chosen VIEW and B3_CODE.**	**33**	
80	B4_CODE	Unsigned8		0	None	S	Selects variable to appear on B4_VAL	**33**	
81	B4_UC	Enumerated	HC TABLE 2	0	None	D / RO	B4 variable Units Code.	**33**	
82	B4_VAL	DS-65		0	S3_UC	D / RO	**Cyclic read depends on chosen VIEW and B4_CODE.**	**33**	
83	C1_CODE	Unsigned8		0	None	S	Selects variable to appear on C1_VAL	**33**	
84	C1_UC	Enumerated	HC TABLE 2	0	None	D / RO	C1 variable Units Code.	**33**	
85	C1_VAL	DS-65		0	S0_UC	D / RO	**Cyclic read depends on chosen VIEW and C1_CODE.**	**33**	
86	C2_CODE	Unsigned8		0	None	S	Selects variable to appear on C2_VAL	**33**	
87	C2_UC	Enumerated	HC TABLE 2	0	None	D / RO	C2 variable Units Code.	**33**	
88	C2_VAL	DS-65		0	S1_UC	D / RO	**Cyclic read depends on chosen VIEW and C2_CODE.**	**33**	

Index	Parameter	Data Type	Valid Range / Options	Default/ Value	Units	Store /Mode	Description	HART Read	Hart Write
89	C3_CODE	Unsigned8		0	None	S	Selects variable to appear on C3_VAL	**33**	
90	C3_UC	Enumerated	HC TABLE 2	0	None	D / RO	C3 variable Units Code.	**33**	
91	C3_VAL	DS-65		0	S2_UC	D / RO	**Cyclic read depends on chosen VIEW and C3_CODE.**	**33**	
92	C4_CODE	Unsigned8		0	None	S	Selects variable to appear on C4_VAL	**33**	
93	C4_UC	Enumerated	HC TABLE 2	0	None	D / RO	C4 variable Units Code.	**33**	
94	C4_VAL	DS-65		0	S3_UC	D / RO	**Cyclic read depends on chosen VIEW and C4_CODE.**	**33**	
95	D1_CODE	Unsigned8		0	None	S	Selects variable to appear on D1_VAL	**33**	
96	D1_UC	Enumerated	HC TABLE 2	0	None	D / RO	D1 variable Units Code.	**33**	
97	D1_VAL	DS-65		0	S0_UC	D / RO	**Cyclic read depends on chosen VIEW and D1_CODE.**	**33**	
98	D2_CODE	Unsigned8		0	None	S	Selects variable to appear on D2_VAL	**33**	
99	D2_UC	Enumerated	HC TABLE 2	0	None	D / RO	D2 variable Units Code.	**33**	
100	D2_VAL	DS-65		0	S1_UC	D / RO	**Cyclic read depends on chosen VIEW and D2_CODE.**	**33**	
101	D3_CODE	Unsigned8		0	None	S	Selects variable to appear on D3_VAL	**33**	
102	D3_UC	Enumerated	HC TABLE 2	0	None	D / RO	D3 variable Units Code.	**33**	
103	D3_VAL	DS-65		0	S2_UC	D / RO	**Cyclic read depends on chosen VIEW and D3_CODE.**	**33**	
104	D4_CODE	Unsigned8		0	None	S	Selects variable to appear on D4_VAL	**33**	
105	D4_UC	Enumerated	HC TABLE 2	0	None	D / RO	D4 variable Units Code.	**33**	
106	D4_VAL	DS-65		0	S3_UC	D / RO	**Cyclic read depends on chosen VIEW and D4_CODE.**	**33**	
107	UPDATE_EVT	DS-73			NA	D	This alert is generated by any change to the static data.		
108	BLK_ALM	DS-72			NA	D	The block alarm is used for all configuration, hardware, connection failure or system problems in the block. The cause of the alert is entered in the subcode field. The first alert to become active will set the Active status in the Status attribute. As soon as the Unreported status is cleared by the alert reporting task, another block alert may be reported without clearing the Active status, if the subcode has changed.		

HVT – HART Variable Template

Index	Parameter	Data Type	Valid Range / Options	Default Value	Units	Store / Mode	Description
0	BLOCK_STRUCTURE	DS-64			NA	S	
1	ST_REV	Unsigned16		0	None	S / RO	
2	TAG_DESC	OctString(32)		Spaces	NA	S	
3	STRATEGY	Unsigned16		0	None	S	
4	ALERT_KEY	Unsigned8	1 to 255	0	None	S	
5	MODE_BLK	DS-69		O/S	NA	S	See Mode Parameter
6	BLK_ERR	Bitstring(2)			None	D / RO	
7	DEV_TAG_SEL	VisibleString(8)		Spaces	NA	S	Write here a valid HIRT.HART_TAG from an installed device to start HVT on demand reading.
8	HCD_SELECTED	OctString(5)		0	None	D / RO	This code is used to identify the specific configuration associated with the chosen device. This code is read from HIRT block and is a combination of MAN_ID, DEV_TYPE, UNI_REV, SPEC_REV, SW_REV.
9	HCD_DEVICE_INFO	VisibleString(32)		Spaces	NA	D / RO	This parameter shows comment related to selected specific configuration.
10	BLK_EXEC_STATE	Unsigned8	**0x00:** Identification **0x01:** Old Data **0x02:** Updating **0x03:** Updated **0x04:** Partially Updated **0x05:** Not Responding **0x06:** Bypass **0x07:** Device Not Found **0x08:** HCD Error **0x09**: TAG Not Found **0x0A:** Writing	0	None	D / RO	Reflects the execution progress or error conditions. See also BLK_ERR.
11	U8B_ARRAY_1	Unsigned8[20]		0	None	D	First array used for 8-bit variables
12	U8B_ARRAY_2	Unsigned8[20]		0	None	D	Second array used for 8-bit variables
13	U8B_ARRAY_3	Unsigned8[20]		0	None	D	Thirdy array used for 8-bit variables
14	U8B_ARRAY_4	Unsigned8[20]		0	None	D	Fourth array used for 8-bit variables
15	U8B_ARRAY_5	Unsigned8[20]		0	None	D	Fifth array used for 8-bit variables
16	FLOAT_ARRAY_1	FloatingPoint[20]		0	None	D	First array used for Floating Point variables
17	FLOAT_ARRAY_2	FloatingPoint[20]		0	None	D	Second array used for Floating Point variables
18	FLOAT_ARRAY_3	FloatingPoint[20]		0	None	D	Third array used for Floating Point variables
19	FLOAT_ARRAY_4	FloatingPoint[20]		0	None	D	Fourth array used for Floating Point variables
20	FLOAT_ARRAY_5	FloatingPoint[20]		0	None	D	Fifth array used for Floating Point variables
21	U16B_ARRAY_1	Unsigned16[20]		0	None	D	First array of 16-bit (2-byte) values
22	U32B_ARRAY_1	Unsigned32[10]		0	None	D	First array of 32-bit variables
23	U32B_ARRAY_2	Unsigned32[10]		0	None	D	Second array of 32-bit variables
24	String_01	VisibleString(8)		Spaces	NA	D	First general string (8 characters)
25	String_02	VisibleString(8)		Spaces	NA	D	8 characters general use string
26	String_03	VisibleString(8)		Spaces	NA	D	8 characters general use string
27	String_04	VisibleString(8)		Spaces	NA	D	8 characters general use string
28	String_05	VisibleString(8)		Spaces	NA	D	8 characters general use string
29	String_06	VIsibleString(16)		Spaces	NA	D	16 characters general use string
30	String_07	VisibleString(16)		Spaces	NA	D	16 characters general use string

Index	Parameter	Data Type	Valid Range / Options	Default Value	Units	Store / Mode	Description
31	String_08	VIsibleString(16)		Spaces	NA	D	16 characters general use string
32	String_09	VisibleString(16)		Spaces	NA	D	16 characters general use string
33	String_10	VIsibleString(16)		Spaces	NA	D	16 characters general use string
34	String_11	VisibleString(32)		Spaces	NA	D	32 charactres general use string
35	String_12	VisibleString(32)		Spaces	NA	D	32 charactres general use string
36	String_13	VisibleString(32)		Spaces	NA	D	32 charactres general use string
37	String_14	VisibleString(32)		Spaces	NA	D	32 charactres general use string
38	String_15	VisibleString(32)		Spaces	NA	D	32 charactres general use string
39	String_16	VisibleString(32)		Spaces	NA	D	32 charactres general use string
40	String_17	VisibleString(32)		Spaces	NA	D	32 charactres general use string
41	String_18	VisibleString(32)		Spaces	NA	D	32 charactres general use string
42	String_19	VisibleString(32)		Spaces	NA	D	32 charactres general use string
43	String_20	VisibleString(32)		Spaces	NA	D	32 charactres general use string
44	UPDATE_EVT	DS-73			NA	D	This alert is generated by any change to the static data.
45	BLK_ALM	DS-72			NA	D	The block alarm is used for all configuration, hardware, connection failure or system problems in the block. The cause of the alert is entered in the subcode field. The first alert to become active will set the Active status in the Status attribute. As soon as the Unreported status is cleared by the alert reporting task, another block alert may be reported without clearing the Active status, if the subcode has changed.

HCD – HART Command Definition

Index	Parameter	Data Type	Valid Range / Options	Default Value	Units	Store / Mode	Description
0	BLOCK_STRUCTURE	DS-64			NA	S	
1	ST_REV	Unsigned16		0	None	S / RO	
2	TAG_DESC	OctString(32)		Spaces	NA	S	
3	STRATEGY	Unsigned16		0	None	S	
4	ALERT_KEY	Unsigned8	1 to 255	0	None	S	
5	MODE_BLK	DS-69		O/S	NA	S	See Mode Parameter
6	BLK_ERR	Bitstring(2)			None	D / RO	
7	HCD_CODE	OctString(5)		0	None	S	This code identifies uniquely this configuration and must be formed by combining MAN_ID, DEV_TYPE, UNI_REV, SPEC_REV and SW_REV of the targeted device.
8	DEVICE_INFO	VisibleString(32)		Spaces	NA	S	This parameter stores the device name or any other comment related to this set of command definitions.
9	CMD_00	OctString(44)	Request and Response parameters	0	NA	S	See HART Command configuration parameters.
10	CMD_01	OctString(44)	Request and Response parameters	0	NA	S	See HART Command configuration parameters.
11	CMD_02	OctString(44)	Request and Response parameters	0	NA	S	See HART Command configuration parameters.
12	CMD_03	OctString(44)	Request and Response parameters	0	NA	S	See HART Command configuration parameters.
13	CMD_04	OctString(44)	Request and Response parameters	0	NA	S	See HART Command configuration parameters.
14	CMD_05	OctString(44)	Request and Response parameters	0	NA	S	See HART Command configuration parameters.
15	CMD_06	OctString(44)	Request and Response parameters	0	NA	S	See HART Command configuration parameters.
16	CMD_07	OctString(44)	Request and Response parameters	0	NA	S	See HART Command configuration parameters.
17	CMD_08	OctString(44)	Request and Response parameters	0	NA	S	See HART Command configuration parameters.
18	CMD_09	OctString(44)	Request and Response parameters	0	NA	S	See HART Command configuration parameters.
19	CMD_10	OctString(44)	Request and Response parameters	0	NA	S	See HART Command configuration parameters.
20	CMD_11	OctString(44)	Request and Response parameters	0	NA	S	See HART Command configuration parameters.
21	CMD_12	OctString(44)	Request and Response parameters	0	NA	S	See HART Command configuration parameters.
22	CMD_13	OctString(44)	Request and Response parameters	0	NA	S	See HART Command configuration parameters.
23	CMD_14	OctString(44)	Request and Response parameters	0	NA	S	See HART Command configuration parameters.
24	CMD_15	OctString(44)	Request and Response parameters	0	NA	S	See HART Command configuration parameters.
25	CMD_16	OctString(44)	Request and Response parameters	0	NA	S	See HART Command configuration parameters.
26	CMD_17	OctString(44)	Request and Response parameters	0	NA	S	See HART Command configuration parameters.
27	CMD_18	OctString(44)	Request and Response parameters	0	NA	S	See HART Command configuration parameters.
28	CMD_19	OctString(44)	Request and Response parameters	0	NA	S	See HART Command configuration parameters.
29	CMD_20	OctString(44)	Request and Response parameters	0	NA	S	See HART Command configuration parameters.
30	CMD_21	OctString(44)	Request and Response parameters	0	NA	S	See HART Command configuration parameters.
31	CMD_22	OctString(44)	Request and Response parameters	0	NA	S	See HART Command configuration parameters.
32	CMD_23	OctString(44)	Request and Response parameters	0	NA	S	See HART Command configuration parameters.
33	CMD_24	OctString(44)	Request and Response parameters	0	NA	S	See HART Command configuration parameters.
34	CMD_25	OctString(44)	Request and Response parameters	0	NA	S	See HART Command configuration parameters.
35	CMD_26	OctString(44)	Request and Response parameters	0	NA	S	See HART Command configuration parameters.
36	CMD_27	OctString(44)	Request and Response parameters	0	NA	S	See HART Command configuration parameters.
37	CMD_28	OctString(44)	Request and Response parameters	0	NA	S	See HART Command configuration parameters.
38	CMD_29	OctString(44)	Request and Response parameters	0	NA	S	See HART Command configuration parameters.

