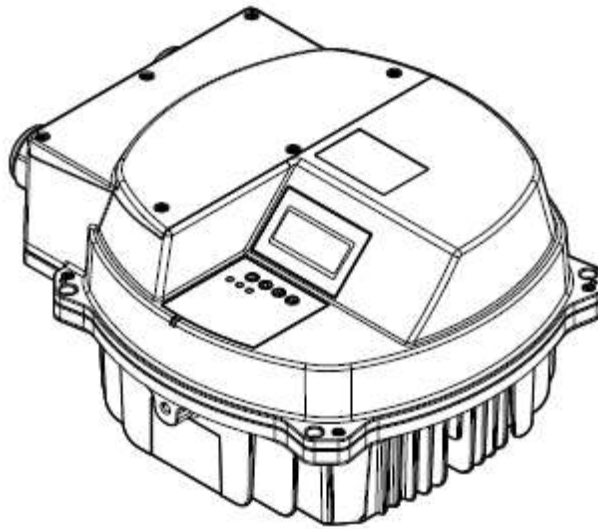


**HYDROVAR®**



# HVL 2.015 – 4.220

## Modbus Protocol & Parameters

HVL Software Version: 2.10

HVL Software Version: 2.20

# 1 Index

<b>1</b>	<b>Index .....</b>	<b>2</b>
<b>2</b>	<b>A Few Facts about the Modbus Protocol on HYDROVAR HVL.....</b>	<b>3</b>
2.1	Communication.....	3
2.2	Broadcasting.....	3
2.3	Data Protection .....	3
2.4	Transmission Mode.....	4
2.5	Supported Function Codes .....	5
2.6	Not supported Modbus Function Codes.....	8
<b>3</b>	<b>Connections and data handling .....</b>	<b>9</b>
3.1	Single HVL connected to an External Device .....	9
3.1.1	HVL data handling when connected to an External Device .....	11
3.2	HVLs in Multi-Pump application connected to an External Device.....	12
3.2.1	HVL data handling when connected in Multi-Pump and to an External Device .....	13
<b>4</b>	<b>HYDROVAR HVL Settings .....</b>	<b>14</b>
<b>5</b>	<b>HYDROVAR HVL Registers list - SW version 2.10, and SW version 2.20 .....</b>	<b>16</b>



**Read and follow the operating instructions and safety instructions carefully before starting operations! All modifications must be done by qualified technicians!**



## 2 A Few Facts about the Modbus Protocol on HYDROVAR HVL



**NOTE:** *The Modbus Protocol is an international standardized Bus Protocol! The general information within this Manual is just a brief overview, for detailed information please use the Modbus Protocol reference guide, or any other source of information (e.g. Modbus org. “MODBUS application protocol specification” available on the Internet).*

This protocol defines a message structure that controllers will recognize and use, regardless of the type of networks over which they communicate. It describes the process a controller uses to request access to another device, how it will respond to requests from the other devices, and how errors will be detected and reported. It establishes a common format for the layout and contents of message fields.

During communications on a Modbus network, the protocol determines how each controller will know its device address, recognize a message addressed to it, determine the kind of action to be taken, and extract any data or other information contained in the message.

### 2.1 Communication

HYDROVAR HVL uses the RS485 serial interface that defines connect pinouts, cabling, signal levels, transmission baud rates and parity checking.

Controllers communicate using a master-slave technique, in which only the master can start a transfer or polling. The other devices (Slaves) respond by supplying the requested data to the master, or by taking the action requested in the query.

### 2.2 Broadcasting

On HYDROVAR HVL broadcast function is not supported.

### 2.3 Data Protection

Standard Modbus serial networks use two kinds of error checking:

- Parity checking (even or odd) can be optionally applied to each character.
- Frame checking (LRC or CRC) is applied to the entire message.

Both the character check and message frame check are generated in the master device and applied to the message contents before transmission. The slave device checks each character and the entire message frame during receipt.

Detailed information can be found in the Modbus Protocol Reference Guide!

## 2.4 *Transmission Mode*

When using the Modbus Protocol the selection to choose is between two transmission Modes: ASCII or RTU.

The different modes determine how information will be packed into the message fields and decoded.

In addition to the desired transmission mode, the serial port communication parameters (baud rate, parity mode...) shall be selected.

! The mode and serial parameters must be the same for all devices on the Modbus network!

The following modes can be selected and are supported by HYDROVAR HVL:

RTU / 8, E, 1	1 start bit, 8 data bits, 1 stop bit, Even parity
RTU / 8, O, 1	1 start bit, 8 data bits, 1 stop bit, Odd parity
RTU / 8, N, 2	1 start bit, 8 data bits, 2 stop bits, No parity
RTU / 8, N, 1	1 start bit, 8 data bits, 1 stop bit, No parity
ASCII / 7, N, 2	1 start bit, 7 data bits, 2 stop bits, No parity
ASCII / 7, E, 1	1 start bit, 7 data bits, 1 stop bit, Even parity
ASCII / 7, O, 1	1 start bit, 7 data bits, 1 stop bit, Odd parity

## 2.5 Supported Function Codes

- **0x03 Read Holding Registers – READ COMMAND**

Read the binary contents of holding registers in the slave!

Note: The Modbus Registers are addressed starting at zero!  
E.g. Holding Register indexed 0x33 has to be addressed as 0x32

Example: Read the Actual Value

### QUERY

Slave Address	0x01	Could be set on HVL via Parameter ADDRESS [P1205]
Function	0x03	Read Holding Register
Starting Address High	0x00	
Starting Address Low	0x32	Modbus Index 0x33 – Actual value has to be addressed
No. of Points High	0x00	
No. of Points Low	0x01	Read one holding register is supported
Error Check CRC-High	0x25	
Error Check CRC-Low	0xC5	Generated CRC-Checksum

### RESPONSE

Slave Address:	0x01	
Function	0x03	
Byte Count	0x02	
Data High	0x02	
Data Low	0x08	=> 208 HEX = 520 DEC => Actual Value = 5.20 bar
Error Check CRC-High	0xB8	
Error Check CRC-Low	0xE2	Generated CRC-Checksum

- **0x06 Write Single Register – WRITE COMMAND**

Writes a value into a single holding register! Broadcast is not supported!

