

INF-B ModBus Communication Option Operator's Manual Supplement

Note: To Enable the MODBUS PROTOCOL via Front LED Display Panel Push Button Menu, enter **MOdbUS** submenu and select option “**yES**” in the “**COMM**” Communication Configuration Menu.

Note: [INF-B Communication Option Operator's Manual](#) is required as Reference Source.

1. Introduction

Modbus protocol defines a message structure that INF-B Universal Input Meters will recognize and use, regardless of the type of networks over which they communicate. It describes the process a meter 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.

The Modbus protocol provides the internal standard that the INF-B universal Input Meters use for parsing messages. During communications on a Modbus network, the protocol determines how each meter 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. If a reply is required, the meter will construct the reply message and send it using Modbus protocol.

Modbus defines a digital communication network to have only one MASTER and one or more SLAVE devices. Either a single (point-to-point) or multi-drop network (multipoint) is possible.

INF-B Universal Input Meters communicate on standard Modbus networks using RTU (Remote terminal unit) transmission mode.

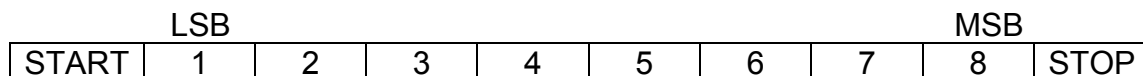
2. RTU mode

In RTU mode, each eight-bit byte in a message contains two four-bit hexadecimal characters. The main advantage of this mode is that its greater character density allows better data throughput than ASCII for the same baud rate. Each message must be transmitted in a continuous stream.

The following format used for each byte sent and received by INF-B Meters in RTU mode:

- 2.1. Eight-bit binary, Hexadecimal (0 ... 9, A ... F)
- 2.2. Two hexadecimal characters contained in each eight-bit field of the message
- 2.3. 1 start bit, 8 data bits, 1 Stop Bit (No Parity Bit)

❖ The figure below shows the bit sequences when byte transmitted in RTU mode.



- LSB – Least Significant Bit sent first

❖ The Modbus Message frame is shown below

DEVICE ADDRESS	FUNCTION CODE	DATA	CRC CHECK
8 BITS hh	8 BITS hh	k x 8 BITS hhh....	16 BITS hhhh

where:

- h (hex. Number) – character,
- k – integers depend on the contents of the data format.

3. Device Address

- The address message frame contains eight bits. The slave device addresses are in the range of 1 ... 199 decimal. A master addresses a slave by placing the slave address in the address field of the message. When the slave sends its response, it places its own address in this address field of the response to let the master know which slave is responding.
- Address 0 is used for the write command broadcast that commands all controllers on network, which all slave devices recognize

4. Function Code

- The function code field of a message frame contains eight bits (RTU). Valid codes are in the range of 1 ... 255 decimal. Of these, some codes are applicable for INF-B Universal Input Meters.
- When a message is sent from a master to a slave device the function code field tells the slave what kind of action to perform.
- The following functions are supported by INF-B Universal Input Meters:

Function Code	Function	Description
03	Read holding register	Reads the binary contents of holding registers in the slave
04	Read input register	Reads the binary contents of input register in the slave
06	Preset (Write to) single register	Preset (Write) a value into single holding register
08	Diagnostic	Series of tests for checking communication between master and slave

- When the slave responds to the master, it uses the function code field to indicate either a normal (error-free) response or that some kind of error occurred (called an exception response). For a normal response, the slave simply echoes the original function code. For an exception response, the slave returns a code that is equivalent to the original function code with its most significant bit set to a logic 1.

5. Data Field

The data field is constructed using sets of two hexadecimal digits, in the range of 00 to FF hexadecimal. The data field of messages sent from a master to slave devices contains additional information, which the slave must use to take the action defined by the function code. This can include items like discrete and register addresses, the quantity of items to be handled, and the count of actual data bytes in the field.

6. CRC Checking

With RTU mode the error checking field contains a 16-bit value implemented as two eight-bit bytes (HI-order byte and Low-order byte). The error check value is the result of a Cyclical Redundancy Check (CRC) calculation performed on the message contents.

After building a message (address, function code, data) the transmitting device calculates a CRC code and puts it to the end of the message. A receiving device will calculate a CRC code from the message it has received and compare against transmitted CRC code. If these CRC codes are different, there has been a communication error. INF-B Universal Input Meters will not reply if they detect a CRC error.