Index	Parameter	Data Type	Valid Range / Options	Default Value	Units	Store / Mode	Description
39	CMD_30	OctString(44)	Request and Response parameters	0	NA	S	See HART Command configuration parameters.
40	CMD_31	OctString(44)	Request and Response parameters	0	NA	S	See HART Command configuration parameters.
41	CMD_32	OctString(44)	Request and Response parameters	0	NA	S	See HART Command configuration parameters.
42	CMD_33	OctString(44)	Request and Response parameters	0	NA	S	See HART Command configuration parameters.
43	CMD_34	OctString(44)	Request and Response parameters	0	NA	S	See HART Command configuration parameters.
44	CMD_35	OctString(44)	Request and Response parameters	0	NA	S	See HART Command configuration parameters.
45	CMD_36	OctString(44)	Request and Response parameters	0	NA	S	See HART Command configuration parameters.
46	CMD_37	OctString(44)	Request and Response parameters	0	NA	S	See HART Command configuration parameters.
47	CMD_38	OctString(44)	Request and Response parameters	0	NA	S	See HART Command configuration parameters.
48	CMD_39	OctString(44)	Request and Response parameters	0	NA	S	See HART Command configuration parameters.
49	CMD_40	OctString(104)	Request and Response parameters	0	NA	S	See HART Command configuration parameters.
50	CMD_41	OctString(104)	Request and Response parameters	0	NA	S	See HART Command configuration parameters.
51	CMD_42	OctString(104)	Request and Response parameters	0	NA	S	See HART Command configuration parameters.
52	CMD_43	OctString(104)	Request and Response parameters	0	NA	S	See HART Command configuration parameters.
53	CMD_44	OctString(104)	Request and Response parameters	0	NA	S	See HART Command configuration parameters.
54	CMD_45	OctString(104)	Request and Response parameters	0	NA	S	See HART Command configuration parameters.
55	CMD_46	OctString(104)	Request and Response parameters	0	NA	S	See HART Command configuration parameters.
56	CMD_47	OctString(104)	Request and Response parameters	0	NA	S	See HART Command configuration parameters.
57	CMD_48	OctString(104)	Request and Response parameters	0	NA	S	See HART Command configuration parameters.
58	CMD_49	OctString(104)	Request and Response parameters	0	NA	S	See HART Command configuration parameters.
59	UPDATE_EVT	DS-73			NA	D	This alert is generated by any change to the static data.
60	BLK_ALM	DS-72			NA	D	The block alarm is used for all configuration, hardware, connection failure or system problems in the block. The cause of the alert is entered in the subcode field. The first alert to become active will set the Active status in the Status attribute. As soon as the Unreported status is cleared by the alert reporting task, another block alert may be reported without clearing the Active status, if the subcode has changed.

HWPC – HART WriteParameter Configuration

Index	Parameter	Data Type	Valid Range / Options	Default Value	Units	Store / Mode	Description
0	BLOCK_STRUCTURE	DS-64			NA	S	
1	ST_REV	Unsigned16		0	None	S / RO	
2	TAG_DESC	OctString(32)		Spaces	NA	S	
3	STRATEGY	Unsigned16		0	None	S	
4	ALERT_KEY	Unsigned8	1 to 255	0	None	S	
5	MODE_BLK	DS-69		O/S	NA	S	See Mode Parameter
6	BLK_ERR	Bitstring(2)			None	D / RO	
7	HWPC_CODE	OctString(5)		0	None	S	Must be equal to the associated HCD.
8	WPC_00	Unsigned8[20]		0	None	S	See parameter to command correlation description.
9	WPC_01	Unsigned8[20]		0	None	S	See parameter to command correlation description.
10	WPC_02	Unsigned8[20]		0	None	S	See parameter to command correlation description.
11	WPC_03	Unsigned8[20]		0	None	S	See parameter to command correlation description.
12	WPC_04	Unsigned8[20]		0	None	S	See parameter to command correlation description.
13	WPC_05	Unsigned8[20]		0	None	S	See parameter to command correlation description.
14	WPC_06	Unsigned8[20]		0	None	S	See parameter to command correlation description.
15	WPC_07	Unsigned8[20]		0	None	S	See parameter to command correlation description.
16	WPC_08	Unsigned8[20]		0	None	S	See parameter to command correlation description.
17	WPC_09	Unsigned8[20]		0	None	S	See parameter to command correlation description.
18	WPC_10	Unsigned8[20]		0	None	S	See parameter to command correlation description.
19	WPC_11	Unsigned8[20]		0	None	S	See parameter to command correlation description.
20	WPC_12	Unsigned8[20]		0	None	S	See parameter to command correlation description.
21	WPC_13	Unsigned8[20]		0	None	S	See parameter to command correlation description.
22	WPC_14	Unsigned8[20]		0	None	S	See parameter to command correlation description.
23	WPC_15	Unsigned8[20]		0	None	S	See parameter to command correlation description.
24	WPC_16	Unsigned8[20]		0	None	S	See parameter to command correlation description.
25	WPC_17	Unsigned8[20]		0	None	S	See parameter to command correlation description.
26	WPC_18	Unsigned8[20]		0	None	S	See parameter to command correlation description.
27	WPC_19	Unsigned8[20]		0	None	S	See parameter to command correlation description.
28	WPC_20	Unsigned8[20]		0	None	S	See parameter to command correlation description.
29	WPC_21	Unsigned8[20]		0	None	S	See parameter to command correlation description.
30	WPC_22	Unsigned8[20]		0	None	S	See parameter to command correlation description.
31	WPC_23	Unsigned8[20]		0	None	S	See parameter to command correlation description.
32	WPC_24	Unsigned8[20]		0	None	S	See parameter to command correlation description.
33	UPDATE_EVT	DS-73			NA	D	This alert is generated by any change to the static data.
34	BLK_ALM	DS-72			NA	D	The block alarm is used for all configuration, hardware, connection failure or system problems in the block. The cause of the alert is entered in the subcode field. The first alert to become active will set the Active status in the Status attribute. As soon as the Unreported status is cleared by the alert reporting task, another block alert may be reported without clearing the Active status, if the subcode has changed.

HBC - Hart Bypass Communication

Index	Parameter	Data Type	Valid Range / Options	Default Value	Units	Store / Mode	Description
0	BLOCK_STRUCTURE	DS-64			NA	S	
1	ST_REV	Unsigned16		0	None	S / RO	
2	TAG_DESC	OctString(32)		Spaces	NA	S	
3	STRATEGY	Unsigned16		0	None	S	
4	ALERT_KEY	Unsigned8	1 to 255	0	None	S	
5	MODE_BLK	DS-69		O/S	NA	S	See Mode Parameter
6	BLK_ERR	Bitstring(2)			None	D / RO	
7	BYPASS_STATUS	Enumerated[8]	**0x00:** Idle, **0x01:** Busy, **0x02:** Timeout, **0x03**: Response Available	Idle	None	D / RO	This array shows the status of HART channels.
8	REQUEST_1	OctString(100)			NA	D	Write the HART frame into this parameter to send it through the channel 1.
9	RESPONSE_1	OctString(100)			NA	D / RO	Response grabbed by the channel if the addressed device has replied.
10	REQUEST_2	OctString(100)			NA	D	Write the HART frame into this parameter to send it through the channel 2.
11	RESPONSE_2	OctString(100)			NA	D / RO	Response grabbed by the channel if the addressed device has replied.
12	REQUEST_3	OctString(100)			NA	D	Write the HART frame into this parameter to send it through the channel 3.
13	RESPONSE_3	OctString(100)			NA	D / RO	Response grabbed by the channel if the addressed device has replied.
14	REQUEST_4	OctString(100)			NA	D	Write the HART frame into this parameter to send it through the channel 4.
15	RESPONSE_4	OctString(100)			NA	D / RO	Response grabbed by the channel if the addressed device has replied.
16	REQUEST_5	OctString(100)			NA	D	Write the HART frame into this parameter to send it through the channel 5.
17	RESPONSE_5	OctString(100)			NA	D / RO	Response grabbed by the channel if the addressed device has replied.
18	REQUEST_6	OctString(100)			NA	D	Write the HART frame into this parameter to send it through the channel 6.
19	RESPONSE_6	OctString(100)			NA	D / RO	Response grabbed by the channel if the addressed device has replied.
20	REQUEST_7	OctString(100)			NA	D	Write the HART frame into this parameter to send it through the channel 7.
21	RESPONSE_7	OctString(100)			NA	D / RO	Response grabbed by the channel if the addressed device has replied.
22	REQUEST_8	OctString(100)			NA	D	Write the HART frame into this parameter to send it through the channel 8.
23	RESPONSE_8	OctString(100)			NA	D / RO	Response grabbed by the channel if the addressed device has replied.
24	UPDATE_EVT	DS-73			NA	D	This alert is generated by any change to the static data.
25	BLK_ALM	DS-72			NA	D	The block alarm is used for all configuration, hardware, connection failure or system problems in the block. The cause of the alert is entered in the subcode field. The first alert to become active will set the Active status in the Status attribute. As so

Appendix B

HART Command Configuration in FLASH memory

HIRT Block

When the device is identified by the command selected in the HIRT.ID_CMD, the HI302 module will send sequentially all the reading commands in the configuration of the memory. Since this set of commands includes some Common Practice commands, the device may not support all the sent commands, causing, then, retransmission and time waste in the communication. In this case, it is necessary for the user to identify the command in the table below and to configure the command index not supported by the HIRT.COMMOM_CMD_FILTER parameter. This parameter is a filter with 5 positions and because its indexes are not executed, a faster HIRT update process with no retransmission is generated. For example, to not send the HART 48 command (READ ADITIONAL STATUS), just write the 24 index in the HIRT. COMMON_CMD_FILTER.[1] parameter.

Sent HART Reading Command used by the HIRT Block

Index	HART Command	Description
12	1	PV Reading [EU].
13	2	Current reading (mA) and PV reading in percentage.
14	3	Read the loop current (mA) and the PV, SV, TV, QV [EU].
16	12	Read the message.
17	13	Read the TAG, DESCRIPTOR and DATE.
18	14	Read the information about the PV sensor.
19	15	Read the information about the PV analog output.
20	16	Read the FINAL ASSEMBLY NUMBER.
24*	48*	Read the ADDITIONAL STATUS INFO.
37*	33*	Read the dynamic variables of the group A.
38*	33*	Read the dynamic variables of the group B.
39*	33*	Read the dynamic variables of the group C.
40*	33*	Read the dynamic variables of the group D.

**Common Practice Command. May not be implemented in the device.*

Identification HART Commands

Identification commands are those used to get the device’s initial data, such as its long address. The HIRT block may use the following commands:

Index	HART Command	Description
11	0	Use the polling address.
36	11	Use the device TAG.

HART Writing and Execution Commands used by the HIRT Block

The execution commands are those for which the module does not wait a response containing data that can update the block parameter. An example of execution command is the RESET command of the device.

Index	HART Command	Description
15	6	Write the polling address.
21	17	Write the MESSAGE.
22	18	Write the TAG, DESCRIPTOR and DATE.
23	19	Write the FINAL ASSEMBLY NUMBER.
25*	40*	Enter or Exit the fixed current mode.
26*	41*	Run the self-test.
27*	42*	Restart the device (RESET).
28*	47*	Write the PV transfer function.
29*	34*	Write the PV DAMPING VALUE.
30*	35*	Write the PV RANGE VALUES.
31*	38*	Reset the configuration change flag.
32*	49*	Write the PV sensor serial number.
33*	44*	Write the PV unit code [EU].
34*	109*	Control the device BURST mode.
35*	43*	Adjust the PV "zero".

*Common Practice Command. May not be implemented in the device.

HVT Block

The configurations for the HVT block include the Common Practice and the specific commands on each device. To use this block, the HI302 should have any configuration to support the commands of the desired HART device.

When the user writes the HART_TAG on the HVT.DEV_TAG_SEL parameter, the HI302 automatically searches for a configuration that supports the HART device. So, the HI302 will send all the reading commands available in this configuration.

See in the Appendix C, the HART variables mapping for the HVT block parameters of each configuration in the HI302's memory.

Appendix C

HVT's ALLOCATION MAP for the FY301

Index	Parameter Name	HART Variable Name	HART Command		Description
11.1	U8B_ARRAY_1[1]	EEPROM_CONTROL	39	w	Eeprom Control
11.2	U8B_ARRAY_1[2]	DISPLAY_CONNECTED	128	r	Indicates if the Display is connected or not
11.3	U8B_ARRAY_1[3]	AIR_TO	128.129	w/r	Air to (open or close)
11.4	U8B_ARRAY_1[4]	LOCAL_KEYS_MODE_CTL	132	r	Local Adjustment (enable or disable)
11.6	U8B_ARRAY_1[6]	TABLE_NUMBER_POINTS	133.135	w/r	Number of Points in the table
11.8	U8B_ARRAY_1[8]	OPERATION_CODE	153	w	Saves Data from Eeprom Main board to Eeprom transducer board or vice-versa.
11.10	U8B_ARRAY_1[10]	INDICATION_DISPLAY	247.248	w/r	Display Indication (SP or PV)
11.11	U8B_ARRAY_1[11]	VALVE_ACTION	156.157	w/r	Valve Action (direct or reverse)
11.12	U8B_ARRAY_1[12]	COUNT_CHNG_MA_OPEN_CLOSE	166	r	Number of times the current work range was changed (Split Range)
11.13	U8B_ARRAY_1[13]	COUNT_CHNG_FLOW_CHAR	166	r	Number of times the Flow Characterization was changed
11.14	U8B_ARRAY_1[14]	COUNT_CHNG_DIRECT_REVERSE	166	r	Number of times the Valve Action was changed
11.15	U8B_ARRAY_1[15]	COUNT_CHNG_SERVO_PID_SP_LMTS	166	r	Number of times the SP limits was changed
11.16	U8B_ARRAY_1[16]	COUNT_CHNG_SERVO_KP	166	r	Number of times the GAIN was changed
11.17	U8B_ARRAY_1[17]	COUNT_CHNG_SERVO_TR	166	r	Number of times the Integral was changed
11.18	U8B_ARRAY_1[18]	COUNT_CHNG_AIR_TO_OPEN_CLOSE	166	r	Number of times the AIR_TO was changed
11.19	U8B_ARRAY_1[19]	COUNT_CHNG_LOCAL_REMOTE	166	r	Number of time the Mode was changed to local or remote
11.20	U8B_ARRAY_1[20]	COUNT_CHNG_SP_VALUE	166	r	Number of times the SETPOINT was changed while the mode in local
12.1	U8B_ARRAY_2[1]	COUNT_CHNG_FAIL_SAFE	166	r	Number of times the FAIL_SAFE_MODE was changed
12.2	U8B_ARRAY_2[2]	COUNT_CHNG_AUTO_SETUP	166	r	Number of times the SETUP process was issued.
12.3	U8B_ARRAY_2[3]	COUNT_CHNG_TRAVEL_CLOSE	166	r	Number of times the Lower trim was done
12.4	U8B_ARRAY_2[4]	COUNT_CHNG_TRAVEL_OPEN	166	r	Number of times the Upper trim was done
12.5	U8B_ARRAY_2[5]	COUNT_CHNG_MOTION_TYPE	166	r	Number of time the motion type was changed (linear or rotative)
12.6	U8B_ARRAY_2[6]	COUNT_CHNG_TRIM_4MA	166	r	Number of times the 4 mA trim was done
12.7	U8B_ARRAY_2[7]	COUNT_CHNG_TRIM_20MA	166	r	Number of time the 20 mA trim was done
12.8	U8B_ARRAY_2[8]	COUNT_CHNG_PASSWORD	166	r	Number of time the Password was changed
12.9	U8B_ARRAY_2[9]	COUNT_CHNG_CONF_LEVEL	166	r	Number of times the Password Configuration Level was changed
12.10	U8B_ARRAY_2[10]	COUNT_CHNG_PERFORMANCE	166	r	Number of times the Performance test was done
12.12	U8B_ARRAY_2[12]	SOURCE_PROTECTION	170	r	Indicates the source of the Write Protection (hardware or software)
12.13	U8B_ARRAY_2[13]	POT_DC	204	r	Pot Dc Value
12.15	U8B_ARRAY_2[15]	UNIT_STROKES	237	r	Strokes unit
12.17	U8B_ARRAY_2[17]	VAL_TYPE_DIAG_REF	233	r	Valve Type Diagnosis Reference
12.18	U8B_ARRAY_2[18]	VALVE_MOTION	187.188	w/r	Valve Type (linear or rotary)
12.19	U8B_ARRAY_2[19]	TRAVEL_DEADBAND_UNIT	189	r	Travel_Deadband unit
12.20	U8B_ARRAY_2[20]	TRAVEL_LIMIT_UNIT	189	r	Travel_limit unit
13.1	U8B_ARRAY_3[1]	TRAVEL_RANGE_UNIT	189	r	Travel_Range unit
13.2	U8B_ARRAY_3[2]	MILEAGE_UNIT	190.237	w/r	Mileage unit
13.3	U8B_ARRAY_3[3]	PRESSURE_STATUS	246	r	Pressure Sensor status
13.6	U8B_ARRAY_3[6]	UNIT_PRESSURE	239.240	w/r	Pressure unit