Note: The Modbus Registers are addressed starting at zero!  
E.g. Holding Register indexed 0xE9 has to be addressed as 0xE8

Example: Set the Required Value 1 to 3.50 bar

#### QUERY

Slave Address	0x01	Could be set on HVL via Parameter ADDRESS [P1205]
Function	0x06	Write Single Register
Register Address High	0x00	
Register Address Low	0xE8	Modbus Index 0xE9 – Req. Value 1 has to be addressed
Preset Data High	0x01	
Preset Data Low	0x5E	=> 15E HEX = 350 DEC => sets the Required Value 1 to 3.50 bar
Error Check CRC-High	0x89	
Error Check CRC-Low	0x96	Generated CRC-Checksum

#### RESPONSE

Slave Address:	0x01	
Function	0x06	
Register Address High	0x00	
Register Address Low	0xE8	
Preset Data High	0x01	
Preset Data Low	0x5E	=> Required Value 1 is set to 3.50 bar
Error Check CRC-High	0x89	
Error Check CRC-Low	0x96	Generated CRC-Checksum

- **0x10 Write Multiple Registers – WRITE COMMAND**

Writes values into a block of contiguous registers! Broadcast is not supported!

Note: The Modbus Registers are addressed starting at zero!  
E.g. Holding Register indexed 0x98 has to be addressed as 0x97

Example: Set Ramp 1 and Ramp 2 to 25sec, Ramp 3 and Ramp 4 to 100sec

#### QUERY

Slave Address	0x01	Could be set on HVL via Parameter ADDRESS [P1205]
Function	0x10	Write Multiple Registers
Starting Address High	0x00	
Starting Address Low	0x97	Modbus Index 0x98 – Ramp 1 has to be addressed
Quantity of Registers Hi	0x00	
Quantity of Registers Lo	0x04	In total 4 registers (Ramp 1 to Ramp 4) to be written
Byte Count	0x08	2 * Quantity of Registers
Reg Value High	0x00	
Reg Value Low	0x19	=> 19 HEX = 25 DEC => sets Ramp 1 to 25sec
Reg Value High	0x00	
Reg Value Low	0x19	=> 19 HEX = 25 DEC => sets Ramp 2 to 25sec
Reg Value High	0x00	
Reg Value Low	0x64	=> 64 HEX = 100 DEC => sets Ramp 3 to 100sec
Reg Value High	0x00	
Reg Value Low	0x64	=> 64 HEX = 100 DEC => sets Ramp 4 to 100sec
Error Check CRC-High	0x55	
Error Check CRC-Low	0x07	Generated CRC-Checksum

#### RESPONSE

Slave Address:	0x01	
Function	0x10	
Starting Address High	0x00	
Starting Address Low	0x97	
Quantity of Registers Hi	0x00	
Quantity of Registers Lo	0x04	In total 4 registers (Ramp 1 to Ramp 4) written
Error Check CRC-High	0x70	
Error Check CRC-Low	0x26	Generated CRC-Checksum

## **2.6 Not supported Modbus Function Codes**

HYDROVAR HVL only supports Function Codes described in par. 2.5; for the sake of clarity, other Modbus Function Codes not used/supported by HVL are:

- 0x01 - Read Coils
- 0x02 - Read Discrete Inputs
- 0x04 - Read Input Registers
- 0x05 - Write Single Coil
- 0x07 - Read Exception Status
- 0x08 - Diagnostics
- 0x0B - Get Comm Event Counter
- 0x0C - Get Comm Event Log
- 0x0F - Write Multiple Coils
- 0x11 - Report Slave ID
- 0x14 - Read File Record
- 0x15 - Write File Record
- 0x16 - Mask Write Register
- 0x17 - Read/Write Multiple registers
- 0x18 - Read FIFO Queue
- 0x2B - Encapsulated Interface Transport



## 3 Connections and data handling

### 3.1 Single HVL connected to an External Device



**NOTE:** For detailed information regarding installation, wiring and configuration of the HYDROVAR HVL, please read and follow the relative Installation, Operation and Maintenance Manual



**WARNING:** This product is intended to be operated by qualified personnel only.

Unscrew the dedicated 6 screws and remove the plastic cover of HYDROVAR HVL, in order to proceed wiring the control terminals; for reference, the wiring harness scheme is reported on the backside of the plastic cover too.