Sequences of CRC calculation:

- 6.1 Load a 16 bit CRC register with all 1's.
- 6.2 Apply first 8 bit byte of the message to the least significant bit (LSB) of the contents of the register.
- 6.3 Exclusive OR these 8 bit with the register contents.
- 6.4 Shift the result one bit to the right with zero entering into the most significant bit (MSB) position and evaluate the LSB.
- 6.5 If over flow bit in LSB is 1, exclusive OR the latest register contents with A001 Hex value.
- 6.6 If over flow bit in LSB is 0, no exclusive OR occurs (repeat step 4).
- 6.7 Repeat steps 4, 5 and 6 until 8 shifts have been performed.
- 6.8 Apply next 8 bit byte of the message to the LSB contents of the register.
- 6.9 Exclusive OR these 8 bit with the register contents.
- 6.10 Repeat steps 4 to 9 until all bytes of the message have been processed.
- 6.11 The final content of the register is the CRC value.

❖ **Note:** When CRC is placed into the end of the message, the low order byte of the CRC will be transmitted first, followed by the high order byte.

7. Modbus RTU Registers

- The table below shows the **Modbus registers** supported by INF-B Universal Input Meters:

FUNCTION CODE	REGISTER	FUNCTION	NUMBER OF BYTE	Refer to Section of Comm. Manual: (A): INF ; (B): INF-B
03/04, 06	01	SP1 : SETPOINT 1 VALUE	3	(*)
03/04, 06	02	SP2 : SETPOINT 2 VALUE	3	(*)
03/04, 06	03	SP3 : SETPOINT 3 VALUE	3	(*)
03/04, 06	04	SP4 : SETPOINT 4 VALUE	3	(*)
03/04, 06	05	RdG.SC : READING SCALE VALUE	3	(B)10.34
03/04, 06	06	RdG.OF : READING OFFSET VALUE	3	(*)
03/04, 06	07	INPUT SCALE VALUE	3	(B)10.34
03/04, 06	08	INPUT OFFSET VALUE	3	(*)
03/04, 06	09	OUTPUT SCALE VALUE	3	(B)10.34
03/04, 06	0A	OUTPUT OFFSET VALUE	3	(*)
03/04	0B	MAIN READING VALUE	3	(*)
03/04	0C	HI RdG : PEAK READING VALUE	3	(*)
03/04	0D	LO RdG : VALLEY READING VALUE	3	(*)
03/04, 06	0E	dAt.FMt : DATA FORMAT	1	(B)10.12
03/04, 06	0F	bUS.FMt : BUS FORMAT	1	(B)10.13
03/04, 06	10	INP.CNF : INPUT CONFIG.	1	Table 10.2
03/04, 06	11	FILtER : FILTER SETUP	1	(B)10.4
03/04, 06	12	RdG.CNF : READING CONFIG.	1	Example 4
03/04, 06	13	Out.CNF : OUTPUT CONFIG.	1	(B)10.19
03/04, 06	14	dEC.Pt & CNt.by : dec.point & count by setup	1	(B)10.3
03/04, 06	15	INPUt : INPUT TYPE	1	(B)10.10
03/04, 06	16	SP.CNF : SETPOINT CONFIG.	1	(B)10.5
03/04, 06	17	AL.CNFG : ALARM CONFIG.	1	(B)10.6
03/04, 06	18	ALARM FUNCTIONS	1	(B)10.7
03/04, 06	19	NUM.dLy : ALARM DELAY	1	(B)10.4
03/04, 06	1A	SERIAL COMM.CONFIGURATION	1	(B)10.8
03/04, 06	1B	AddRES : SERIAL COMM. ADDRESS	1	(B)10.29
03/04, 06	1C	CHAR. : SERIAL COMM.RECOGN.CHAR	1	(B)10.31
03/04, 06	1D	MENU LOCKOUT	1	(B)10.14 &15
03/04, 06	1E	MENU LOCKOUT & NORMAL COLOR CONFIG.	1	(B)10.14 &15
03/04, 06	1F	SP1,SP2 & ALARM COLOR 1&2	1	(B)10.16
03/04, 06	20	Only applied for MENU 2 : L4 CNF	1	(A):10-3
03/04, 06	21	SP.db : SetPoint Deadband	2	(B): 10.36
03/04, 06	22	AL.db : Alarm Deadband	2	(B): 10.37

Table 7.1 Modbus Registers

(*) **Note:** Refer to Example 3 and 6 for description of How to configure and its format.

8. Command Format

The following formats are used to SEND commands by computer and RETURNED by device.