HVT's ALLOCATION MAP for the FY301					
Index	Parameter Name	HART Variable Name	HART Command		Description
13.7	U8B_ARRAY_3[7]	FAIL_SAFE_MODE	203.15	w/r	Fail Safe Position (closed or open)
13.8	U8B_ARRAY_3[8]	UNIT_HIGHEST_TEMP	237	r	Highest_Temp unit
13.9	U8B_ARRAY_3[9]	UNIT_LOWEST_TEMP	237	r	Lowest_Temp unit
13.10	U8B_ARRAY_3[10]	UNIT_REVERSALS	237	r	Reversal unit
13.11	U8B_ARRAY_3[11]	SERVO_PID_MODE	214.215.216	w/r	Configure if the servo-pid is enable or disable
13.12	U8B_ARRAY_3[12]	PROCESS_VARIABLES_UNIT	216	r	PV unit
13.13	U8B_ARRAY_3[13]	SETPOINT_UNIT	216.225.226	w/r	SP unit
13.14	U8B_ARRAY_3[14]	MANIPULATE_VARIABLE_UNIT	216	r	MV unit
13.15	U8B_ARRAY_3[15]	SETPOINT_TRACKING	216	r	Not used by FY301
13.16	U8B_ARRAY_3[16]	CONST_SERVO_PID	224	w	Increment the SP changing counter
13.17	U8B_ARRAY_3[17]	ERROR_UNIT_CODE	216	r	Error unit (SP-PV)
13.18	U8B_ARRAY_3[18]	CONTROLLER_MODE	223.224	w/r	FY301 SP Mode Operation (local or remote)
13.19	U8B_ARRAY_3[19]	CONTROLLER_COUNTER	226	w	Configure SP value while the mode was local
13.20	U8B_ARRAY_3[20]	UNIT_CODE	227	r	Power_Up setpoint unit
14.1	U8B_ARRAY_4[1]	PERCENT_UNIT_CODE	228	w	Power_Up setpoint unit (always percentage)
14.2	U8B_ARRAY_4[2]	DIAGNOSTIC_FLAG	230.231	w/r	Diagnosis function (enable or disable)
14.3	U8B_ARRAY_4[3]	DIAGNOSTIC_FLAG_ADDRESS	232	w	Reset wished Diagnositic Variable (Stroke, Reversal or Travel)
14.4	U8B_ARRAY_4[4]	VAL_ACTION_DIAG_REF	233	r	Valve Action Diagnosis References
14.5	U8B_ARRAY_4[5]	AIR_TO_DIAG_REF	233	r	Air To Diagnostic Diagnosis References
14.6	U8B_ARRAY_4[6]	VAL_CHAR_DIAG_REF	233	r	Valve Char Diagnostic References
14.7	U8B_ARRAY_4[7]	EXE_TRIM_LOWER_POSITION	130	w	Trim Lower Position
14.8	U8B_ARRAY_4[8]	EXE_TRIM_UPPER_POSITION	131	w	Trim Upper Position
14.9	U8B_ARRAY_4[9]	EXE_SETUP_COMMAND	152	w	Start Setup procedure
14.10	U8B_ARRAY_4[10]	EXE_WRITE_DIAG_REF	234	w	Backup the Diagnostic reference settings
14.11	U8B_ARRAY_4[11]	EXE_PERFORMANCE_TIME	235	w	Start the process to obtain the performance time
14.12	U8B_ARRAY_4[12]	ANALOG_LATCH	204	r	Analog Latch configuration
14.13	U8B_ARRAY_4[13]	PRESSURE_MODE	242	r	Indicates the Pressure Mode (Instaled or Not Instaled)
16.1	FLOAT_ARRAY_1[1]	MEAS_PV_CURR_LEVEL	45	w	Current value used in the AD trim current
16.2	FLOAT_ARRAY_1[2]	TRIM_UPPER_POS	168	w	Valve Position
16.3	FLOAT_ARRAY_1[3]	TRIM_PRESSURE_IN_LOWER	241	w	Trim Pressure In Lower
16.4	FLOAT_ARRAY_1[4]	TRIM_PRESSURE_OUT1_UPPER	241	w	Trim Pressure Out1 Upper
16.5	FLOAT_ARRAY_1[5]	TRIM_PRESSURE_OUT1_LOWER	241	w	Trim Pressure Out1 Lower
16.6	FLOAT_ARRAY_1[6]	TRIM_PRESSURE_OUT2_UPPER	241	w	Trim Pressure Out2 Upper
16.7	FLOAT_ARRAY_1[7]	PRESSURE_LOW_LIMIT	244.245	w/r	Lower pressure limit for the input sensor
16.8	FLOAT_ARRAY_1[8]	TRIM_LOWER_POS	168	w	Valve position
16.9	FLOAT_ARRAY_1[9]	TRAVEL_DEADBAND	189.190	w/r	Travel deadband value
16.10	FLOAT_ARRAY_1[10]	TRAVEL_LIMIT	189.190	w/r	It is the maximum distance covered by the valve before an alarm is generated
16.11	FLOAT_ARRAY_1[11]	TRAVEL_RANGE	189.190	w/r	It is the lengh or the distance covered by the valve motion
16.12	FLOAT_ARRAY_1[12]	DEVIATION_DEADBAND	193.194	w/r	Deviation deadband value

HVT's ALLOCATION MAP for the FY301					
Index	**Parameter Name**	**HART Variable Name**	**HART Command**		**Description**
16.13	FLOAT_ARRAY_1[13]	DEVIATION_TIME	193.194	w/r	Time value in seconds that error must exceed the deviation deadband before an alarm is generate
16.14	FLOAT_ARRAY_1[14]	REVERSAL_DEADBAND	193.194	w/r	REVERSALS deadband value
16.15	FLOAT_ARRAY_1[15]	REVERSAL_LIMIT	193.194	w/r	REVERSALS limit value
16.16	FLOAT_ARRAY_1[16]	PRESSURE_HIGH_LIMIT	244.245	w/r	Upper pressure limit for the input sensor
16.17	FLOAT_ARRAY_1[17]	TRIM_PRESSURE_IN_UPPER	241	w	Trim Pressure In Upper
16.18	FLOAT_ARRAY_1[18]	HIGHEST_TEMP	237	r	Highest temperature felt by the positioner
16.19	FLOAT_ARRAY_1[19]	LOWEST_TEMP	237	r	Lowest temperature felt by the positioner
16.20	FLOAT_ARRAY_1[20]	PROCESS_VARIABLE	216	r	MV value
17.1	FLOAT_ARRAY_2[1]	SETPOINT	216.225.226	w/r	SP value
17.2	FLOAT_ARRAY_2[2]	MANIPULATE_VARIABLE	216	r	MV value
17.3	FLOAT_ARRAY_2[3]	ERROR	216	r	Error value
17.4	FLOAT_ARRAY_2[4]	PROPORTIONAL_FACTOR	217.218	w/r	Servo-pid gain value
17.5	FLOAT_ARRAY_2[5]	INTEGRAL_TIME	217.219	w/r	Servo-pid integral value
17.6	FLOAT_ARRAY_2[6]	DERIVATIVE_TIME	217	r	Not used
17.7	FLOAT_ARRAY_2[7]	NON_LINEAR_FACTOR	217	r	Not used
17.8	FLOAT_ARRAY_2[8]	DERIVATIVE_FACTOR	217	r	Not used
17.9	FLOAT_ARRAY_2[9]	PID_DEADBAND	217.220	w/r	PID deadband value
17.10	FLOAT_ARRAY_2[10]	RATE_UP_TIME	221.222	w/r	Rate time to open the valve
17.11	FLOAT_ARRAY_2[11]	RATE_DOWN_TIME	221.222	w/r	Rate time to close the valve
17.12	FLOAT_ARRAY_2[12]	TIGHT_SHUT_OFF	221.222	w/r	Tight-shut-off value
17.13	FLOAT_ARRAY_2[13]	SP_HIGH_LIMIT	221.222	w/r	SetPoint upper limit value
17.14	FLOAT_ARRAY_2[14]	SP_LOW_LIMIT	221.222	w/r	SetPoint lower limit value
17.15	FLOAT_ARRAY_2[15]	TIGHT_SHUT_OFF_DEADBAND	221.222	w/r	Tight-shut-off deadband value
17.16	FLOAT_ARRAY_2[16]	POWER_UP_SETPOINT	227.228	w/r	Power-up setpoint value
17.17	FLOAT_ARRAY_2[17]	SP_RATE_UP_TIME_DIAG_REF	233	r	SP_Rate Up_time Diagnosis References
17.18	FLOAT_ARRAY_2[18]	SP_RATE_DN_TIME_DIAG_REF	233	r	SP_Rate Dn_Time Diagnosis References
17.19	FLOAT_ARRAY_2[19]	PID_KP_DIAG_REF	233	r	Pid gain value
17.20	FLOAT_ARRAY_2[20]	PID_TR_DIAG_REF	233	r	Pid integral value
18.1	FLOAT_ARRAY_3[1]	LOWER_HALL	206	r	Lower Hall value
18.2	FLOAT_ARRAY_3[2]	UPPER_HALL	206	r	Upper Hall value
18.3	FLOAT_ARRAY_3[3]	MILEAGE	237.238	w/r	Total of the distance covered by the valve
18.4	FLOAT_ARRAY_3[4]	STROKES	237.238	w/r	Number of times the valve reach the maximum and the minimum travel
18.5	FLOAT_ARRAY_3[5]	REVERSALS	237.238	w/r	Number of times the valve change its motion
18.6	FLOAT_ARRAY_3[6]	UPPER_DA	206	r	Upper DA value
18.7	FLOAT_ARRAY_3[7]	LOWER_DA	206	r	Lower DA value
18.8	FLOAT_ARRAY_3[8]	MEAS_PV_CURR_LEVEL_GAIN	46	w	Measured PV Current for DAC Gain
18.9	FLOAT_ARRAY_3[9]	OPEN_TIME_DIAG_REF	236	r	Opening_Time Diagnosis References
18.10	FLOAT_ARRAY_3[10]	CLOSE_TIME_DIAG_REF	236	r	Closing_Time Diagnosis References
18.11	FLOAT_ARRAY_3[11]	OPEN_TIME	236	r	Open Time