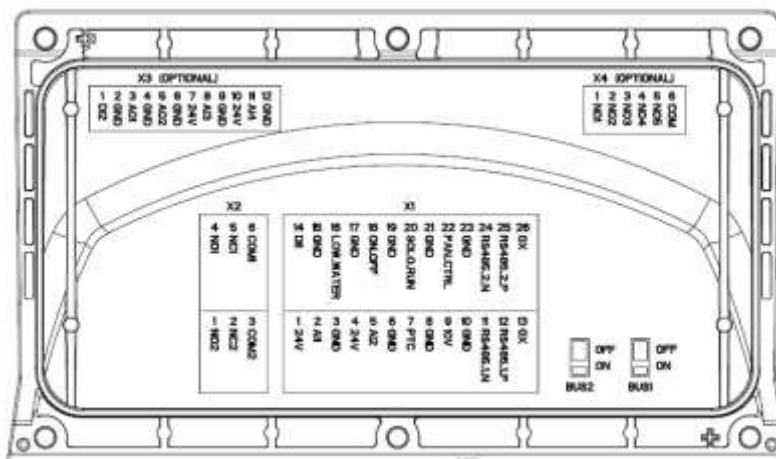
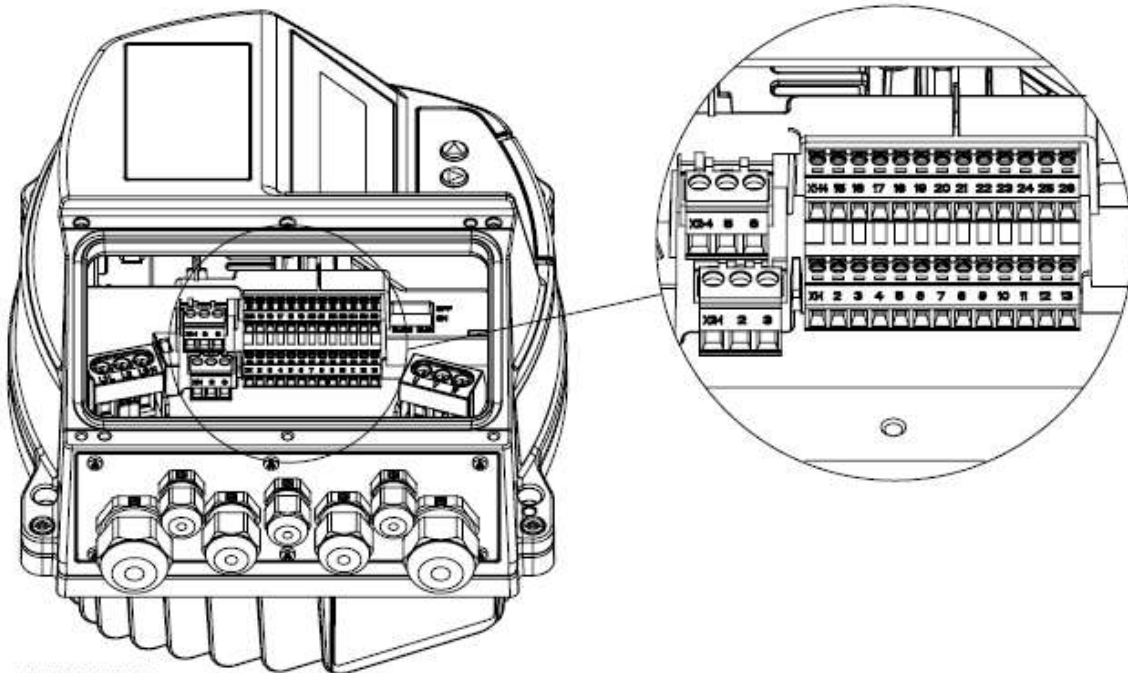
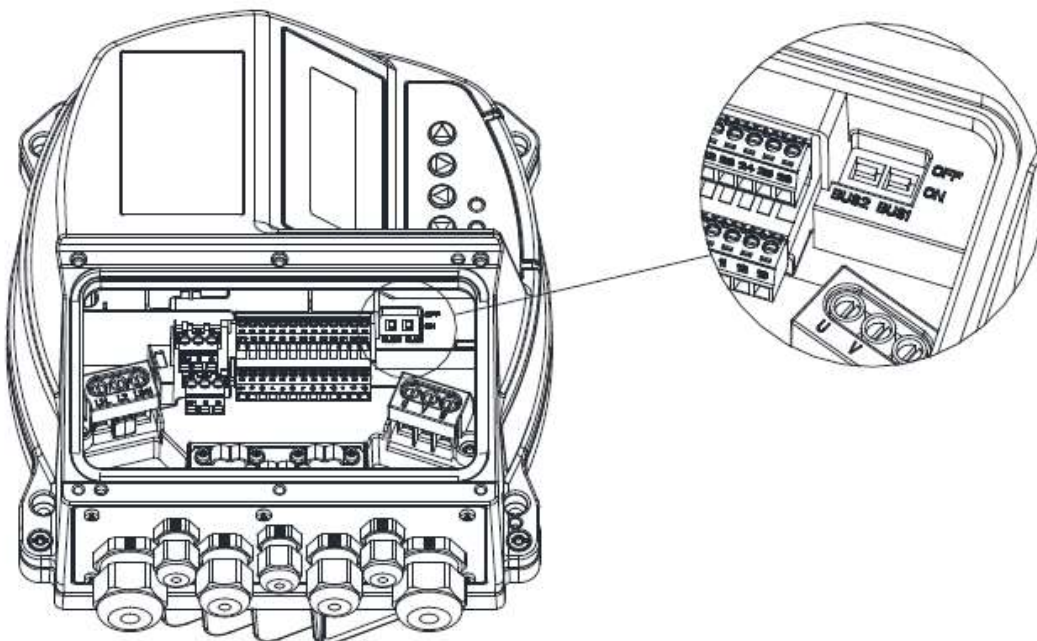


Figure 1



**Figure 2**

Terminals X1/24, X1/25 and X1/26 are used for the communication with an external-control-device (e.g. PLC, BMS or a PC too); a dedicated termination resistor switch (BUS2, see image below) is made available to add a parallel terminator resistor to this RS485 port: if the resistor is needed put BUS2 switch on ON position.



**Figure 3**

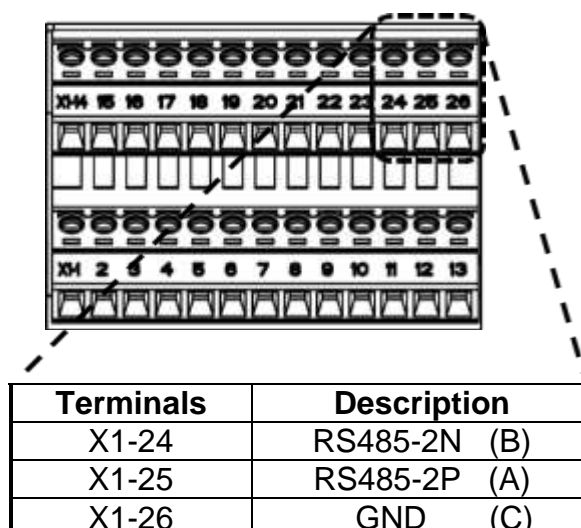


Figure 4



**NOTE:** Do not connect the ground of the control card to other voltage potentials. All ground terminals and ground of the RS485 connection are connected internally.

For connecting HYDROVAR HVL to a PC, is it possible to simply refer to the provided “USB/RS485 Cable Kit M&C” (cod.109395920), which integrates a USB/RS485 converter, and then to the relative wiring table below

USB/RS485 Cable Kit M&C (cod. 109395920)	Terminals	Description
B = YL (Yellow)	X1-24	RS485 port 2: RS485-2N
A = OR (Orange)	X1-25	RS485 port 2: RS485-2P
GND = BK (Black)	X1-26	GND, electronic ground

### 3.1.1 HVL data handling when connected to an External Device

When the Modbus communication is active between the HVL and an External Device, HVL “locks” the possibility to change most of the system parameters from the local push buttons on the inverter. This is to prevent conflicts between the values (read and written) by the (remote) External Device, and the parameters values that could be over-written/changed by the local operator. The HVL display is showing a “REMOTE” indication to signal the active locking on parameter edit during an External Device Modbus access.

The parameters free from REMOTE change lock are the same related to the function of parameter P115 “LOCK FUNCT.”, e.g. Stop/Start manual command on HOME display, and P61 “PASSWORD”.

To un-lock the parameter editing by the local push buttons on the HYDROVAR HVL, the Modbus communication with the device must be stopped or interrupted.

### 3.2 HVLs in Multi-Pump application connected to an External Device

Terminals X1/11, X1/12 and X1/13 are used for the communication among up to 8 HYDROVAR HVL in a Multi-Pump application; a dedicated termination resistor switch (BUS1, see image below) is made available to add a parallel terminator resistor to this RS485 port: if the resistor is needed put BUS1 switch to ON position, using as a reference the Modbus org. “MODBUS over serial line specification and implementation guide” that can be found on the Internet.