8.1. Read Multiple Register (03 or 04)

❖ Send to device Command string format:

DEVICE ADDRESS	FUNCTION CODE 03 or 04	DATA				CRC	
		STARTING REGISTER		NUMBER OF REGISTERS			
1 BYTE hh	1 BYTE 03	HB 00	LB hh	HB 00	LB hh	LB hh	HB hh

❖ Returned by device Command string format:

DEVICE ADDRESS	FUNCTION CODE 03 or 04	DATA						CRC	
		NUMBER OF BYTES	First REGISTER	n REGISTERS				
1 BYTE hh	1 BYTE 03	1 BYTE hh	HB hh	LB hh	HB hh	LB hh	LB hh	HB hh

Where: HB – High Order Byte
 LB – Lower Order Byte
 Unused bits are set to zero
 hh : Hex. Numbers

❖ **Note:** INF-B Universal Input Meters support only Read Single Register, so the number of registers should always set to 1.

❖ Screenshot for following examples 1- 3:

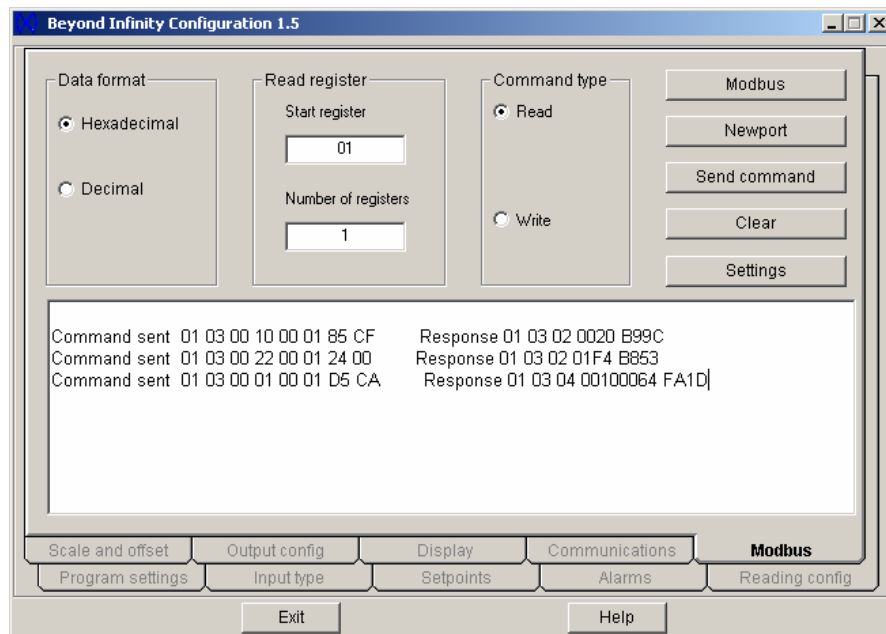


Figure 8.1: Read Command Samples via Infinity Configuration Software.

Example 1: For one byte data registers: 1st data string of Command sent section as shown on the Figure 8.1:

- ❖ 01 03 0010 0001 85CF : is to read INPUT CONFIGURATION (*INF.CNF* and *INPUT*).

DEVICE ADDRESS	FUNCTION CODE 03 or 04	DATA				CRC	
		STARTING REGISTER		NUMBER OF REGISTERS			
01	03	00	10	00	01	85	CF

- ❖ 01 03 02 0000 B844 : 1st Command Response on Figure 8.1 which device responded to the 1st read command.

DEVICE ADDRESS	FUNCTION CODE	NUMBER OF BYTES	VALUES OF REGISTERS (1 Byte)		CRC	
01	03	02	00 (N/A)	20	B9	9C

For detail description of the VALUES OF THE REGISTER above, refer to Table 10.2 of INF-B Communication Manual.

Example 2: For 2 bytes data registers: as shown on the Figure 8.1 as 2nd data string of the Command sent section:

- ❖ 01 03 0022 0001 2400: is to read the value of Alarm Deadband (*AL.db*).

DEVICE ADDRESS	FUNCTION CODE 03 or 04	DATA				CRC	
		STARTING REGISTER		NUMBER OF REGISTERS			
01	03	00	22	00	01	24	00

- ❖ 01 03 02 01F4 B853 : 2nd Command Response on Figure 8.1 which device responded to the 2nd read command.

DEVICE ADDRESS	FUNCTION CODE	NUMBER OF BYTES	VALUES OF REGISTERS (2 Bytes)		CRC	
01	03	02	01	F4	B8	53

For detail description of the VALUES OF THE REGISTER above, refer to Section 10.37 of INF-B Communication Manual.