HVT's ALLOCATION MAP for the FY301					
Index	**Parameter Name**	**HART Variable Name**	**HART Command**		**Description**
18.12	FLOAT_ARRAY_3[12]	CLOSE_TIME	236	r	Close Time
18.13	FLOAT_ARRAY_3[13]	VALUE_TEMPERATURE	155.3	w/r	Trim Temperature value
18.14	FLOAT_ARRAY_3[14]	X1	133.134	w/r	Table Coord X1
18.15	FLOAT_ARRAY_3[15]	X2	133.134	w/r	Table Coord X2
18.16	FLOAT_ARRAY_3[16]	X3	133.134	w/r	Table Coord X3
18.17	FLOAT_ARRAY_3[17]	X4	133.134	w/r	Table Coord X4
18.18	FLOAT_ARRAY_3[18]	X5	133.134	w/r	Table Coord X5
18.19	FLOAT_ARRAY_3[19]	X6	133.134	w/r	Table Coord X6
18.20	FLOAT_ARRAY_3[20]	X7	133.134	w/r	Table Coord X7
19.1	FLOAT_ARRAY_4[1]	X8	133.134	w/r	Table Coord X8
19.2	FLOAT_ARRAY_4[2]	X9	133.134	w/r	Table Coord X9
19.3	FLOAT_ARRAY_4[3]	X10	133.134	w/r	Table Coord X10
19.4	FLOAT_ARRAY_4[4]	X11	133.134	w/r	Table Coord X11
19.5	FLOAT_ARRAY_4[5]	X12	133.134	w/r	Table Coord X12
19.6	FLOAT_ARRAY_4[6]	X13	133.134	w/r	Table Coord X13
19.7	FLOAT_ARRAY_4[7]	X14	133.134	w/r	Table Coord X14
19.8	FLOAT_ARRAY_4[8]	X15	133.134	w/r	Table Coord X15
19.9	FLOAT_ARRAY_4[9]	X16	133.134	w/r	Table Coord X16
19.10	FLOAT_ARRAY_4[10]	Y1	133.134	w/r	Table Coord Y1
19.11	FLOAT_ARRAY_4[11]	Y2	133.134	w/r	Table Coord Y2
19.12	FLOAT_ARRAY_4[12]	Y3	133.134	w/r	Table Coord Y3
19.13	FLOAT_ARRAY_4[13]	Y4	133.134	w/r	Table Coord Y4
19.14	FLOAT_ARRAY_4[14]	Y5	133.134	w/r	Table Coord Y5
19.15	FLOAT_ARRAY_4[15]	Y6	133.134	w/r	Table Coord Y6
19.16	FLOAT_ARRAY_4[16]	Y7	133.134	w/r	Table Coord Y7
19.17	FLOAT_ARRAY_4[17]	Y8	133.134	w/r	Table Coord Y8
19.18	FLOAT_ARRAY_4[18]	Y9	133.134	w/r	Table Coord Y9
19.19	FLOAT_ARRAY_4[19]	Y10	133.134	w/r	Table Coord Y10
19.20	FLOAT_ARRAY_4[20]	Y11	133.134	w/r	Table Coord Y11
20.1	FLOAT_ARRAY_5[1]	Y12	133.134	w/r	Table Coord Y12
20.2	FLOAT_ARRAY_5[2]	Y13	133.134	w/r	Table Coord Y13
20.3	FLOAT_ARRAY_5[3]	Y14	133.134	w/r	Table Coord Y14
20.4	FLOAT_ARRAY_5[4]	Y15	133.134	w/r	Table Coord Y15
20.5	FLOAT_ARRAY_5[5]	Y16	133.134	w/r	Table Coord Y16
20.6	FLOAT_ARRAY_5[6]	TRIM_PRESSURE_OUT2_LOWER	241	w	Lower Trim Pressure for Out2
29	String_06	ACTUATOR_ID_NUMBER	183.184	w/r	Atuator identificatin number
30	String_07	VALVE_ID_NUMBER	185.186	w/r	Valve identification number
34	String_11	ORDERING_CODE	173.174	w/r	Factory device information

HVT's ALLOCATION MAP for the LD301

Index	Parameter Name	HART Variable Name	HART Command		Description
11	U8B_ARRAY_1				
11.1	U8B_ARRAY_1[1]	FLANGE_TYPE	128.129	w/r	Flange Type
11.2	U8B_ARRAY_1[2]	FLANGE_MATERIAL	128.129	w/r	Flange Material
11.3	U8B_ARRAY_1[3]	O_RING	128.129	w/r	O_Ring
11.4	U8B_ARRAY_1[4]	METER_INSTALLATION	128.129	w/r	Meter Installation
11.5	U8B_ARRAY_1[5]	DRAIN_VENT_MATERIAL	128.129	w/r	Drain Vent Material
11.6	U8B_ARRAY_1[6]	REMOTE_SEAL_TYPE	128.129	w/r	Remote Seal Type
11.7	U8B_ARRAY_1[7]	REMOTE_SEAL_FILL_FLUID	128.129	w/r	Remote Seal Fill Fluid
11.8	U8B_ARRAY_1[8]	REMOTE_SEAL_ISO_DIA_MATERIAL	128.129	w/r	Remote Seal Isolating Diaphragm Material
11.9	U8B_ARRAY_1[9]	NUMBER_REMOTE_SEAL	128.129	w/r	Number of Remote Seals
11.10	U8B_ARRAY_1[10]	SENSOR_FILL_FLUID	128.129	w/r	Sensor Fill Fluid
11.11	U8B_ARRAY_1[11]	SENSOR_ISO_DIA_MATERIAL	128.129	w/r	Sensor Isolating Diaphragm Material
11.12	U8B_ARRAY_1[12]	SENSOR_TYPE	128	r	Sensor Type
11.13	U8B_ARRAY_1[13]	SENSOR_RANGE	128	r	Sensor Range
11.14	U8B_ARRAY_1[14]	SENSOR_RANGE_UNIT	128	r	Sensor Range Unit
11.15	U8B_ARRAY_1[15]	SPECIAL_TRANSFER_FUNCTION	128	r	Special Transfer Function
11.16	U8B_ARRAY_1[16]	LOCAL_KEYS_MODE	128	r	Local Keys Mode
11.19	U8B_ARRAY_1[19]	TABLE_NUMBER_POINTS	133.135	w/r	Number of points in the table
12.1	U8B_ARRAY_2[1]	CONTROLLER_TYPE	136.137	r	LD301 Controller Type Code is 23
12.2	U8B_ARRAY_2[2]	POWER_UP_MODE	136.137	w/r	Indicates if LD301 Power Up (Automatic,Last PID Mode Selected,Manual)
12.3	U8B_ARRAY_2[3]	CONTROLLER_ACTION	136.137	w/r	Indicates if Controller Action is Reverse or Direct
12.4	U8B_ARRAY_2[4]	CONTROLLER_MODE	138.139	w/r	Indicates if LD301 mode operation is local or remote
12.5	U8B_ARRAY_2[5]	PROCESS_VARIABLE_UNIT	193	r	PV unit
12.6	U8B_ARRAY_2[6]	SET_POINT_UNIT	146.193.194.151	r	SP unit
12.7	U8B_ARRAY_2[7]	MANIPULATED_VARIABLE_UNIT	147.193	r	MV unit
12.8	U8B_ARRAY_2[8]	SET_POINT_TRACKING	140.141.193	w/r	Indicates if Set Point Tracking is on or off
12.9	U8B_ARRAY_2[9]	PID_MODE	140.141.193	w/r	Indicates if Pid Mode is Automatic or Manual
12.10	U8B_ARRAY_2[10]	ERROR_UNIT	140.193	r	Error Unit
12.14	U8B_ARRAY_2[14]	CUTOFF_MODE	156.191	w/r	Cutoff Mode
12.16	U8B_ARRAY_2[16]	CHARACT_MODE	160	r	Indicates if the characterization curve is enabled or disable
12.17	U8B_ARRAY_2[17]	NUMBER_CHARACT_POINT	160.161	w/r	Number of Characterization Points
12.19	U8B_ARRAY_2[19]	CHAR_AND_DISPLAY_MODE	163	w	Characterization Trim mode and Display
13.1	U8B_ARRAY_3[1]	FIRST_DISPLAY_CODE	164.165	w/r	First Display
13.2	U8B_ARRAY_3[2]	SECOND_DISPLAY_CODE	164.165	w/r	Second Display
13.3	U8B_ARRAY_3[3]	ZERO_SPAN_CHANGE_COUNT	166	r	Number of times the Zero Span was done
13.4	U8B_ARRAY_3[4]	FUNCTION_CHANGE_COUNT	166	r	Number of times the Function was done
13.5	U8B_ARRAY_3[5]	TRIM_4MA_CHANGE_COUNT	166	r	Number of times the 4 mA trim was done
13.6	U8B_ARRAY_3[6]	TRIM_20MA_CHANGE_COUNT	166	r	Number of times the 20 mA trim was done

HVT's ALLOCATION MAP for the LD301

Index	Parameter Name	HART Variable Name	HART Command		Description
13.7	U8B_ARRAY_3[7]	LOWER_TRIM_CHANGE_COUNT	166	r	Number of times the Lower Trim was done
13.8	U8B_ARRAY_3[8]	UPPER_TRIM_CHANGE_COUNT	166	r	Number of times the Upper Trim was done
13.9	U8B_ARRAY_3[9]	RESERVED	166	r	Reserved
13.10	U8B_ARRAY_3[10]	MODE_CHANGE_COUNT	166	r	Number of times the Mode was done
13.11	U8B_ARRAY_3[11]	CHARAC_TRIM_CHANGE_COUNT	166	r	Number of times the Characterization Trim was done
13.12	U8B_ARRAY_3[12]	LOCAL_ADJUST_CHANGE_COUNT	166	r	Number of times the Local Adjust was done
13.13	U8B_ARRAY_3[13]	WRITE_PROTECTION_CHANGE_COUNT	166	r	Number of times the Write Protection was done
13.14	U8B_ARRAY_3[14]	MULTIDROP_CHANGE_COUNT	166	r	Number of times the Multidrop was done
13.15	U8B_ARRAY_3[15]	PASSWORD_LEVEL_CHANGE_COUNT	166	r	Number of times the Password Level was done
13.16	U8B_ARRAY_3[16]	TOTALIZATION_CHANGE_COUNT	166	r	Number of times the Totalization was done
13.17	U8B_ARRAY_3[17]	COMMUNIC_PROTECT_MODE	169	w/r	Communication Write Protection mode
13.18	U8B_ARRAY_3[18]	LOCAL_ADJUST_JUMP	170	r	Local Adjust Jump
13.19	U8B_ARRAY_3[19]	LOCAL_ADJUST_SOFTWARE	170	r	Local Adjust Software
13.20	U8B_ARRAY_3[20]	LOCAL_ADJUST	170	r	Local Adjust
14.1	U8B_ARRAY_4[1]	JUMPER_SWITCH	170	r	Jumper Switch
14.2	U8B_ARRAY_4[2]	PV_ALARM_SELECT	203	w/r	Primary Variable Alarm Selection
14.3	U8B_ARRAY_4[3]	USER_UNIT	176.177	w/r	User Unit
14.4	U8B_ARRAY_4[4]	USER_UNIT_MODE	178.180	w/r	User Unit Mode
14.5	U8B_ARRAY_4[5]	TOTALIZER_MODE	183.185	w/r	Totalizer Mode
14.6	U8B_ARRAY_4[6]	TOTAL_UNIT	185.189.190	w/r	Total Unit
14.7	U8B_ARRAY_4[7]	EXE_PV_UPPER_RANGE_VALUE	36	w	Set Primary Variable Upper Range Value
14.8	U8B_ARRAY_4[8]	EXE_PV_LOWER_RANGE_VALUE	37	w	Set Primary Variable Lower Range Value
14.9	U8B_ARRAY_4[9]	EXE_EEPROM_CONTROL	39	w	Eeprom Control
14.10	U8B_ARRAY_4[10]	EXE_PV_ZERO	43	w	Set Primary Variable Zero
14.11	U8B_ARRAY_4[11]	EXE_RESET_TOTALIZER	184	w	Reset Totalizer
14.12	U8B_ARRAY_4[12]	READ_FROM_SENSOR	153	w	Read from Sensor
14.13	U8B_ARRAY_4[13]	FULL_WRITE_ON_SENSOR	153	w	Full Write On Sensor
14.14	U8B_ARRAY_4[14]	WRITE_ON_SENSOR	153	w	Write On Sensor
14.15	U8B_ARRAY_4[15]	FULL_READ_FROM_SENSOR	153	w	Full Read From Sensor
15.11	U8B_ARRAY_5[11]	TRIM_UNIT	130.131	r	Trim Unit
15.12	U8B_ARRAY_5[12]	MEASURED_POINT_UNIT	162	r	Measured Point Unit
15.13	U8B_ARRAY_5[13]	POWER_UP_SETPOINT_UNIT	151	r	Power Up Set Point Unit
15.14	U8B_ARRAY_5[14]	LOAD_RESTORE_TRIM	163	w	Load/Restore Trim
16.1	FLOAT_ARRAY_1[1]	PV_CURR_LEVEL_DAC_ZERO	45	w	Set value of the Trim Primary Variable Current Dac Zero
16.2	FLOAT_ARRAY_1[2]	UPPER_SENSOR	128	r	Upper Sensor Value
16.3	FLOAT_ARRAY_1[3]	LOWER_SENSOR	128	r	Lower Sensor Value
16.4	FLOAT_ARRAY_1[4]	UPPER_TRIM_POINT	130	w	Write Upper Sensor Trim Point
16.5	FLOAT_ARRAY_1[5]	LOWER_TRIM_POINT	131	w	Write Lower Sensor Trim Point
16.6	FLOAT_ARRAY_1[6]	MEASURED_POINT_1	160.162	w/r	Measured characterization Trim Curve Point 1