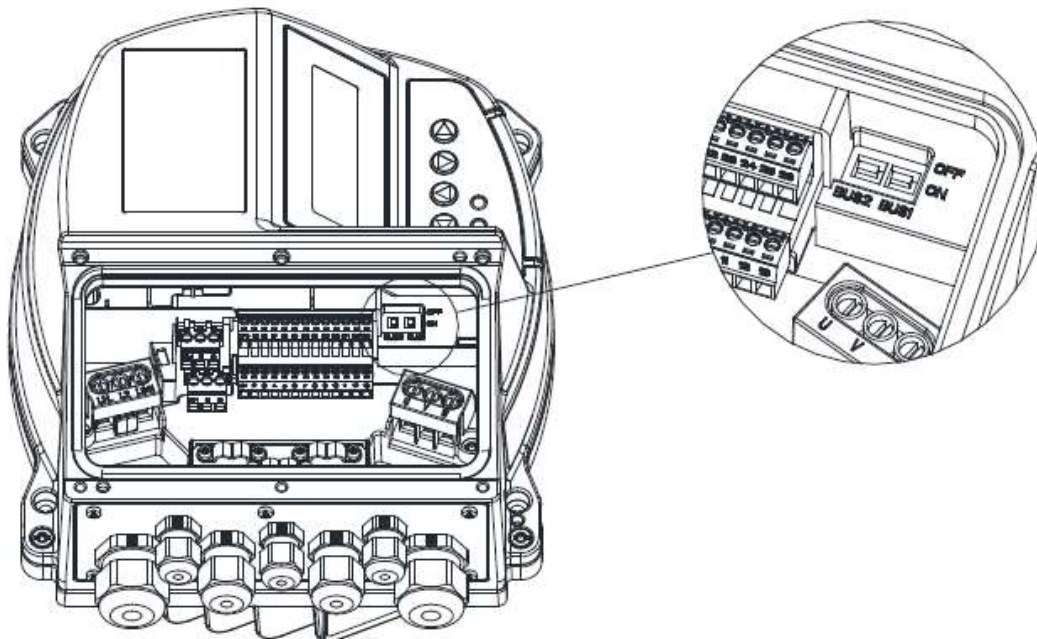


Figure 5

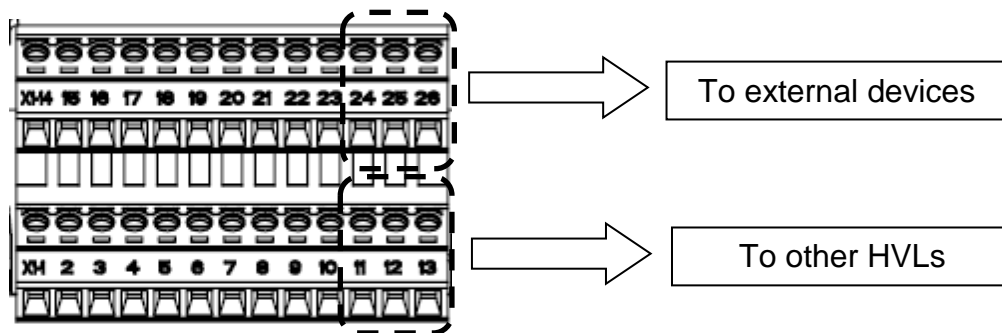


Figure 6

In a Multi-Pump application communicating with an external device (e.g. PLC or BMS), each HYDROVAR HVL shall be connected to the external device by wiring the terminals X1/24, X1/25 and X1/26, as described in par. 3.1

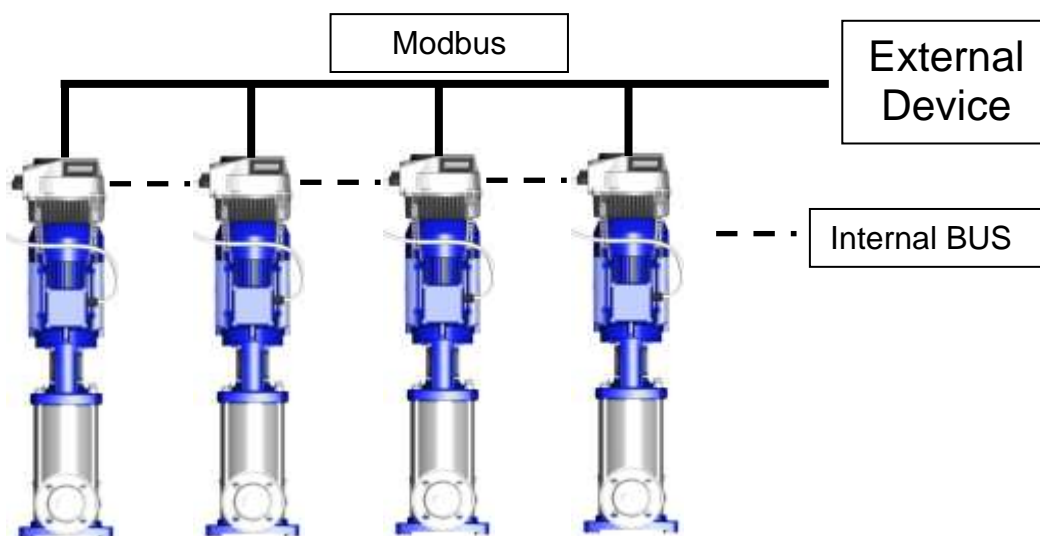


Figure 7

### 3.2.1 HVL data handling when connected in Multi-Pump and to an External Device

Special care should be taken when an External Device (by mean of Modbus protocol), requires to read and write HVL parameters, being the drive connected into a Multi-Pump system too.

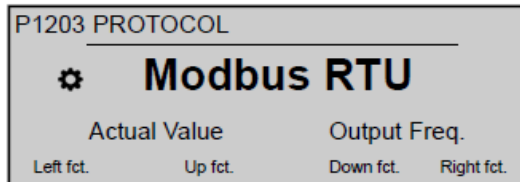
In particular

- In a Multi-Pump application, to “Read Holding Registers” requests on the Modbus, each HVL returns to the External Device exclusively its own parameters, not parameters of other drives (HVL) connected in the booster set
- In a Multi-Pump application, requests of “Write Single Registers” or “Write Multiple Registers” on the Modbus must be sent, by the external device, to all HVLs connected, even if the parameters to be written are “Global” (for the booster set)

As previously detailed at par. 3.1.1, if the Modbus communication is active between the HYDROVAR HVL linked within a booster set and connected to an External Device, then it will be “locked” the possibility to change most system parameters from the local push buttons on the Hydrovar, to prevent conflicts between the values read/written by the remote External Device. The REMOTE lock for parameters edit can be disabled stopping the Modbus communication.