Example 3: For 3 bytes data registers: as shown on the Figure 8.1 as 3rd data string of the Command sent section:

- ❖ 01 03 0001 0001 D5CA: is to read Setpoint 1 value (**SP1**).

DEVICE ADDRESS	FUNCTION CODE 03 or 04	DATA				CRC	
		STARTING REGISTER		NUMBER OF REGISTERS			
01	03	00	01	00	01	D5	CA

- ❖ 01 03 04 0010 0064 FA1D: 3rd Command Response on Figure 8.1 which device responded to the 3rd read command.

DEVICE ADDRESS	FUNCTION CODE	NUMBER OF BYTES	VALUES OF REGISTERS (3 Bytes)			CRC	
01	03	04	00 (N/A)	10	00	64	FA 1D

- ❖ **NOTE:** SetPoints, In/Output Offset and process values (Main, Peak and Valley Reading Values) must be determined in following manner: (using above example)

Value Format	VALUES OF REGISTERS (3 Hex. Bytes)					
Hex.	00	10		00	64	
Binary bits pattern	N/A	XXXX (binary bits)	XXXX (binary bits)	XXXX XXXX (binary bits)	XXXX XXXX (binary bits)	
Binary	N/A	0001		0000	0000 0000	0110 0100
Setpoints, In/Output Offset, Main/ Peak/ Valley Readings	N/A	X	XXX	Absolute Value	Absolute Value	Absolute Value
		0	001			
		SIGN	DECIMAL POINT			
		+	FFFF.			
Equivalent Decimal	+100.					

Table 8.1 Format of 3 bytes Data Value Description of SetPoints, In/Output Offset and process values.

For detail description of How to determine Sign (+ or -) and Decimal Point Position, refer to Section 10.28 and Table 10.16 of INF-B Communication Manual.

- ❖ **NOTE:** Following bits patterns is the format for 3 bytes data registers of **READING SCALE (“RdG.SC”), INPUT SCALE (“INP.SC”), AND OUTPUT SCALE (“Out.SC”)** which are determined differently than the above example.

Value Format	VALUES OF REGISTERS (3 Hex. Bytes)						
Hex. Pattern	00	hh (2 hex.characters)		hh (2 hex.characters)		hh (2 hex.characters)	
Binary bits pattern	N/A	XXXX (binary bits)	XXXX (binary bits)	XXXX XXXX (binary bits)		XXXX XXXX (binary bits)	
Decimal	N/A	DECIMAL POINT	X	XXX	Absolute Value		Absolute Value
			SIGN	Absolute Value			

Table 8.2 : Format of 3 bytes Data Value Description of all type of Scale Function.

For detail description of How to determine Sign (+ or -), Decimal Point Position and absolute value, refer to Section 10.34 and Table 10.28 of INF-B Communication Manual.

8.2. Write to Single Register (06)

The following command will write a parameter to the single register.
Send / Response Command string format:

DEVICE ADDRESS	FUNCTION CODE 06	DATA				CRC	
		REGISTER		DATA VALUE			
1 BYTE hh	1BYTE 06	HB 00	LB hh	HB 00	LB hh	LB hh	HB hh

Example 4: For Write Command of one byte data registers: using example on Section & Table 10.1 of INF-B Serial Communication Option Operator's manual to learn how to modify Reading configuration (*RdG.CNF*), sent/responded data string as following:

DEVICE ADDRESS	FUNCTION CODE 06	DATA				CRC	
		REGISTER		DATA VALUE (1 valid byte)			
1 BYTE 01	1 BYTE 06	HB 00	LB 12	HB 00	LB 14	LB 29	HB C0

Here is the screenshot of the Example 4:

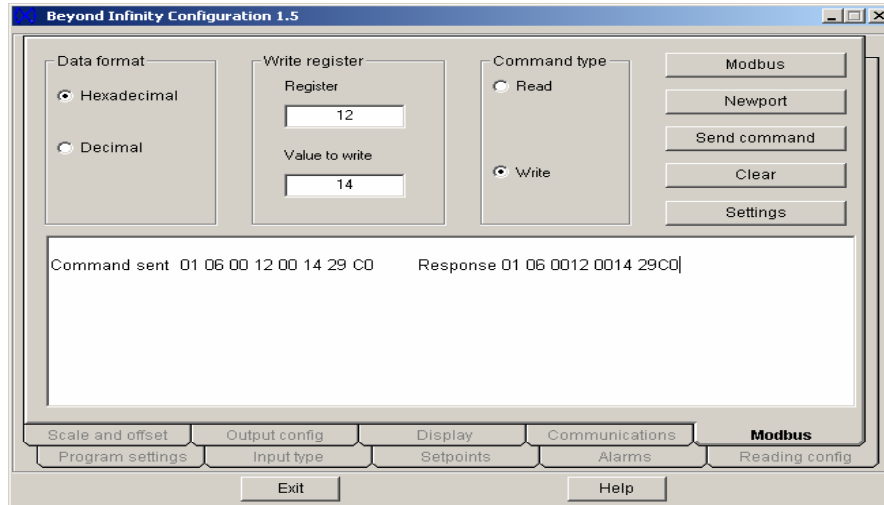


Figure 8.2: Write (1 valid Data byte) Command Sample via Infinity Configuration Software.

Example 5: For Write Command of two bytes data registers: There are only two registers in this format: Setpoint Deadband (**SP.db**): Register #21 and Alarm Deadband (**AL.db**) Register #22.

Using Example on Section 10.36 of INF-B Serial Communication Option Operator’s manual to write new value of SetPoint Deadband as 6,800 count.

DEVICE ADDRESS	FUNCTION CODE 06	DATA				CRC	
		REGISTER		DATA VALUE (2 valid bytes)			
1 BYTE 01	1BYTE 06	HB 00	LB 21	HB 1A	LB 90	LB D2	HB C2

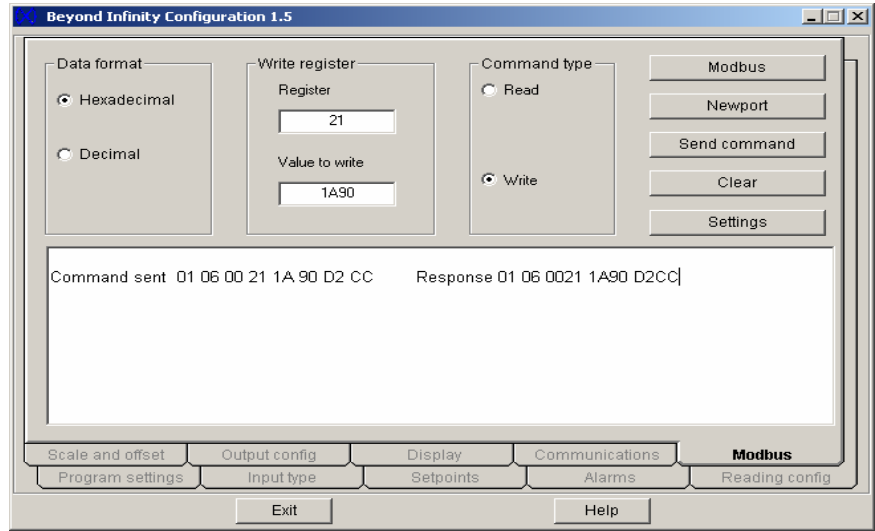


Figure 8.3: Write (2 valid Data bytes) Command Sample via Infinity Configuration Software.

Example 6: For Write Command of three bytes data registers: In order to modify or configure, users must send two write commands to accomplish this task. First one only change register’s 2 Low Order Bytes (LB)and is just similar to the previous example , However second command must write on converted register of original one to write or modify value of High Order Byte (HB)(as example below illustrate How to set decimal point and sign of value for detail description of Sign & Decimal Point configuration, refer to Table 10.26 of INF-B Communication).

Following Table illustrates how to convert the register:

Register	Format	Register (Hex. Number)	Register (Binary Number)	
		<u>Method:</u> change MSB of character 0 of Register to 1	hh	<u>MSB</u> x x x
Example:	Reg. of SP1 value	01	0000	0001
	Converted SP1 value:	81	1000	0001

Table 8.3: Conversion Method of 3 Bytes Register for Write Function.