HVT's ALLOCATION MAP for the LD301

Index	Parameter Name	HART Variable Name	HART Command		Description
16.7	FLOAT_ARRAY_1[7]	MEASURED_POINT_2	160.162	w/r	Measured characterization Trim Curve Point 2
16.8	FLOAT_ARRAY_1[8]	MEASURED_POINT_3	160.162	w/r	Measured characterization Trim Curve Point 3
16.9	FLOAT_ARRAY_1[9]	MEASURED_POINT_4	160.162	w/r	Measured characterization Trim Curve Point 4
16.10	FLOAT_ARRAY_1[10]	MEASURED_POINT_5	160.162	w/r	Measured characterization Trim Curve Point 5
16.11	FLOAT_ARRAY_1[11]	PROCESS_VARIABLE	140	r	Process Variable (PV) Value
16.12	FLOAT_ARRAY_1[12]	SET_POINT	140.146	w/r	Set Point (SP) Value
16.13	FLOAT_ARRAY_1[13]	MANIPULATED_VARIABLE	140.147.193	w/r	Manipulated Variable (MV) Value
16.14	FLOAT_ARRAY_1[14]	ERROR	140.193	r	Error Value
16.15	FLOAT_ARRAY_1[15]	PROPORTIONAL_FACTOR(Kp)	142.143	w/r	Proportional Factor (Kp) Value
16.16	FLOAT_ARRAY_1[16]	INTEGRAL_TIME(Tr)	142.144	w/r	Integral Time (Tr) Value
16.17	FLOAT_ARRAY_1[17]	DERIVATIVE_TIME(Td)	142.145	w/r	Derivative Time (Td) Value
16.18	FLOAT_ARRAY_1[18]	NON_LINEAR_FACTOR(Knl)	142	r	Non Linear Factor (Knl) Value
16.19	FLOAT_ARRAY_1[19]	DERIVATIVE_FACTOR(DG)	142	r	Derivative Factor (DG) Value
16.20	FLOAT_ARRAY_1[20]	MV_HIGH_LIMIT	148.149	w/r	Manipulated Variable High Limit
17.1	FLOAT_ARRAY_2[1]	MV_LOW_LIMIT	148.149	w/r	Manipulated Variable Low Limit
17.2	FLOAT_ARRAY_2[2]	MV_RATE_CHANGE	148.149	w/r	Manipulated Variable Rate of Change
17.3	FLOAT_ARRAY_2[3]	POWER_UP_SETPOINT	150.151.194	w/r	Power Up Set Point
17.4	FLOAT_ARRAY_2[4]	POWER_UP_MANIPULATED_VARIABLE	150.151	w/r	Power Up Manipulated Variable
17.5	FLOAT_ARRAY_2[5]	CUTOFF_POINT	156.157	w/r	Square Root Cutoff
17.6	FLOAT_ARRAY_2[6]	TRIM_TEMP	155		Trim Temp
17.8	FLOAT_ARRAY_2[8]	USER_UNIT_UPPER	178.179	w/r	User Unit Upper Value
17.9	FLOAT_ARRAY_2[9]	USER_UNIT_LOWER	178.179	w/r	User Unit Lower Value
17.10	FLOAT_ARRAY_2[10]	TOTAL	185	r	Total Value
17.11	FLOAT_ARRAY_2[11]	MAXIMUM_FLOW	186.187	w/r	Maximum Flow Value
17.12	FLOAT_ARRAY_2[12]	TOTAL_UNIT_CONV_FACTOR	186.188	w/r	Total Unit Conversion Factor Value
17.13	FLOAT_ARRAY_2[13]	PV_CURR_LEVEL_DAC_GAIN	46	w	Set value of the Trim Primary Variable Current Dac Gain
17.14	FLOAT_ARRAY_2[14]	SPECIAL_UNIT_OUT	193	r	Special Unit Output Value
17.15	FLOAT_ARRAY_2[15]	SET_POINT_VARIABLE	193	r	Set Point Variable Value
17.16	FLOAT_ARRAY_2[16]	X1	133.134	w/r	Table Coord X1
17.17	FLOAT_ARRAY_2[17]	X2	133.134	w/r	Table Coord X2
17.18	FLOAT_ARRAY_2[18]	X3	133.134	w/r	Table Coord X3
17.19	FLOAT_ARRAY_2[19]	X4	133.134	w/r	Table Coord X4
17.20	FLOAT_ARRAY_2[20]	X5	133.134	w/r	Table Coord X5
18.1	FLOAT_ARRAY_3[1]	X6	133.134	w/r	Table Coord X6
18.2	FLOAT_ARRAY_3[2]	X7	133.134	w/r	Table Coord X7
18.3	FLOAT_ARRAY_3[3]	X8	133.134	w/r	Table Coord X8
18.4	FLOAT_ARRAY_3[4]	X9	133.134	w/r	Table Coord X9
18.5	FLOAT_ARRAY_3[5]	X10	133.134	w/r	Table Coord X10
18.6	FLOAT_ARRAY_3[6]	X11	133.134	w/r	Table Coord X11

HVT's ALLOCATION MAP for the LD301					
Index	**Parameter Name**	**HART Variable Name**	**HART Command**		**Description**
18.7	FLOAT_ARRAY_3[7]	X12	133.134	w/r	Table Coord X12
18.8	FLOAT_ARRAY_3[8]	X13	133.134	w/r	Table Coord X13
18.9	FLOAT_ARRAY_3[9]	X14	133.134	w/r	Table Coord X14
18.10	FLOAT_ARRAY_3[10]	X15	133.134	w/r	Table Coord X15
18.11	FLOAT_ARRAY_3[11]	X16	133.134	w/r	Table Coord X16
18.12	FLOAT_ARRAY_3[12]	Y1	133.134	w/r	Table Coord Y1
18.13	FLOAT_ARRAY_3[13]	Y2	133.134	w/r	Table Coord Y2
18.14	FLOAT_ARRAY_3[14]	Y3	133.134	w/r	Table Coord Y3
18.15	FLOAT_ARRAY_3[15]	Y4	133.134	w/r	Table Coord Y4
18.16	FLOAT_ARRAY_3[16]	Y5	133.134	w/r	Table Coord Y5
18.17	FLOAT_ARRAY_3[17]	Y6	133.134	w/r	Table Coord Y6
18.18	FLOAT_ARRAY_3[18]	Y7	133.134	w/r	Table Coord Y7
18.19	FLOAT_ARRAY_3[19]	Y8	133.134	w/r	Table Coord Y8
18.20	FLOAT_ARRAY_3[20]	Y9	133.134	w/r	Table Coord Y9
19.1	FLOAT_ARRAY_4[1]	Y10	133.134	w/r	Table Coord Y10
19.2	FLOAT_ARRAY_4[2]	Y11	133.134	w/r	Table Coord Y11
19.3	FLOAT_ARRAY_4[3]	Y12	133.134	w/r	Table Coord Y12
19.4	FLOAT_ARRAY_4[4]	Y13	133.134	w/r	Table Coord Y13
19.5	FLOAT_ARRAY_4[5]	Y14	133.134	w/r	Table Coord Y14
19.6	FLOAT_ARRAY_4[6]	Y15	133.134	w/r	Table Coord Y15
19.7	FLOAT_ARRAY_4[7]	Y16	133.134	w/r	Table Coord X16
19.8	FLOAT_ARRAY_4[8]	ACTUAL_POINT_1	160.162	w/r	Actual characterization Trim Curve Point 1
19.9	FLOAT_ARRAY_4[9]	ACTUAL_POINT_2	160.162	w/r	Actual characterization Trim Curve Point 2
19.10	FLOAT_ARRAY_4[10]	ACTUAL_POINT_3	160.162	w/r	Actual characterization Trim Curve Point 3
19.11	FLOAT_ARRAY_4[11]	ACTUAL_POINT_4	160.162	w/r	Actual characterization Trim Curve Point 4
19.12	FLOAT_ARRAY_4[12]	ACTUAL_POINT_5	160.162	w/r	Actual characterization Trim Curve Point 5
24	String_01	TOTAL_UNIT_STRING	189.190	w/r	Total Unit String
25	String_02	USER_UNIT_STRING	176.177	w/r	User Unit String
34	String_11	ORDERING_CODE	173.174	w/r	Ordering Code

HVT's ALLOCATION MAP for the TT301

Index	Parameter Name	HART Variable Name	HART Command		Description
11	U8B_ARRAY_1				
11.1	U8B_ARRAY_1[1]	EEPROM_CONTROL	39	w	Eeprom Control
11.2	U8B_ARRAY_1[2]	PV_XMTR	50.51	w/r	Transmitter Variable Assigned to the Primary Variable
11.3	U8B_ARRAY_1[3]	SV_XMTR	50.51	w/r	Transmitter Variable Assigned to the Secondary Variable
11.4	U8B_ARRAY_1[4]	TV_XMTR	50.51	w/r	Transmitter Variable Assigned to the Tertiary Variable
11.5	U8B_ARRAY_1[5]	4TH_XMTR	50.51	w/r	Transmitter Variable Assigned to the 4th Variable
11.7	U8B_ARRAY_1[7]	SENSOR TYPE	130.131	w/r	Sensor type
11.8	U8B_ARRAY_1[8]	NUMBER_WIRES	130.131	w/r	Number of wires
11.9	U8B_ARRAY_1[9]	METER_INST	130	r	Meter installation (installed or not)
11.10	U8B_ARRAY_1[10]	LIN_MODE	130	r	Linearization mode (linear with temperature)
11.11	U8B_ARRAY_1[11]	USER_CHARACTERIZATION	132.135	w/r	Calibration source (factory or user)
11.12	U8B_ARRAY_1[12]	SENSOR_UNIT	132	r	Sensor Unit
11.14	U8B_ARRAY_1[14]	CONTROLLER_MODE	136.138.139.184	w/r	Controler Mode (transmitter or controller)
11.15	U8B_ARRAY_1[15]	POWER_UP	136.137	w/r	Power Up configuration(Automatic,Last PID Mode Selected or Manual)
11.16	U8B_ARRAY_1[16]	CONTROLLER_ACTION	136.137	w/r	Controller Action (Reverse or Direct)
11.18	U8B_ARRAY_1[18]	PROCESS_VARIABLE_UNIT	140	r	PV unit
11.19	U8B_ARRAY_1[19]	SET_POINT_UNIT	140.146	w/r	SP unit
11.20	U8B_ARRAY_1[20]	MANIPULATED_VARIABLE_UNIT	140.147	w/r	MV unit
12.1	U8B_ARRAY_2[1]	SETPOINT_TRACKING	140.141.184	w/r	SP tracking
12.2	U8B_ARRAY_2[2]	PID_MODE	140.141.184	w/r	Pid mode (Automatic or Manual)
12.3	U8B_ARRAY_2[3]	ERROR_UNIT	140	r	Error Unit
12.4	U8B_ARRAY_2[4]	SETPOINT_GENERATOR_MODE	152.154.184	w/r	SP generator mode (pause or running)
12.5	U8B_ARRAY_2[5]	SETPOINT_TIME_GENERATOR_MODE	152.155.184	w/r	SP time generator mode (by curve or extern sp)
12.7	U8B_ARRAY_2[7]	NUMBER_POINTS	156.157	w/r	Number of points in the curve
12.9	U8B_ARRAY_2[9]	ALARM_0_ACTION	159.160	w/r	Alarm 0 Action (on or off)
12.10	U8B_ARRAY_2[10]	ALARM_1_ACTION	159.160	w/r	Alarm 1 action (low, high or disable)
12.11	U8B_ARRAY_2[11]	ALARM_2_ACTION	159.160	w/r	Alarm 2 action (low, high or disable)
12.12	U8B_ARRAY_2[12]	ALARM_0_STATUS	159	r	Alarm 0 Status (on or off)
12.13	U8B_ARRAY_2[13]	ALARM_1_STATUS	159	r	Alarm 1 Status (on or off)
12.14	U8B_ARRAY_2[14]	ALARM_2_STATUS	159	r	Alarm 2 Status (on or off)
12.15	U8B_ARRAY_2[15]	ALARM_0_ACKNOWLEDGE	159.161	w/r	Acknowledge alarm 0
12.16	U8B_ARRAY_2[16]	ALARM_1_ACKNOWLEDGE	159.161	w/r	Acknowledge alarm 1
12.17	U8B_ARRAY_2[17]	ALARM_2_ACKNOWLEDGE	159.161	w/r	Acknowledge alarm 2
12.18	U8B_ARRAY_2[18]	FAIL_SAFE_MODE	162.163	w/r	Fail_safe mode (low or high)
12.19	U8B_ARRAY_2[19]	PV_DISPLAY	164.165	w/r	PV for Display indication
12.20	U8B_ARRAY_2[20]	SV_DISPLAY	164.165	w/r	SV for Display indication
13.1	U8B_ARRAY_3[1]	COUNT_CHANGE_ZERO_SPAN	166	r	Number of time the zero and span trim was done
13.2	U8B_ARRAY_3[2]	COUNT_CHANGE_FUNCTION	166	r	Number of time the function was changed

HVT's ALLOCATION MAP for the TT301					
Index	Parameter Name	HART Variable Name	HART Command		Description
13.3	U8B_ARRAY_3[3]	COUNT_CHANGE_TRIM_4MA	166	r	Number of times the 4 mA trim was done
13.4	U8B_ARRAY_3[4]	COUNT_CHANGE_TRIM_20MA	166	r	Number of time the 20 mA trim was done
13.5	U8B_ARRAY_3[5]	COUNT_CHANGE_TRIM_USER	166	r	Number of time the user trim was done
13.6	U8B_ARRAY_3[6]	COUNT_CHANGE_BURNOUT	166	r	Number of times the Burnout action was changed
13.7	U8B_ARRAY_3[7]	COUNT_CHANGE_SENSOR	166	r	Number of time the sensor was changed
13.8	U8B_ARRAY_3[8]	COUNT_CHANGE_OPERATION_MODE	166	r	Number of time the mode was changed
13.9	U8B_ARRAY_3[9]	COUNT_CHANGE_LOCAL_ADJUST	166	r	Number of time the local adjust protection was changed
13.10	U8B_ARRAY_3[10]	COUNT_CHANGE_COMMUNICATION	166	r	Number of time the communication protection was changed
13.11	U8B_ARRAY_3[11]	COUNT_CHANGE_MULTIDROP	166	r	Number of time the multidrop address was changed
13.12	U8B_ARRAY_3[12]	COUNT_CHANGE_PASSWORD	166	r	Number of time the password was changed
13.13	U8B_ARRAY_3[13]	COMMUNICATION_PROTECTION	168.169	w/r	Communication Protection (enable or disable)
13.14	U8B_ARRAY_3[14]	LOCAL_ADJUST_PROTECTION	168.169	w/r	Local Adjust (enable or disable)
13.15	U8B_ARRAY_3[15]	LOCAL_ADJUST_MODE	170	r	Local Adjust mode (Simple or Complete)
13.16	U8B_ARRAY_3[16]	UNIT_CODE	175.176	w/r	Special Unit Code
13.17	U8B_ARRAY_3[17]	INPUT_UNIT_CODE	179.180	w/r	Special Unit Characterization
13.18	U8B_ARRAY_3[18]	SPECIAL_SENSOR_GAIN	179.180	w/r	Special sensor gain
13.19	U8B_ARRAY_3[19]	SPECIAL_SENSOR_CONNECTION	179.180	w/r	Special sensor connection
13.20	U8B_ARRAY_3[20]	MV_SPECIAL_COLD_JUNCTION	186.187	w/r	Special Cold Junction mode (enable or disable)
14.1	U8B_ARRAY_4[1]	EXE_PV_UPPER_RANGE_VALUE	36	w	Set Primary Variable Upper Range Value
14.2	U8B_ARRAY_4[2]	EXE_PV_LOWER_RANGE_VALUE	37	w	Set Primary Variable Lower Range Value
14.3	U8B_ARRAY_4[3]	FACTORY_TRIM	185	w	Factory Trim
15.11	U8B_ARRAY_5[11]	CAL_POINT_UNIT	133.134	w	Calibration unit
15.12	U8B_ARRAY_5[12]	RANGE_UNITS	158	w	Sensor Range Unit
16.1	FLOAT_ARRAY_1[1]	MEAS_PV_CURR_LEVEL	45	w	Set value of the Trim Primary Variable Current to Dac Zero
16.2	FLOAT_ARRAY_1[2]	UPPER_CAL_POINT_LIMIT	132	r	Upper calibration point limit
16.3	FLOAT_ARRAY_1[3]	LOWER_CAL_POINT_LIMIT	132	r	Lower calibration point limit
16.4	FLOAT_ARRAY_1[4]	CAL_POINT_SPAN	133	w	User span calibration point
16.5	FLOAT_ARRAY_1[5]	PROCESS_VARIABLE	140.184	r	PV Value
16.6	FLOAT_ARRAY_1[6]	SET_POINT	140.146.184	w/r	SP Value
16.7	FLOAT_ARRAY_1[7]	MANIPULATED_VARIABLE	140.147	w/r	MV Value
16.8	FLOAT_ARRAY_1[8]	ERROR	140	r	Error Value
16.9	FLOAT_ARRAY_1[9]	PROPORTIONAL_FACTOR(KP)	142.143	w/r	Proportional Factor Value
16.10	FLOAT_ARRAY_1[10]	INTEGRAL_TIME(TR)	142.144	w/r	Integral Time Value
16.11	FLOAT_ARRAY_1[11]	DERIVATIVE_TIME(TD)	142.145	w/r	Derivative Time Value
16.12	FLOAT_ARRAY_1[12]	NON_LINEAR_FATOR(KNL)	142	r	Non Linear Factor Value (always 0)
16.13	FLOAT_ARRAY_1[13]	DERIVATIVE_FACTOR(DG)	142	r	Derivative Factor Value (always 10)
16.14	FLOAT_ARRAY_1[14]	MANIPULATED_VARIABLE_HIGH_LIMIT	148.149	w/r	MV maximum limit
16.15	FLOAT_ARRAY_1[15]	MANIPULATED_VARIABLE_LOW_LIMIT	148.149	w/r	MV minimum limit
16.16	FLOAT_ARRAY_1[16]	MANIPULATED_VARIABLE_MAXIMUM	148.149	w/r	MV maximum rate of changing