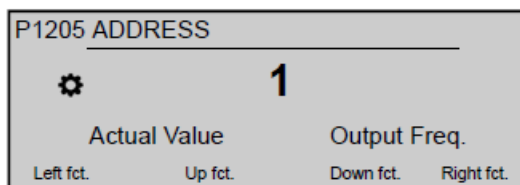
## 4 HYDROVAR HVL Settings

The following 4 parameters, on HYDROVAR HVL, have to be set to guarantee correct Modbus communication.

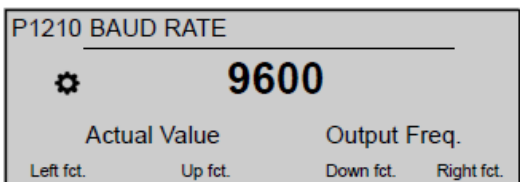


Sets the desired communication protocol; possible settings are:

- Disabled
- Modbus RTU
- Modbus ASCII
- BACNet MS/TP




Sets the desired Modbus address (possible setting 1 - 247): each drive must be allocated its own address!



Set the BAUD RATE for the communication; possible settings are:

- 1200
- 2400
- 4800
- 9600
- 14400
- 19200
- 38400
- 57600
- 76800
- 115200

P1215 FORMAT			
 <b>8, N, 1</b>			
Actual Value		Output Freq.	
Left fct.	Up fct.	Down fct.	Right fct.

Set the data FORMAT for the communication port, depending on the value of P1203 PROTOCOL; possible settings are:

- 8, E, 1
- 8, O, 1
- 8, N, 2
- 8, N, 1
- 7, E, 1
- 7, O, 1
- 7, N, 2

## 5 HYDROVAR HVL Registers list - SW version 2.10, and SW version 2.20

Modbus address		Function codes	Menu Index	Description	Datum type	Values range		Enum Description
(HEX)	(DEC)					Minimum	Maximum	
0x0031	49	0x03, 0x06, 0x10		"STOP/START"	U08	0	1	0 = Off 1 = On
0x0032	50	0x03		"ACTUAL VALUE"	S16	0	P420 Note 1	
0x0033	51	0x03	P46	"OUTPUT FREQ."	S16	0	700	
0x0037	55	0x03	P03	"EFF.REQ.VAL."	U16	0	P420 Note 1	
0x0038	56	0x03, 0x06, 0x10	P04	"START VALUE"	U08	0	100	
0x0039	57	0x03, 0x06, 0x10	P05	"LANGUAGE"	U08	0	27	0 = English 1 = Italian 2 = French 3 = German 4 = Spanish 5 = Portuguese 6 = Dutch 7 = Danish 8 = Norwegian 9 = Swedish 10 = Finnish 11 = Icelandic 12 = Estonian 13 = Latvian 14 = Lithuanian 15 = Polish 16 = Czech 17 = Slovak 18 = Hungarian 19 = Romanian 20 = Bulgarian 21 = Slovene 22 = Croatian 23 = Serbian 24 = Greek 25 = Turkish 26 = Russian 27 = Ukrainian
0x003A	58	0x03, 0x10	P06	"DATE"	U32	0x00010100 Note 3	0x630C1F07 Note 3	
0x0040	64	0x03, 0x10	P07	"TIME"	U32	0x00000000 Note 4	0x00173B3B Note 4	
0x0046	70	0x03, 0x06, 0x10	P08	"AUTO-START"	U08	0	1	0 = Off 1 = On



Modbus address		Function codes	Menu Index	Description	Datum type	Values range		Enum Description
(HEX)	(DEC)					Minimum	Maximum	
0x0047	71	0x03	P09	"OPERAT.TIME"	U32	0x00000000 Note 5	0x7FFF3B3B Note 5	
0x0059	89	0x03	P21	"STATUS UNITS"	_B0	0	0xFF	
0x005C	92	0x03, 0x06, 0x10		"Enable Device – Motor relay 1"	U08	0	1	0 = Disable 1 = Enable
0x005D	93	0x03, 0x06, 0x10		"Enable Device – Motor relay 2"	U08	0	1	0 = Disable 1 = Enable
0x005E	94	0x03, 0x06, 0x10		"Enable Device – Motor relay 3"	U08	0	1	0 = Disable 1 = Enable
0x005F	95	0x03, 0x06, 0x10		"Enable Device – Motor relay 4"	U08	0	1	0 = Disable 1 = Enable
0x0060	96	0x03, 0x06, 0x10		"Enable Device – Motor relay 5"	U08	0	1	0 = Disable 1 = Enable
0x0061	97	0x03, 0x06, 0x10	P24	"ENABLE DEVICE"	U08	0	1	0 = Disable 1 = Enable
0x006B	107	0x03	P25	"MOTOR HOURS"	U32	0x00000000 Note 5	0x7FFF3B3B Note 5	
0x006D	109	0x03	P35	"KWH COUNTER"	U32	0	0x0000FFFF	
0x0081	129	0x03	P41	"PROD.DATE"	U32	0	0xFFFFFFFF	
0x0085	133	0x03	P43	"TEMP.INVERTER"	S08	-128	127	
0x0087	135	0x03	P44	"CURR.INVERTER"	U16	0	10000	
0x0088	136	0x03	P45	"VOLT.INVERTER"	U16	0	1000	
0x0089	137	0x03	P47	"VER.INVERTER"	U32	0 Note 6	0xFFFFFFFF Note 6	
0x008B	139	0x03 0x06, 0x10 (Note 9)	P105	"MODE"	U08	0	4	0 = Controller 1 = Cascade Relay 2 = Cascade Serial 3 = Actuator 4 = Cascade Synchron
0x008D	141	0x03, 0x06, 0x10	P115	"LOCK FUNCT."	U08	0	1	0 = Off 1 = On
0x008E	142	0x03, 0x06, 0x10	P120	"DISP.CONTR."	U08	0	100	
0x008F	143	0x03, 0x06, 0x10	P125	"DISP.BRIGHT."	U08	0	100	
0x0090	144	0x03, 0x06, 0x10	P130	"DISP.ROTATION"	U08	0	1	0 = Disable 1 = Enable
0x0092	146	0x03 0x06, 0x10 (Note 2)	P270	"MOTOR POLES"	U08	2	4	
0x0095	149	0x03, 0x06, 0x10	P205	"MAX.UNITS"	U08	1	8	
0x0097	151	0x03, 0x06, 0x10	P215	"RAMP 1"	U16	1	250 - 1000 (*)	
0x0098	152	0x03, 0x06, 0x10	P220	"RAMP 2"	U16	1	250 - 1000 (*)	
0x0099	153	0x03, 0x06, 0x10	P225	"RAMP 3"	U16	1	1000	
0x009A	154	0x03, 0x06, 0x10	P230	"RAMP 4"	U16	1	1000	
0x009B	155	0x03, 0x06, 0x10	P235	"RAMP FMIN A"	U08	10	250	
0x009C	156	0x03, 0x06, 0x10	P240	"RAMP FMIN D"	U08	10	250	
0x009D	157	0x03, 0x06, 0x10	P245	"MAX.FREQ."	U16	300	700	