Lets examine how to write SP1 with different values: +1000 and -100.0:
 Send / Response Command string format:

❖ Value +1000:

Desired Values: +1000	DEVICE ADDRESS	FUNCTION CODE 06	DATA				CRC	
			REGISTER		DATA VALUE (2 bytes used)			
1 st command (for absolute value)	1 BYTE 01	1BYTE 06	HB 00	LB 01	HB 03	LB E8	LB D8	HB B4
2 nd command (for Sign & dec.point)	1 BYTE 01	1BYTE 06	HB 00	LB 81	HB 00	LB 10	LB D8	HB 2E

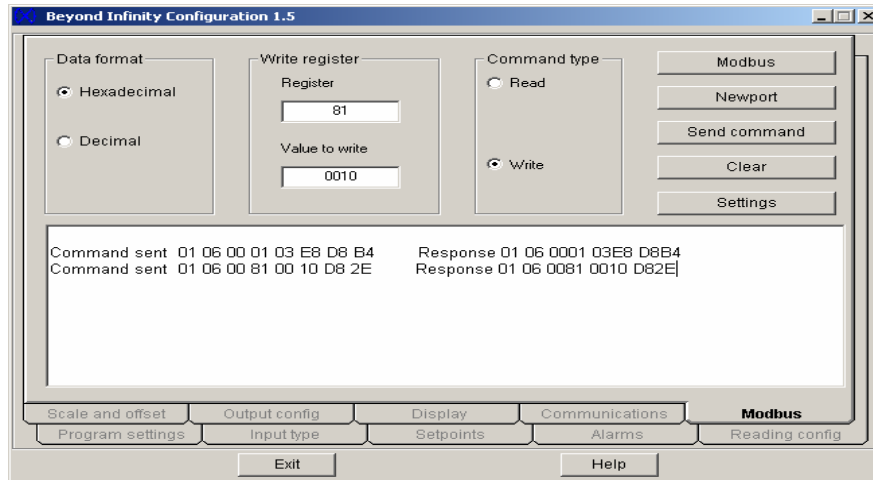


Figure 8.4: Write (3 Data bytes) Commands Sample via Infinity Configuration Software.

❖ Value -100:

Desired Values: -100	DEVICE ADDRESS	FUNCTION CODE 06	DATA				CRC	
			REGISTER		DATA VALUE (2 bytes used)			
1 st command (for absolute value)	1 BYTE 01	1BYTE 06	HB 00	LB 01	HB 00	LB 64	LB D8	HB B4
2 nd command (for Sign & dec. point)	1 BYTE 01	1BYTE 06	HB 00	LB 81	HB 00	LB 90	LB D8	HB 2E

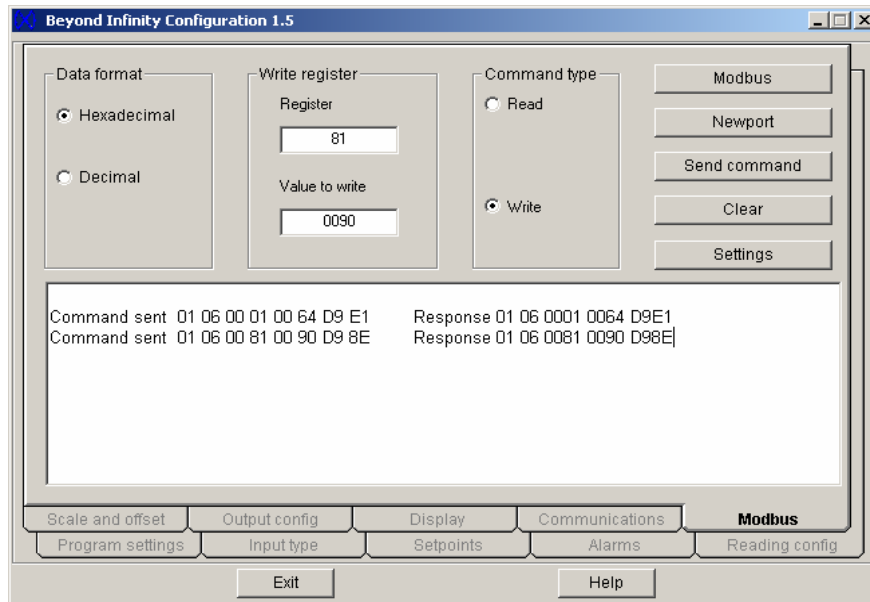


Figure 8.5: Write (3 Data bytes) Commands Sample via Infinity Configuration Software.

8 Diagnostic command.

This command echoes the sent message to indicate that the communication link is established correctly.

Send to/Return from device:

DEVICE ADDRESS	FUNCTION CODE	DIAGNOSTIC CODE		LOOPBACK DATA		CRC	
		HB	LB	HB	LB	LB	HB
1 BYTE hh	1 BYTE 08	00	00	hh	hh	hh	hh

where:

- Diagnostic Code is two byte code to determine the type of test to be performed.
- INF-B Universal Input Meters supported only “00” code which requested slave to echo sent command back to the master.