HVT's ALLOCATION MAP for the TT301					
Index	**Parameter Name**	**HART Variable Name**	**HART Command**		**Description**
16.17	FLOAT_ARRAY_1[17]	POWER_ON_SETPOINT	150.151	w/r	Power_On SP
16.18	FLOAT_ARRAY_1[18]	POWER_ON_MANUAL_OUTPUT	150.151	w/r	Fail safe value
16.19	FLOAT_ARRAY_1[19]	SETPOINT_TIME	152.153.184	w/r	SP time
17.2	FLOAT_ARRAY_2[2]	UPPER_RANGE_VALUE	158	w	Save Upper Range Value to eeprom
17.3	FLOAT_ARRAY_2[3]	LOWER_RANGE_VALUE	158	w	Save Lower Range Value to eeprom
17.4	FLOAT_ARRAY_2[4]	ALARME_1_VALUE	159.160	w/r	Alarm 1 value limit
17.5	FLOAT_ARRAY_2[5]	ALARME_2_VALUE	159.160	w/r	Alarm 2 value limit
17.6	FLOAT_ARRAY_2[6]	SPECIAL_SENSOR_UPPER_RANGE_LIMIT	177.178	w/r	Special sensor upper range limit
17.7	FLOAT_ARRAY_2[7]	SPECIAL_SENSOR_LOWER_RANGE_LIMIT	177.178	w/r	Special sensor lower range limit
17.8	FLOAT_ARRAY_2[8]	SPECIAL_SENSOR_MINIMUM_SPAN	177.178	w/r	Special sensor minimum span limit
17.9	FLOAT_ARRAY_2[9]	SPECIAL_SENSOR_HIGH_LIMIT	177.178	w/r	Special sensor high limit
17.10	FLOAT_ARRAY_2[10]	SPECIAL_SENSOR_LOW_LIMIT	177.178	w/r	Special sensor low limit
17.11	FLOAT_ARRAY_2[11]	SPECIAL_SENSOR_OHMS_MV_UPPER_LIMIT	179.180	w/r	Special sensor mv or ohm upper limit
17.12	FLOAT_ARRAY_2[12]	SPECIAL_SENSOR_OHMS_MV_LOWER_LIMIT	179.180	w/r	Special sensor mv or ohm lower limit
17.13	FLOAT_ARRAY_2[13]	OUTPUT_VARIABLE	184	r	Output Variable (OUT) Value
17.14	FLOAT_ARRAY_2[14]	MEAS_PV_CURR_LEVEL_GAIN	46	w	Set value of the Trim Primary Variable Current Dac Gain
17.15	FLOAT_ARRAY_2[15]	CAL_POINT_ZERO	134	w	Zero user calibration point
17.16	FLOAT_ARRAY_2[16]	SP_CURVE_X1	156.157	w/r	Coordinate X of set point curve index 1
17.17	FLOAT_ARRAY_2[17]	SP_CURVE_X2	156.157	w/r	Coordinate X of set point curve index 2
17.18	FLOAT_ARRAY_2[18]	SP_CURVE_X3	156.157	w/r	Coordinate X of set point curve index 3
17.19	FLOAT_ARRAY_2[19]	SP_CURVE_X4	156.157	w/r	Coordinate X of set point curve index 4
17.20	FLOAT_ARRAY_2[20]	SP_CURVE_X5	156.157	w/r	Coordinate X of set point curve index 5
18.1	FLOAT_ARRAY_3[1]	SP_CURVE_X6	156.157	w/r	Coordinate X of set point curve index 6
18.2	FLOAT_ARRAY_3[2]	SP_CURVE_X7	156.157	w/r	Coordinate X of set point curve index 8
18.3	FLOAT_ARRAY_3[3]	SP_CURVE_X8	156.157	w/r	Coordinate X of set point curve index 8
18.4	FLOAT_ARRAY_3[4]	SP_CURVE_X9	156.157	w/r	Coordinate X of set point curve index 9
18.5	FLOAT_ARRAY_3[5]	SP_CURVE_X10	156.157	w/r	Coordinate X of set point curve index 10
18.6	FLOAT_ARRAY_3[6]	SP_CURVE_X11	156.157	w/r	Coordinate X of set point curve index 11
18.7	FLOAT_ARRAY_3[7]	SP_CURVE_X12	156.157	w/r	Coordinate X of set point curve index 12
18.8	FLOAT_ARRAY_3[8]	SP_CURVE_X13	156.157	w/r	Coordinate X of set point curve index 13
18.9	FLOAT_ARRAY_3[9]	SP_CURVE_X14	156.157	w/r	Coordinate X of set point curve index 14
18.10	FLOAT_ARRAY_3[10]	SP_CURVE_X15	156.157	w/r	Coordinate X of set point curve index 15
18.11	FLOAT_ARRAY_3[11]	SP_CURVE_X16	156.157	w/r	Coordinate X of set point curve index 16
18.12	FLOAT_ARRAY_3[12]	SP_CURVE_Y1	156.157	w/r	Coordinate Y of set point curve index 1
18.13	FLOAT_ARRAY_3[13]	SP_CURVE_Y2	156.157	w/r	Coordinate Y of set point curve index 2
18.14	FLOAT_ARRAY_3[14]	SP_CURVE_Y3	156.157	w/r	Coordinate Y of set point curve index 3
18.15	FLOAT_ARRAY_3[15]	SP_CURVE_Y4	156.157	w/r	Coordinate Y of set point curve index 4
18.16	FLOAT_ARRAY_3[16]	SP_CURVE_Y5	156.157	w/r	Coordinate Y of set point curve index 5
18.17	FLOAT_ARRAY_3[17]	SP_CURVE_Y6	156.157	w/r	Coordinate Y of set point curve index 6

HVT's ALLOCATION MAP for the TT301					
Index	**Parameter Name**	**HART Variable Name**	**HART Command**		**Description**
18.18	FLOAT_ARRAY_3[18]	SP_CURVE_Y7	156.157	w/r	Coordinate Y of set point curve index 7
18.19	FLOAT_ARRAY_3[19]	SP_CURVE_Y8	156.157	w/r	Coordinate Y of set point curve index 8
18.20	FLOAT_ARRAY_3[20]	SP_CURVE_Y9	156.157	w/r	Coordinate Y of set point curve index 9
19.1	FLOAT_ARRAY_4[1]	SP_CURVE_Y10	156.157	w/r	Coordinate Y of set point curve index 10
19.2	FLOAT_ARRAY_4[2]	SP_CURVE_Y11	156.157	w/r	Coordinate Y of set point curve index 11
19.3	FLOAT_ARRAY_4[3]	SP_CURVE_Y12	156.157	w/r	Coordinate Y of set point curve index 12
19.4	FLOAT_ARRAY_4[4]	SP_CURVE_Y13	156.157	w/r	Coordinate Y of set point curve index 13
19.5	FLOAT_ARRAY_4[5]	SP_CURVE_Y14	156.157	w/r	Coordinate Y of set point curve index 14
19.6	FLOAT_ARRAY_4[6]	SP_CURVE_Y15	156.157	w/r	Coordinate Y of set point curve index 15
19.7	FLOAT_ARRAY_4[7]	SP_CURVE_Y16	156.157	w/r	Coordinate Y of set point curve index 16
22.1	U32B_ARRAY_1[1]	DEVICE_ID	181	w	Device Identification Number
24	String_01	UNIT_STRING	175.176	w/r	8 characters general use string
34	String_11	SMAR_ORDER_CODE	173.174	w/r	Factory device information

HVT's ALOCATION MAP for the DT301					
Index	**Parameter Name**	**HART Variable Name**	**HART Command**		**Description**
11.1	U8B_ARRAY_1[1]	EEPROM_CONTROL	39	w	Eeprom Control
11.2	U8B_ARRAY_1[2]	FLANGE_TYPE	128.129	w/r	Flange Type
11.3	U8B_ARRAY_1[3]	PROBE_MATERIAL	128.129	w/r	Probe Material
11.4	U8B_ARRAY_1[4]	O-RING_MATERIAL	128.129	w/r	O_Ring
11.5	U8B_ARRAY_1[5]	METER_INSTALLATION	128	r	Meter Installation
11.6	U8B_ARRAY_1[6]	INSTALLATION_TYPE	128.129	w/r	Installation Type
11.7	U8B_ARRAY_1[7]	PROBE_DIAPHRAGM	128.129	w/r	Probe Diaphragm
11.8	U8B_ARRAY_1[8]	PROBE_FLUID	128.129	w/r	Probe Fluid
11.9	U8B_ARRAY_1[9]	DIAPHRAGM_MATERIAL	128.129	w/r	Diaphragm Material
11.10	U8B_ARRAY_1[10]	ELETRIC_CONNECTION	128.129	w/r	Electric Connection
11.11	U8B_ARRAY_1[11]	SENSOR_FILL_FLUID	128	r	Sensor Fill Fluid
11.12	U8B_ARRAY_1[12]	ISOLATION_DIAPHRAGM	128	r	Isolating Diaphragm
11.13	U8B_ARRAY_1[13]	SENSOR_TYPE	128	r	Sensor Type
11.14	U8B_ARRAY_1[14]	SENSOR_RANGE	128	r	Sensor Range
11.15	U8B_ARRAY_1[15]	SENSOR_RANGE_UNIT	128	r	Sensor Range Unit
11.16	U8B_ARRAY_1[16]	SPECIAL_TRANSFER_FUNCTION	128	r	Special Transfer Function
11.17	U8B_ARRAY_1[17]	RANGE_CODE_DT	128.129	w/r	Range Code
11.19	U8B_ARRAY_1[19]	LOCAL_KEYS_MODE	132		
12.1	U8B_ARRAY_2[1]	TABLE_NUMBER_POINTS	133.135	w/r	Number of points in the table
12.2	U8B_ARRAY_2[2]	OPERATION_CODE_W_S	153	w	Write On Sensor (Simple)
12.3	U8B_ARRAY_2[3]	OPERATION_CODE_W_C	153	w	Write On Sensor (Complete)
12.4	U8B_ARRAY_2[4]	OPERATION_CODE_R_S	153	w	Read From Sensor (Simple)
12.5	U8B_ARRAY_2[5]	OPERATION_CODE_R_C	153	w	Read From Sensor (Complete)
12.8	U8B_ARRAY_2[8]	FIRST_DISPLAY_CODE	164.165	w/r	First Display
12.9	U8B_ARRAY_2[9]	SECOND_DISPLAY_CODE	164.165	w/r	Second Display
12.10	U8B_ARRAY_2[10]	COUNT_CHANG_ZERO_SPAN	166	r	Number of times the Zero Span was done
12.11	U8B_ARRAY_2[11]	COUNT_CHANGE_FUNCTION	166	r	Number of times the Function was done
12.12	U8B_ARRAY_2[12]	COUNT_CHANGE_TRIM_4	166	r	Number of times the 4 mA trim was done
12.13	U8B_ARRAY_2[13]	COUNT_CHANGE_TRIM_20	166	r	Number of times the 20 mA trim was done
12.14	U8B_ARRAY_2[14]	COUNT_CHANGE_TRIM_LOWER	166	r	Number of times the Lower Trim was done
12.15	U8B_ARRAY_2[15]	COUNT_CHANGE_TRIM_UPPER	166	r	Number of times the Upper Trim was done
12.16	U8B_ARRAY_2[16]	RESERVED	166	r	Reserved
12.17	U8B_ARRAY_2[17]	COUNT_CHANGE_MODE	166	r	Number of times the Mode was done
12.18	U8B_ARRAY_2[18]	COUNT_CHANGE_CHARACTERIZ	166	r	Number of times the characterization was done
12.19	U8B_ARRAY_2[19]	COUNT_CHANGE_LOCAL_ADJUST	166	r	Number of times the local adjust was done
12.20	U8B_ARRAY_2[20]	COUNT_CHANGE_MULTIDROP	166	r	Number of times the multidrop was done
13.1	U8B_ARRAY_3[1]	COUNT_CHANGE_PASSWORD	166	r	Number of times the password was done
13.2	U8B_ARRAY_3[2]	SELF_CALIBRATION_PRESSURE_TRIM	168	w	used to self calibration trim
13.3	U8B_ARRAY_3[3]	COMMUNIC_PROTECT_MODE	169.15	w/r	Communication Write Protection mode