Modbus address		Function codes	Menu Index	Description	Datum type	Values range		Enum Description
(HEX)	(DEC)					Minimum	Maximum	
0x009E	158	0x03, 0x06, 0x10	P250	"MIN.FREQ."	U16	0	P245 Note 1	
0x009F	159	0x03, 0x06, 0x10	P255	"CONF.FMIN"	U08	0	1	0 = f -> 0 1 = f -> fmin
0x00A0	160	0x03, 0x06, 0x10	P260	"FMIN TIME"	U08	0	100	
0x00A1	161	0x03 0x06, 0x10 (Note 2)	P281	"BOOST"	U08	0	25	
0x00A2	162	0x03 0x06, 0x10 (Note 2)	P282	"KNEE FREQ."	U16	300	P245 Note 1	
0x00A4	164	0x03, 0x06, 0x10	P283	"SEL.SW.FREQ."	U08	Freq.>=P284 (*) Note 1	10 (*)	1 = rand. 5kHz 2 = 2kHz 3 = 3kHz 4 = 4kHz 5 = 5kHz 6 = 6kHz 7 = 8kHz 8 = 10kHz 9 = 12kHz 10 = 16kHz
0x00A5	165	0x03, 0x06, 0x10	P261	"SKIP FRQ.CTR."	U16	0	P245 Note 1	
0x00A6	166	0x03, 0x06, 0x10	P262	"SKIP FRQ.RNG."	U16	0	50	
0x00A7	167	0x03, 0x06, 0x10	P295	"CURR.LIM.FUNCT."	U08	0	1	0 = Disable 1 = Enable
0x00A8	168	0x03, 0x06, 0x10	P296	"CURR.LIMIT SET"	U16	10	300	
0x00A9	169	0x03 0x06, 0x10 (Note 2)	P284	"MIN.SW.FREQ."	U08	2 (*)	10 (*)	2 = 2kHz 3 = 3kHz 4 = 4kHz 5 = 5kHz 6 = 6kHz 7 = 8kHz 8 = 10kHz 9 = 12kHz 10 = 16kHz
0x00AA	170	0x03, 0x06, 0x10	P310	"WINDOW"	U08	0	100	
0x00AB	171	0x03, 0x06, 0x10	P315	"HYSTERESIS"	U08	0	100	
0x00AC	172	0x03, 0x06, 0x10	P320	"REG.MODE"	U08	0	1	0 = Normal 1 = Inverse
0x00AD	173	0x03, 0x06, 0x10	P325	"FRQ.LIFT"	U16	0	P245 Note 1	
0x00AE	174	0x03, 0x06, 0x10	P330	"LIFT AMOUNT"	U16	0	2000	
0x00B3	179	0x03, 0x06, 0x10	P405	"DIMENSION UNIT"	U08	0	12	0 = bar 1 = psi 2 = m3/h 3 = g/min 4 = m/H2O 5 = ft

Modbus address		Function codes	Menu Index	Description	Datum type	Values range		Enum Description
(HEX)	(DEC)					Minimum	Maximum	
								6 = °C 7 = °F 8 = l/sec 9 = l/min 10 = m/sec 11 = ... 12 = %
0x00B4	180	0x03, 0x06, 0x10	P410	"CONF.SENSOR"	U08	0	7 (*)	0 = Sensor1 1 = Sensor2 2 = Auto 3 = Switch Dig1 4 = Switch Dig2 5 = Auto Lower 6 = Auto Higher 7 = Sens1 – Sens2
0x00B5	181	0x03, 0x06, 0x10	P415	"SENSOR TYPE"	U08	0	2	0 = Analog U 0-10V 1 = Analog I 0-20mA 2 = Analog I 4-20mA
0x00B6	182	0x03, 0x06, 0x10	P420	"SENSOR RANGE"	U16	1	10000	
0x00B7	183	0x03, 0x06, 0x10	P425	"SENSOR CURVE"	U08	0	1	0 = Linear 1 = Quadratic
0x00B8	184	0x03, 0x06, 0x10	P430	"SENS.1 CAL.0"	S16	-100	100	
0x00B9	185	0x03, 0x06, 0x10	P435	"SENS.1 CAL.X"	S16	-100	100	
0x00BA	186	0x03, 0x06, 0x10	P440	"SENS.2 CAL.0"	S16	-100	100	
0x00BB	187	0x03, 0x06, 0x10	P445	"SENS.2 CAL.X"	S16	-100	100	
0x00BC	188	0x03, 0x06, 0x10		"RANGE FACTOR"	U08	0	2	
0x00BD	189	0x03, 0x06, 0x10	P505	"ACT.VAL.INC."	U16	0	P420 Note 1	
0x00BE	190	0x03, 0x06, 0x10	P510	"ACT.VAL.DEC."	U16	0	P420 Note 1	
0x00BF	191	0x03, 0x06, 0x10	P515	"ENABLE FRQ."	U16	0	700	
0x00C0	192	0x03, 0x06, 0x10	P520	"ENABLE DLY."	U08	0	100	
0x00C1	193	0x03, 0x06, 0x10	P525	"SWITCH DLY."	U08	0	100	
0x00C2	194	0x03, 0x06, 0x10	P530	"DISABLE FRQ."	U16	0	700	
0x00C3	195	0x03, 0x06, 0x10	P535	"DISABLE DLY."	U08	0	100	
0x00C4	196	0x03, 0x06, 0x10	P540	"DROP FRQ."	U16	0	700	
0x00C5	197	0x03, 0x06, 0x10	P545	"OVERVALUE"	U16	0	P420 Note 1	
0x00C6	198	0x03, 0x06, 0x10	P550	"OVERVAL.DLY."	U08	0	100	
0x00C7	199	0x03, 0x06, 0x10	P555	"SWITCH INTV."	U08	0	250	
0x00C8	200	0x03, 0x06, 0x10	P560	"SYNCHR.LIM."	U16	0	P245 Note 1	
0x00C9	201	0x03, 0x06, 0x10	P565	"SYNCHR.WND."	U16	0	100	
0x00CB	203	0x03 0x06, 0x10 (Note 2)	P280	"SWITCHING CONTROL"	U08	0	1	0 = V/f 1 = HVC

Modbus address		Function codes	Menu Index	Description	Datum type	Values range		Enum Description
(HEX)	(DEC)					Minimum	Maximum	
0x00D1	209	0x03, 0x06, 0x10	P605	"MIN.THRESH."	U16	0	P420 Note 1	
0x00D2	210	0x03, 0x06, 0x10	P610	"DELAY-TIME"	U08	1	100	
0x00D3	211	0x03, 0x06, 0x10	P615	"ERROR RESET"	U08	0	1	0 = Off 1 = On
0x00DB	219	0x03, 0x06, 0x10	P705	"ANALOG OUT.1"	U08	0	1	0 = Actual Value 1 = Output Freq.
0x00DC	220	0x03, 0x06, 0x10	P710	"ANALOG OUT.2"	U08	0	1	0 = Actual Value 1 = Output Freq.
0x00DD	221	0x03, 0x06, 0x10	P715	"CONF.REL.1"	U08	0	5	0 = Power 1 = Running 2 = Errors 3 = Warnings 4= Stand By 5 = Err. Reset
0x00DE	222	0x03, 0x06, 0x10	P720	"CONF.REL.2"	U08	0	5	0 = Power 1 = Running 2 = Errors 3 = Warnings 4= Stand By 5 = Err. Reset
0x00E5	229	0x03, 0x06, 0x10	P805	"C.REQ.VAL.1"	U08	1	4 (*)	1 = Digital 2 = Analog U 0-10V 3 = Analog I 0-20mA 4= Analog I 4-20mA
0x00E6	230	0x03, 0x06, 0x10	P810	"C.REQ.VAL.2"	U08	0	4 (*)	0 = Off 1 = Digital 2 = Analog U 0-10V 3 = Analog I 0-20mA 4= Analog I 4-20mA
0x00E7	231	0x03, 0x06, 0x10	P815	"SW.REQ.VAL."	U08	0	3 (*)	0 = Setpoint 1 1 = Setpoint 2 2 = Switch Dig 1 3 = Switch Dig 2
0x00E8	232	0x03, 0x06, 0x10	P820	"REQ.VAL.1"	U16	0	P420 Note 1	
0x00E9	233	0x03, 0x06, 0x10	P825	"REQ.VAL.2"	U16	0	P420 Note 1	
0x00EA	234	0x03, 0x06, 0x10	P830	"ACTUAT.FRQ.1"	U16	P250 Note 1	P245 Note 1	
0x00EB	235	0x03, 0x06, 0x10	P835	"ACTUAT.FRQ.2"	U16	P250 Note 1	P245 Note 1	
0x00F9	249	0x03, 0x06, 0x10	P1005	"TEST RUN"	U08	0	100	
0x00FA	250	0x03, 0x06, 0x10	P1010	"TEST RUN FRQ."	U16	P250 Note 1	P245 Note 1	
0x00FB	251	0x03, 0x06, 0x10	P1015	"TEST R.BOOST"	U08	0	25	
0x00FC	252	0x03, 0x06, 0x10	P1020	"TEST R.TIME"	U08	0	180	

Modbus address		Function codes	Menu Index	Description	Datum type	Values range		Enum Description
(HEX)	(DEC)					Minimum	Maximum	
0x010D	269	0x03, 0x06, 0x10	P1205	"ADDRESS"	U08	1 (Modbus) 0 (BACNet)	247 (Modbus) 127 (BACNet)	
0x010E	270	0x03, 0x06, 0x10	P1210	"BAUD RATE"	U08	0	9	0 = 1200 1 = 2400 2 = 4800 3 = 9600 4 = 14400 5 = 19200 6 = 38400 7 = 57600 8 = 76800 9 = 115200
0x010F	271	0x03, 0x06, 0x10	P1215	"FORMAT"	U08	0	6 (Modbus ASCII) 3 (any other protocol)	0 = 8, N, 1 1 = 8, N, 2 2 = 8, E, 1 3 = 8, O, 1 4 = 7, N, 2 5 = 7, E, 1 6 = 7, O, 1
0x0110	272	0x03, 0x06, 0x10	P1220	"PUMP ADDR."	U08	1	8	
0x0111	273	0x03, 0x06, 0x10	P1203	"PROTOCOL"	U08	0	3	0 = Disabled 1 = Modbus RTU 2 = Modbus ASCII 3 = BACNet MS/TP
0x0112	274	0x03, 0x10	P1221	"BACNET DEV.ID"	U32	0	0x3FFFFFFF	
0x0117	279	0x03	P265	"MOTOR NOM.POWER"	U08	7 (*)	20 (*)	
0x0118	280	0x03	P266	"MOTOR NOM.VOLT."	U16	208 (*)	460 (*)	
0x0119	281	0x03	P267	"MOTOR NOM.FRQ."	U16	30	70	
0x011A	282	0x03	P268	"MOTOR NOM.CURR."	U32	100 (*)	13200 (*)	
0x011C	284	0x03	P269	"MOTOR NOM.SPEED"	U16	1000	3600	
0x011D	285	0x03	P290	"STC MOTOR PROT."	U08	2	4	
0x011E	286	0x03	P291	"STC MOTOR THERMAL"	U08	0	100	
0x012B	299	0x03	P202	"SOFTWARE"	U32	0 Note 6	0xFFFFFFFF Note 6	
0x012D	301	0x03		"Errors, H3"	_B2	0 Note 7	0xFFFFFFFF Note 7	
0x0137	311	0x03	P1225	"SSID NUMBER"	_B2	0	0xFFFFFFFF	
0x0139	313	0x03	P1226	"SEC.KEY NUMBER"	_B2	0	0xFFFFFFFF	
0x013F	319	0x03, 0x06, 0x10	P905	"OFFS.INPUT"	U08	0	6 (*)	0 = Off 1 = An.U1 0-10V 2 = An.U2 0-10V 3 = An.I1 0-20mA 4 = An.I1 4-20mA 5 = An.I2 0-20mA 6 = An.I2 4-20mA
0x0140	320	0x03, 0x06, 0x10	P907	"OFFSET RANGE"	U16	0	10000	