HVT's ALOCATION MAP for the DT301					
Index	**Parameter Name**	**HART Variable Name**	**HART Command**		**Description**
13.5	U8B_ARRAY_3[5]	USER_UNIT	176.177.178	w/r	User Unit
13.7	U8B_ARRAY_3[7]	FAIL_SAFE_MODE	203.15	w/r	Fail Safe Mode
13.10	U8B_ARRAY_3[10]	EXE_PV_ZERO	43	w	Set Primary Variable Zero
15.11	U8B_ARRAY_5[11]	DISABLE_PRESSURE_TRIM	156.157	w	Disable Pressure Trim
16.1	FLOAT_ARRAY_1[1]	MEAS_PV_CURRENT_LEVEL_ZERO	45	w	Set value of the Trim Primary Variable Current Dac Zero
16.2	FLOAT_ARRAY_1[2]	UPPER_SENSOR_TRIM	128	w	Upper Sensor Value
16.3	FLOAT_ARRAY_1[3]	LOWER_SENSOR_TRIM	128	w	Lower Sensor Value
16.4	FLOAT_ARRAY_1[4]	UPPER_TRIM_POINT	130	w	Write Upper Sensor Trim Point
16.5	FLOAT_ARRAY_1[5]	LOWER_TRIM_UNIT	131	w	Write Lower Sensor Trim Point
16.11	FLOAT_ARRAY_1[11]	GL	142.143	w/r	Parameter "gl",
16.12	FLOAT_ARRAY_1[12]	Ap	142	r	Parameter "Ap"
16.13	FLOAT_ARRAY_1[13]	HO	142	r	Parameter "Ho"
16.14	FLOAT_ARRAY_1[14]	ALPHA	142	r	Parameter "alpha"
16.15	FLOAT_ARRAY_1[15]	TEMP_MST_ZERO	154	r	Temperature Mst Zero
16.16	FLOAT_ARRAY_1[16]	TEMP_MST_SPAN	154	r	Temperature Mst Span
16.17	FLOAT_ARRAY_1[17]	T_ZERO	154	r	T Zero
16.18	FLOAT_ARRAY_1[18]	TEMP_TMED	154	r	Temperature Tmed
16.19	FLOAT_ARRAY_1[19]	CONCENT_UPPER	156	w	Concentration Trim Upper
16.20	FLOAT_ARRAY_1[20]	USER_UNIT_UPPER_VALUE	178.179	w/r	User Unit Upper Value
17.1	FLOAT_ARRAY_2[1]	USER_UNIT_LOWER_VALUE	178.179	w/r	User Unit Lower Value
17.2	FLOAT_ARRAY_2[2]	UPPER_POLYNOMIAL_LIMIT	183.184	w/r	Upper Polynomial Limit
17.3	FLOAT_ARRAY_2[3]	LOWER_POLYNOMIAL_LIMIT	183.184	w/r	Lower Polynomial Limit
17.4	FLOAT_ARRAY_2[4]	AS0	185.186	w/r	As0 value
17.5	FLOAT_ARRAY_2[5]	AS1	185.186	w/r	As1value
17.6	FLOAT_ARRAY_2[6]	AS2	185.186	w/r	As2value
17.7	FLOAT_ARRAY_2[7]	AS3	185.186	w/r	As3 value
17.8	FLOAT_ARRAY_2[8]	AS4	185.186	w/r	As4 value
17.9	FLOAT_ARRAY_2[9]	AS5	185.186	w/r	As5 value
17.10	FLOAT_ARRAY_2[10]	CAL_PRESSURE_VALUE	194	w	Pressure value
17.11	FLOAT_ARRAY_2[11]	CAL_TEMPERATURE_VALUE	194	w	Temperature value
17.12	FLOAT_ARRAY_2[12]	MEAS_PV_CURRENT_LEVEL_GAIN	46	w	Set value of the Trim Primary Variable Current Dac Gain
17.13	FLOAT_ARRAY_2[13]	CONCENT_LOWER	157	w	Concentration Trim Lower
17.14	FLOAT_ARRAY_2[14]	X1	133.134	w/r	Table Coord X1
17.15	FLOAT_ARRAY_2[15]	X2	133.134	w/r	Table Coord X2
17.16	FLOAT_ARRAY_2[16]	X3	133.134	w/r	Table Coord X3
17.17	FLOAT_ARRAY_2[17]	X4	133.134	w/r	Table Coord X4
17.18	FLOAT_ARRAY_2[18]	X5	133.134	w/r	Table Coord X5
17.19	FLOAT_ARRAY_2[19]	X6	133.134	w/r	Table Coord X6
17.20	FLOAT_ARRAY_2[20]	X7	133.134	w/r	Table Coord X7
18.1	FLOAT_ARRAY_3[1]	X8	133.134	w/r	Table Coord X8

HVT's ALOCATION MAP for the DT301

Index	Parameter Name	HART Variable Name	HART Command		Description
18.2	FLOAT_ARRAY_3[2]	X9	133.134	w/r	Table Coord X9
18.3	FLOAT_ARRAY_3[3]	X10	133.134	w/r	Table Coord X10
18.4	FLOAT_ARRAY_3[4]	X11	133.134	w/r	Table Coord X11
18.5	FLOAT_ARRAY_3[5]	X12	133.134	w/r	Table Coord X12
18.6	FLOAT_ARRAY_3[6]	X13	133.134	w/r	Table Coord X13
18.7	FLOAT_ARRAY_3[7]	X14	133.134	w/r	Table Coord X14
18.8	FLOAT_ARRAY_3[8]	X15	133.134	w/r	Table Coord X15
18.9	FLOAT_ARRAY_3[9]	X16	133.134	w/r	Table Coord X16
18.10	FLOAT_ARRAY_3[10]	Y1	133.134	w/r	Table Coord Y1
18.11	FLOAT_ARRAY_3[11]	Y2	133.134	w/r	Table Coord Y2
18.12	FLOAT_ARRAY_3[12]	Y3	133.134	w/r	Table Coord Y3
18.13	FLOAT_ARRAY_3[13]	Y4	133.134	w/r	Table Coord Y4
18.14	FLOAT_ARRAY_3[14]	Y5	133.134	w/r	Table Coord Y5
18.15	FLOAT_ARRAY_3[15]	Y6	133.134	w/r	Table Coord Y6
18.16	FLOAT_ARRAY_3[16]	Y7	133.134	w/r	Table Coord Y7
18.17	FLOAT_ARRAY_3[17]	Y8	133.134	w/r	Table Coord Y8
18.18	FLOAT_ARRAY_3[18]	Y9	133.134	w/r	Table Coord Y9
18.19	FLOAT_ARRAY_3[19]	Y10	133.134	w/r	Table Coord Y10
18.20	FLOAT_ARRAY_3[20]	Y11	133.134	w/r	Table Coord Y11
19.1	FLOAT_ARRAY_4[1]	Y12	133.134	w/r	Table Coord Y12
19.2	FLOAT_ARRAY_4[2]	Y13	133.134	w/r	Table Coord Y13
19.3	FLOAT_ARRAY_4[3]	Y14	133.134	w/r	Table Coord Y14
19.4	FLOAT_ARRAY_4[4]	Y15	133.134	w/r	Table Coord Y15
19.5	FLOAT_ARRAY_4[5]	Y16	133.134	w/r	Table Coord X16
24	String_01	USER_UNIT_STRING	176.177	w/r	User Unit String
34	String_11	SMAR_ORDER_CODE	173.174	w/r	Ordering Code

HVT's ALOCATION MAP para o TP301

Index	Parameter Name	HART Variable Name	HART Command		Description
11.1	U8B_ARRAY_1[1]	EEPROM_CONTROL	39	w	Eeprom Control
11.2	U8B_ARRAY_1[2]	DISPLAY_CONNECTED	128	r	display connected
11.3	U8B_ARRAY_1[3]	LOCAL_KEYS_MODE	132	r	Local Keys Control Mode
11.4	U8B_ARRAY_1[4]	TRANSDUCER_ACTION	156.157	w/r	Transducer Action
11.5	U8B_ARRAY_1[5]	FIRST_DISPLAY_CODE	164.165	w/r	First Display
11.6	U8B_ARRAY_1[6]	SECOND_DISPLAY_CODE	164.165	w/r	Second Display
11.7	U8B_ARRAY_1[7]	COUNT_FUNCTION	166	r	Number of times the Function was done
11.8	U8B_ARRAY_1[8]	COUNT_UPPER_POSITION	166	r	Number of times the Upper Position was done
11.9	U8B_ARRAY_1[9]	COUNT_LOWER_POSITION	166	r	Number of times the Lower Position was done
11.10	U8B_ARRAY_1[10]	COUNT_DIRECT_REVERSE	166	r	Number of times the Direct/Reverse was done
11.11	U8B_ARRAY_1[11]	COUNT_TRIM_4MA	166	r	Number of times the 4 mA trim was done
11.12	U8B_ARRAY_1[12]	COUNT_TRIM_20MA	166	r	Number of times the 20 mA trim was done
11.13	U8B_ARRAY_1[13]	COUNT_PASSWORD	166	r	Number of times the password was done
11.14	U8B_ARRAY_1[14]	COUNT_CONF_LEVEL	166	r	Number of times the Conf Level was done
11.15	U8B_ARRAY_1[15]	COUNT_WRITE_PROTECT	166	r	Number of times the Write Protected was done
11.16	U8B_ARRAY_1[16]	COUNT_EQUIPAMENT_DATA	166	r	Number of times the Equipament Data was done
11.17	U8B_ARRAY_1[17]	COUNT_FACTORY	166	r	Number of times the Factory was done
11.18	U8B_ARRAY_1[18]	COMMUNIC_PROTECT_MODE	169	w	Communication Write Protection mode
11.19	U8B_ARRAY_1[19]	PROTECTION_BYTE	170	r	read if the write protection is controlled by software or hardware key
11.20	U8B_ARRAY_1[20]	USER_UNIT_CODE	176.177	w/r	User Unit
12.1	U8B_ARRAY_2[1]	FLAG_USER_UNIT	178	r	Flag User Unit
12.2	U8B_ARRAY_2[2]	USER_UNIT_MODE	180	w	User Unit Mode
12.3	U8B_ARRAY_2[3]	FAIL_SAFE_MODE	203.15	w/r	Fail Safe Mode
12.4	U8B_ARRAY_2[4]	EXE_TRIM_LOWER_POSITION	130	w	Execute Trim Lower Position
12.5	U8B_ARRAY_2[5]	EXE_TRIM_UPPER_POSITION	131	w	Execute Trim Upper Position
16.1	FLOAT_ARRAY_1[1]	MEAS_PV_CURRENT_LEVEL_ZERO	45	w	Set value of the Trim Primary Variable Current Dac Zero
16.2	FLOAT_ARRAY_1[2]	USER_UNIT_UPPER_VALUE	178.179	w/r	User Unit Upper Value
16.3	FLOAT_ARRAY_1[3]	USER_UNIT_LOWER_VALUE	178.179	w/r	User Unit Lower Value
16.4	FLOAT_ARRAY_1[4]	MEAS_PV_CURRENT_LEVEL_GAIN	46	w	Set value of the Trim Primary Variable Current Dac Gain
24	String_01	USER_UNIT_STRING	176.177	w/r	User Unit String
34	String_11	SMAR_ORDER_CODE	173.174	w/r	Ordering Code

HVT's ALOCATION MAP for the LD301

Index	Parameter Name	HART Variable Name	HART Command		Description
11.1	U8B_ARRAY_1[1]	FLANGE_TYPE	128.129	w/r	Flange Type
11.2	U8B_ARRAY_1[2]	FLANGE_MATERIAL	128.129	w/r	Flange Material
11.3	U8B_ARRAY_1[3]	O_RING	128.129	w/r	O_Ring
11.4	U8B_ARRAY_1[4]	METER_INSTALLATION	128.129	w/r	Meter Installation
11.5	U8B_ARRAY_1[5]	DRAIN_VENT_MATERIAL	128.129	w/r	Drain Vent Material
11.6	U8B_ARRAY_1[6]	REMOTE_SEAL_TYPE	128.129	w/r	Remote Seal Type
11.7	U8B_ARRAY_1[7]	REMOTE_SEAL_FILL_FLUID	128.129	w/r	Remote Seal Fill Fluid
11.8	U8B_ARRAY_1[8]	REMOTE_SEAL_ISO_DIA_MATERIAL	128.129	w/r	Remote Seal Isolating Diaphragm Material
11.9	U8B_ARRAY_1[9]	NUMBER_REMOTE_SEAL	128.129	w/r	Number of Remote Seals
11.10	U8B_ARRAY_1[10]	SENSOR_FILL_FLUID	128.129	w/r	Sensor Fill Fluid
11.11	U8B_ARRAY_1[11]	SENSOR_ISO_DIA_MATERIAL	128.129	w/r	Sensor Isolating Diaphragm Material
11.12	U8B_ARRAY_1[12]	SENSOR_TYPE	128	r	Sensor Type
11.13	U8B_ARRAY_1[13]	SENSOR_RANGE	128	r	Sensor Range
11.14	U8B_ARRAY_1[14]	SENSOR_RANGE_UNIT	128	r	Sensor Range Unit
11.15	U8B_ARRAY_1[15]	SPECIAL_TRANSFER_FUNCTION	128	r	Special Transfer Function
11.16	U8B_ARRAY_1[16]	LOCAL_KEYS_MODE	128.132	r	Local Keys Mode
11.19	U8B_ARRAY_1[19]	TABLE_NUMBER_POINTS	133.135	w/r	Number of points in the table
12.1	U8B_ARRAY_2[1]	PV_ALARM_SELECT	203	w/r	Primary Variable Alarm Selection
12.2	U8B_ARRAY_2[2]	USER_UNIT_MODE	178.180	w/r	User Unit Mode
12.3	U8B_ARRAY_2[3]	USER_UNIT	176.177	w/r	User Unit
12.4	U8B_ARRAY_2[4]	JUMPER_SWITCH	170	r	Jumper Switch
12.5	U8B_ARRAY_2[5]	LOCAL_ADJUST	170	r	Local Adjust
12.6	U8B_ARRAY_2[6]	LOCAL_ADJUST_SOFTWARE	170	r	Local Adjust Software
12.7	U8B_ARRAY_2[7]	LOCAL_ADJUST_JUMP	170	r	Local Adjust Jump
12.8	U8B_ARRAY_2[8]	COMMUNIC_PROTECT_MODE	169	w/r	Communication Write Protection mode
12.9	U8B_ARRAY_2[9]	TOTALIZATION_CHANGE_COUNT	166	r	Number of times the Totalization was done
12.10	U8B_ARRAY_2[10]	PASSWOR_LEVEL_CHANGE_COUNT	166	r	Number of times the Password Level was done
12.11	U8B_ARRAY_2[11]	MULTIDROP_CHANGE_COUNT	166	r	Number of times the Multidrop was done
12.12	U8B_ARRAY_2[12]	WRITE_PROTECTION_CHANGE_COUNT	166	r	Number of times the Write Protection was done
12.14	U8B_ARRAY_2[14]	LOCAL_ADJUST_CHANGE_COUNT	166	r	Number of times the Local Adjust was done
12.16	U8B_ARRAY_2[16]	CHARACT_MODE	160	r	Indicates if the characterization curve is enabled or disable
12.17	U8B_ARRAY_2[17]	NUMBER_CHARACT_POINT	160.161	w/r	Number of Characterization Points
12.19	U8B_ARRAY_2[19]	CHAR_AND_DISPLAY_MODE	163	w	Characterization Trim mode and Display
13.1	U8B_ARRAY_3[1]	FIRST_DISPLAY_CODE	164.165	w/r	First Display
13.2	U8B_ARRAY_3[2]	SECOND_DISPLAY_CODE	164.165	w/r	Second Display
13.3	U8B_ARRAY_3[3]	ZERO_SPAN_CHANGE_COUNT	166	r	Number of times the Zero Span was done
13.4	U8B_ARRAY_3[4]	FUNCTION_CHANGE_COUNT	166	r	Number of times the Function was done
13.5	U8B_ARRAY_3[5]	TRIM_4MA_CHANGE_COUNT	166	r	Number of times the 4 mA trim was done

HVT's ALOCATION MAP for the LD301

Index	Parameter Name	HART Variable Name	HART Command		Description
13.6	U8B_ARRAY_3[6]	TRIM_20MA_CHANGE_COUNT	166	r	Number of times the 20 mA trim was done
13.7	U8B_ARRAY_3[7]	LOWER_TRIM_CHANGE_COUNT	166	r	Number of times the Lower Trim was done
13.8	U8B_ARRAY_3[8]	UPPER_TRIM_CHANGE_COUNT	166	r	Number of times the Upper Trim was done
13.9	U8B_ARRAY_3[9]	RESERVED	166	r	Reserved
13.10	U8B_ARRAY_3[10]	MODE_CHANGE_COUNT	166	r	Number of times the Mode was done
13.11	U8B_ARRAY_3[11]	CHARAC_TRIM_CHANGE_COUNT	166	r	Number of times the Characterization Trim was done
13.12	U8B_ARRAY_3[12]	EXE_PV_UPPER_RANGE_VALUE	36	w	Set Primary Variable Upper Range Value
13.13	U8B_ARRAY_3[13]	EXE_PV_LOWER_RANGE_VALUE	37	w	Set Primary Variable Lower Range Value
13.14	U8B_ARRAY_3[14]	EXE_EEPROM_CONTROL	39	w	Eeprom Control
13.15	U8B_ARRAY_3[15]	EXE_PV_ZERO	43	w	Set Primary Variable Zero
13.16	U8B_ARRAY_3[16]	READ_FROM_SENSOR	153	w	Read from Sensor
13.17	U8B_ARRAY_3[17]	FULL_WRITE_ON_SENSOR	153	w	Full Write On Sensor
13.18	U8B_ARRAY_3[18]	WRITE_ON_SENSOR	153	w	Write On Sensor
13.19	U8B_ARRAY_3[19]	FULL_READ_FROM_SENSOR	153	w	Full Read From Sensor
15.11	U8B_ARRAY_5[11]	TRIM_UNIT	130.131	r	Trim Unit
15.12	U8B_ARRAY_5[12]	MEASURED_POINT_UNIT	162	r	Measured Point Unit
15.13	U8B_ARRAY_5[13]	LOAD_RESTORE_TRIM	163	w	Load/Restore Trim
16.1	FLOAT_ARRAY_1[1]	PV_CURR_LEVEL_ZERO	45	w	Set value of the Trim Primary Variable Current Dac Zero
16.2	FLOAT_ARRAY_1[2]	UPPER_SENSOR	128	r	Upper Sensor Value
16.3	FLOAT_ARRAY_1[3]	LOWER_SENSOR	128	r	Lower Sensor Value
16.4	FLOAT_ARRAY_1[4]	UPPER_TRIM_POINT	130	w	Write Upper Sensor Trim Point
16.5	FLOAT_ARRAY_1[5]	LOWER_TRIM_POINT	131	w	Write Lower Sensor Trim Point
16.6	FLOAT_ARRAY_1[6]	MEASURED_POINT_1	160.162	w/r	Measured characterization Trim Curve Point 1
16.7	FLOAT_ARRAY_1[7]	MEASURED_POINT_2	160.162	w/r	Measured characterization Trim Curve Point 2
16.8	FLOAT_ARRAY_1[8]	MEASURED_POINT_3	160.162	w/r	Measured characterization Trim Curve Point 3
16.9	FLOAT_ARRAY_1[9]	MEASURED_POINT_4	160.162	w/r	Measured characterization Trim Curve Point 4
16.10	FLOAT_ARRAY_1[10]	MEASURED_POINT_5	160.162	w/r	Measured characterization Trim Curve Point 5
16.11	FLOAT_ARRAY_1[11]	USER_UNIT_LOWER	178.179	w/r	User Unit Upper Value
16.12	FLOAT_ARRAY_1[12]	USER_UNIT_UPPER	178.179	w/r	User Unit Lower Value
16.15	FLOAT_ARRAY_1[15]	PV_CURR_LEVEL_GAIN	46	w	Set value of the Trim Primary Variable Current Dac Gain
16.16	FLOAT_ARRAY_1[16]	X1	133.134	w/r	Table Coord X1
16.17	FLOAT_ARRAY_1[17]	X2	133.134	w/r	Table Coord X2
16.18	FLOAT_ARRAY_1[18]	X3	133.134	w/r	Table Coord X3
16.19	FLOAT_ARRAY_1[19]	X4	133.134	w/r	Table Coord X4
16.20	FLOAT_ARRAY_1[20]	X5	133.134	w/r	Table Coord X5
17.1	FLOAT_ARRAY_2[1]	X6	133.134	w/r	Table Coord X6
17.2	FLOAT_ARRAY_2[2]	X7	133.134	w/r	Table Coord X7
17.3	FLOAT_ARRAY_2[3]	X8	133.134	w/r	Table Coord X8
17.4	FLOAT_ARRAY_2[4]	X9	133.134	w/r	Table Coord X9
17.5	FLOAT_ARRAY_2[5]	X10	133.134	w/r	Table Coord X10

HVT's ALOCATION MAP for the LD301					
Index	**Parameter Name**	**HART Variable Name**	**HART Command**		**Description**
17.6	FLOAT_ARRAY_2[6]	X11	133.134	w/r	Table Coord X11
17.7	FLOAT_ARRAY_2[7]	X12	133.134	w/r	Table Coord X12
17.8	FLOAT_ARRAY_2[8]	X13	133.134	w/r	Table Coord X13
17.9	FLOAT_ARRAY_2[9]	X14	133.134	w/r	Table Coord X14
17.10	FLOAT_ARRAY_2[10]	X15	133.134	w/r	Table Coord X15
17.11	FLOAT_ARRAY_2[11]	X16	133.134	w/r	Table Coord X16
17.12	FLOAT_ARRAY_2[12]	Y1	133.134	w/r	Table Coord Y1
17.13	FLOAT_ARRAY_2[13]	Y2	133.134	w/r	Table Coord Y2
17.14	FLOAT_ARRAY_2[14]	Y3	133.134	w/r	Table Coord Y3
17.15	FLOAT_ARRAY_2[15]	Y4	133.134	w/r	Table Coord Y4
17.16	FLOAT_ARRAY_2[16]	Y5	133.134	w/r	Table Coord Y5
17.17	FLOAT_ARRAY_2[17]	Y6	133.134	w/r	Table Coord Y6
17.18	FLOAT_ARRAY_2[18]	Y7	133.134	w/r	Table Coord Y7
17.19	FLOAT_ARRAY_2[19]	Y8	133.134	w/r	Table Coord Y8
17.20	FLOAT_ARRAY_2[20]	Y9	133.134	w/r	Table Coord Y9
18.1	FLOAT_ARRAY_3[1]	Y10	133.134	w/r	Table Coord Y10
18.2	FLOAT_ARRAY_3[2]	Y11	133.134	w/r	Table Coord Y11
18.3	FLOAT_ARRAY_3[3]	Y12	133.134	w/r	Table Coord Y12
18.4	FLOAT_ARRAY_3[4]	Y13	133.134	w/r	Table Coord Y13
18.5	FLOAT_ARRAY_3[5]	Y14	133.134	w/r	Table Coord Y14
18.6	FLOAT_ARRAY_3[6]	Y15	133.134	w/r	Table Coord Y15
18.7	FLOAT_ARRAY_3[7]	Y16	133.134	w/r	Table Coord X16
18.8	FLOAT_ARRAY_3[8]	ACTUAL_POINT_1	160.162	w/r	Actual characterization Trim Curve Point 1
18.9	FLOAT_ARRAY_3[9]	ACTUAL_POINT_2	160.162	w/r	Actual characterization Trim Curve Point 2
18.10	FLOAT_ARRAY_3[10]	ACTUAL_POINT_3	160.162	w/r	Actual characterization Trim Curve Point 3
18.11	FLOAT_ARRAY_3[11]	ACTUAL_POINT_4	160.162	w/r	Actual characterization Trim Curve Point 4
18.12	FLOAT_ARRAY_3[12]	ACTUAL_POINT_5	160.162	w/r	Actual characterization Trim Curve Point 5
24	String_01	USER_UNIT_STRING	176.177	w/r	User Unit String
34	String_11	ORDERING_CODE	173.174	w/r	Ordering Code

Appendix D

Codes for Smar HART variables and configuration of the HIRT block for the AssetView

The variables in these tables can be accessed by the HART command 33 through the correct configuration of the XX_CODE parameters of the HIRT block. Also remember to configure the correct VIEW in the VIEW_SELECTION parameter for efficient reading of the desired variables.

FY301 Intelligent Valve Positioner

Available Variables

Index	Variable
0	Input Current in milliamperes
1	Process Variable in percentage
2	Current Setpoint in percentage
3	PID MV in percentage
4	PID ERRO in percentage
5	Desired Pos in percentage
6	Setpoint in percentage
7	PID Integral in percentage
8	Hall
9	Temperature in °C
10	Piezo Voltage
11	Temperature in °F
12	Travel
13	Strokes
14	Reversals
15	Lowest Temperature
16	Highest Temperature
17	None
18	None
19	None
20	Opening Time
21	Closing Time
22	Setup Watchdog
23	Out Press 1
24	Out Press 2
25	In Press
251	None (Only employed to cancel the secondary variable) - preferred

Configuration of the HIRT block for AssetView:

HIRT Parameter	Value	Description
VIEW_SELECTION	VIEW_09	Default View
A1_CODE	23	PressOut1
A2_CODE	24	PressOut2
A3_CODE	25	PressIn
A4_CODE	1	PV (Valve posistion)
B1_CODE	21	Closing Time
B2_CODE	20	Opening Time
B3_CODE	10	Piezo
B4_CODE	8	Hall
C1_CODE	13	Strokes
C2_CODE	14	Reverse
C3_CODE	12	Mileage
C4_CODE	17	Setup Progress

LD301 Intelligent Pressure Transmitter

Variables list of the HART command #33 for the LD301

Index	Variable
0	OUTPUT IN MILLIAMPERES
1	OUTPUT IN PERCENT
2	PRESSURE (PRIMARY VARIABLE)
3	PROCESS VARIABLE PERCENT
4	PROCESS VARIABLE
5	TEMPERATURE (SECONDARY VARIABLE)
6	SETPOINT PERCENT
7	SETPOINT
8	ERROR
9	TOTAL
251	None (Only employed to cancel the secondary variable) – preferred
255	None (Only employed to cancel the secondary variable)

Configuration of the HIRT block for AssetView:

Available Soon

TT301 Intelligent Temperature Transmitter

Variables list of the HART command #33 for the TT301

Index	Variable
0	Output in miliamperes - Out
1	Output in percent - Out%
2	Temperature - (PV)
3	Environment temperature - Temp
4	Process variable percent- PV%
5	Setpoint percent - SP%
6	Setpoint - SP
7	Setpoint time - SPTIME
8	Error - ER%
9	Pid_KP -KP
10	Pid_TR - TR
11	Pid_TD - TD
12	Damping - Damp
13	Manual register - MV
14-24	Reserved
25	Input variable (used for trim)
26-254	Reserved
255	None (Only for secondary display code)

Configuration of the HIRT block for AssetView:

Available Soon

DT301 Intelligent Density Transmitter

Variables list of the HART command #33 for the DT301

Index	Variable
0	Output in miliamperes - Out
1	Output in percent - Out%
2	Temperature - (PV)
3	Environment temperature - Temp
4	Process variable percent- PV%
5	Setpoint percent - SP%
6	Setpoint - SP
7	Setpoint time - SPTIME
8	Error - ER%
9	Pid_KP -KP
10	Pid_TR - TR
11	Pid_TD - TD

Configuration of the HIRT block for AssetView:

Available Soon

TP301 Intelligent Pressure Transmitter

Variables list of the HART command #33 for the TP301

Index	Variable
0	Posição em porcentagem
1	Saída em Miliamperes
2	Temperatura em °C
3	Temperatura em F (Fahrenheit)
4	Hall
5	unidade de posição P. V. (E.U.)
6	% of Hall

Configuration of the HIRT block for AssetView:

Available Soon