Modbus address		Function codes	Menu Index	Description	Datum type	Values range		Enum Description
(HEX)	(DEC)					Minimum	Maximum	
0x0141	321	0x03, 0x06, 0x10	P910	"LEVEL 1"	U16	0	10000	
0x0142	322	0x03, 0x06, 0x10	P912	"OFFSET X1"	U16	0	10000	
0x0143	323	0x03, 0x06, 0x10	P913	"OFFSET Y1"	U16	0	P420 Note 1	
0x0144	324	0x03, 0x06, 0x10	P915	"LEVEL 2"	U16	0	10000	
0x0145	325	0x03, 0x06, 0x10	P917	"OFFSET X2"	U16	0	10000	
0x0146	326	0x03, 0x06, 0x10	P918	"OFFSET Y2"	U16	0	P420 Note 1	
0x0149	329	0x03	P1304	"PRE-SET MOTOR?"	U08	0	1	
0x014A	330	0x03 0x06, 0x10 (Note 9)	P1311	"CONTROL MODE"	U08	0	1	
0x014B	331	0x03, 0x06, 0x10	P1322	"START-UP COMPLETE?"	U08	0	1	
0x01C1	449	0x03		"Extended Device Status, H4"	_B1	0 Note 8	0xFFFF Note 8	
0x0131	305	0x03	P135	"BACK.COMP."	U08	0	1	

Data types are:

- U08 – Unsigned 8 bits
- U16 – Unsigned 16 bits
- U32 – Unsigned 32 bits (can be written by mean of Write Multiple Register only, function code 0x10)
- S08 – Signed 8 bits
- S16 – Signed 16 bits
- \_B0 – 8 bits bitmap
- \_B1 – 16 bits bitmap
- \_B2 – 32 bits bitmap

(\*) : parameter have values boundaries/selections that could be available only based on HVL model variant, and/or optional cards

**Note 1:** some parameters have values, properties and/or boundaries that are runtime dependent on other parameters values. In particular, if a parameter value range (min. value, max. value) is reported to have dependence on other parameters value(s), the device may report Modbus write exceptions on the dependent parameter.

**Note 2:** exclusively whilst the motor is running, HVL answers an exception to Modbus writing function requests (0x06, 0x10)

**Note 3:** parameter data fields: DATE

U32 0xMNPQRSTV	Value ranges	Description	Formatted on HV display as text “DD-MM-20YY”
Byte 0xMN	0..99	Year	the decimal value is shown on display at position YY
Byte 0xPQ	1..12	Month	the decimal value is shown on display at position MM
Byte 0xRS	1..31	Day	the decimal value is shown on display at position DD
Byte 0xTV	0..255	reserved/for internal use	

**Note 4:** parameter data fields: TIME

U32 0xMNPQRSTV	Value ranges	Description	Formatted on HV display as text “hh:mm”
Byte 0xMN	0	reserved/for internal use	
Byte 0xPQ	0..23	Hours	the decimal value is shown on display at position hh
Byte 0xRS	0..59	Minutes	the decimal value is shown on display at position mm
Byte 0xTV	0..59	reserved/for internal use	

**Note 5:** parameter data fields: TIME COUNTERs

U32 0xMNPQRSTV	Value ranges	Description	Formatted on HV display as text “hhhhh:mm”
Word 0xMNPQ	0..65535	Hours	the decimal value is shown on display at position hhhhh
Byte 0xRS	0..59	Minutes	the decimal value is shown on display at position mm
Byte 0xTV	0..255	reserved/for internal use	

Examples:

0x00082000, 8hours, 32 minutes, other byte: reserved, the time counter is 8:32  
0x0220013B, 544hours, 1 minute, other byte: reserved, the time counter is 544:01

**Note 6:** parameter data fields: VERSIONs

U32 0xMNPQRSTV	Value ranges	Description	Formatted on HV display as text “V##.**”
Byte 0xMN	0..99	Major version value	the two digits decimal value is shown at position ##
	100..255	--	“- -” is shown on display at position ##
Byte 0xPQ	0..99	Minor version value	the two digits decimal value is shown at position **
	100..255	--	“- -” is shown on display at position **
Byte 0xRS	0..255	reserved/for internal use	
Byte 0xTV	0..255	reserved/for internal use	

## Examples:

0x02080000, major version:2, minor version: 8, other bytes: reserved, the version is 2.08, display format is “V02.08”.  
 0x0220ABAB, major version:2, minor version: 32, other bytes: reserved, the version is 2.32, display format is “V02.32”.  
 0x02FFAA55, major version:2, minor version: --, other bytes: reserved, the version is 2.--, display format is “V02.--”.  
 0xFFFFFFFF, major version:--, minor version: --, other bytes: reserved, the version is --.--, display format is “V--.--”.

**Note 7:** parameter data fields: Error status

_B2	Value ranges	Description
Bit 00 Bit 01 Bit 02 Bit 03 Bit 04 Bit 05 Bit 06 Bit 07 Bit 08 Bit 09 Bit 10 Bit 11	0 = error absent 1 = error present	ERROR 11 OVERCURRENT ERROR 12 OVERLOAD ERROR 13 OVERVOLTAGE ERROR 16 PHASE LOSS ERROR 14 INVERTER OVERHEAT ERROR 15 MOTOR OVERHEAT ERROR 21 LACK OF WATER ERROR 22 MINIMUM THRESHOLD ERROR 23 ACT. VAL. SENSOR 1 ERROR 24 ACT. VAL. SENSOR 2 ERROR 25 SETPOINT 1 I<4mA ERROR 26 SETPOINT 2 I<4mA
Bits 12.. 31	0..1	Reserved/for internal use

## Example:

0x00000040, Bit 06 is set to 1, “Lack Of Water” error #21 is present.



**Note 8:** parameter data fields: Extended Device Status

_B2	Value ranges	Description
Bit 00 Bit 01 Bit 02 Bit 03 Bit 04 Bit 05 Bit 06	0 = bit not set = False 1 = bit set = True	Device is preset Device is ready for regulation (but maybe stopped) Device has an error Device has a warning External ON/OFF terminal enabled/disabled Device is enabled with start button Motor is running
Bit 07	0..1	Reserved/for internal use
Bits 08 .. 11	0..8 9..15	Group sequence number of the pump (0 if pump is not in group) Reserved/for internal use
Bit 12 Bit 13 Bit 14 Bit 15	0 = bit not set = False 1 = bit set = True	Control card is present Device is master Solo-Run ON/OFF Inverter STOP/START

**Note 9:** parameters associated to internal device reconfiguration actions: a Modbus writing action to the parameter will restore other HVL dependent parameters to their default values. It is recommended to not write to the parameter unless the present value differs from the desired value.

