FAST EnergyCam wired M-Bus Slave Protocol

Table of Contents

Introduction 1
Supported M-Bus Telegrams 2
SND_NKE 2
REQ_UD2 2
Example response for energy meter (Wh) 2
Example response for gas meter (m³) 3
Example response for water meter (m³) 3
Detailed description of example response to REQ_UD2 3
Set SlaveAddress 5
Set Medium 5
Set BaudRate 6
Set back to Modbus protocol 6
Manufacture specific data block 6
Get AppFirmwareBuildNumber 7
Set ActionOCRInstallation 7
Set ActionOCR 7
Set ActionPowerDown 8
History 9

Index of Tables

Table 1: SND_NKE (example for slave address 3) 2
Table 2: Request for REQ_UD2 (example for slave address 3) 2
Table 3: Example response to REQ_UD2 (energy in Wh) 2
Table 4: Major data in example response for electricity meter (energy in Wh) 2
Table 5: Example response to REQ_UD2 gas meter (volume in m³) 3
Table 6: Major data in example response for gas meter (volume in m³) 3
Table 7: Example response to REQ_UD2 water meter (volume in m³) 3
Table 8: Major data in example response for water meter (volume in m³) 3
Table 9: Detailed description of example response to REQ_UD2 5
Table 10: VIF bytes coding for medium UNKNOWN 5
Table 11: Telegram to set new slave address 5
Table 12: Telegram to set new medium (example for slave address 3) 5
Table 13: Telegram to set new baud rate (example for slave address 1) 6
Table 14: Telegram to set back to Modbus protocol (example for slave address 3) 6
Table 15: Telegram to get firmware build number with tunneled Modbus telegram (TX) 7
Table 16: EnergyCam response telegram with firmware build number (RX) 7
Table 17: Trigger an OCR installation with tunneled Modbus telegram (TX) 7
Table 18: EnergyCam response for successful write to holding register and trigger OCR installation (RX) 7
Table 19: Trigger an OCR reading with tunneled Modbus telegram (TX) 7
Table 20: EnergyCam response for successful write to holding register and trigger OCR reading (RX) 8
Table 21: Trigger a power down with tunneled Modbus telegram (TX) 8
Table 22: EnergyCam response for successful write to holding register and trigger of a power down (RX) 8
Table 23: History 9

Introduction

FAST FORWARD EnergyCam is able to communicate via a wired connection in M-Bus protocol according to EN1434-3 (EC13757). The interface uses even parity and 1 stop bit.

See also http://www.m-bus.com.

Supported M-Bus Telegrams

Supported baud rates are: 1200, 2400, 9600, 19200, 38400, no auto baud detection.

Selection by primary address. Selection by secondary address with wildcards.

SND_NKE

Standard SND_NKE is supported, response is standard 0xE5.

Byte	1	2	3	4	5
Name	Start	C	A	CRC	Stop
Hex	10	40	03	43	16

Table 1: SND_NKE (example for slave address 3)

REQ_UD2

EnergyCam needs a standard REQ_UD2 as follows:

Byte	1	2	3	4	5
Name	Start	C	A	CRC	Stop
Hex	10	7B	03	7E	16

Table 2: Request for REQ_UD2 (example for slave address 3)

Example response for energy meter (Wh)

EnergyCam responds to a standard REQ_UD2 as follows (example):

Byte	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27
Name	Start	Len	Len	Start	C	A	CI	ID1	ID2	ID3	ID4	MAN1	MAN2	VER	MED	TC	Stat	SIG1	SIG2	DIF	VIF	OCR1	OCR2	OCR3	OCR4	CRC	Stop
Hex	68	15	15	68	08	03	72	64	16	10	23	C4	18	01	02	00	00	00	00	04	05	FD	85	0A	00	9E	16

Table 3: Example response to REQ_UD2 (energy in Wh)

Example shows a Meter configured as:

What	Value
Manufacturer ID	"FFD" (Three letter code), not changeable by user
Ident Number	0x23101664, not changeable by user
Version	0x01, not changeable by user
Type	0x02 (electricity), changeable by user
Value	68966.1 kWh

Table 4: Major data in example response for electricity meter (energy in Wh)

Example response for gas meter (m³)

EnergyCam responds to a standard REQ_UD2 as follows (example):

Byte	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27
Name	Start	Len	Len	Start	C	A	CI	ID1	ID2	ID3	ID4	MAN1	MAN2	VER	MED	TC	Stat	SIG1	SIG2	DIF	VIF	OCR1	OCR2	OCR3	OCR4	CRC	Stop
Hex	68	15	15	68	08	03	72	64	16	10	23	C4	18	01	03	00	00	00	00	04	15	FD	85	0A	00	AF	16

Table 5: Example response to REQ_UD2 gas meter (volume in m³)

Example shows a Meter configured as:

What	Value
Manufacturer ID	"FFD" (Three letter code), not changeable by user
Ident Number	0x23101664, not changeable by user
Version	0x01, not changeable by user
Type	0x03 (gas), changeable by user
Value	68966.1 m³

Table 6: Major data in example response for gas meter (volume in m³)

Example response for water meter (m³)

EnergyCam responds to a standard REQ_UD2 as follows (example):

Byte	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27
Name	Start	Len	Len	Start	C	A	CI	ID1	ID2	ID3	ID4	MAN1	MAN2	VER	MED	TC	Stat	SIG1	SIG2	DIF	VIF	OCR1	OCR2	OCR3	OCR4	CRC	Stop
Hex	68	15	15	68	08	03	72	64	16	10	23	C4	18	01	07	00	00	00	00	04	15	FD	85	0A	00	B3	16

Table 7: Example response to REQ_UD2 water meter (volume in m³)

Example shows a Meter configured as:

What	Value
Manufacturer ID	"FFD" (Three letter code), not changeable by user
Ident Number	0x23101664, not changeable by user
Version	0x01, not changeable by user
Type	0x07 (water), changeable by user
Value	68966.1 m³

Table 8: Major data in example response for water meter (volume in m³)

Detailed description of example response to REQ_UD2

Byte #	Value [hex]	Name	Description
1	68	Start	
2	15	Len	
3	15	Len	
4	68	Start	

Byte #	Value [hex]	Name	Description
5	08	C	
6	03	A	
7	72	CI	
8	64	ID1	IdentNumber [0]: 0x23101664 (read as 8 BCD numbers)
9	16	ID2	IdentNumber [1]:
10	10	ID3	IdentNumber [2]
11	23	ID4	IdentNumber [3]
12	C4	MAN1	ManufacturerID [0]: **0x18C4**="FFD" for FastForwarD (see www.dlms.com/flag)
13	18	MAN2	ManufacturerID [1]: Multibyte fields are transmitted lower byte first
14	01	VER	Version of Meter (constant, always 1 for EC), forms together with Manufacturer-ID, IdentNumber and Type a unique wM-Bus address
15	02	MED	Type of Meter (medium) 0x01 Oil **0x02** Energy (electricity) 0x03 Gas 0x07 Water 0x0F UNKNOWN (see VIF dimensionless)
16	00	TC	Transmission counter (access number)
17	00	STAT	Signals error STATe of meter ST **0x00**: no errors (Other codes here not used by EC)
18	00	SIG1	SIGnature1
19	00	SIG2	SIGnature2
20	04	DIF	DataInformationField for OCR value [7] Extension bit **0**: no DIF byte following (i.e. next byte is VIF) 1: next byte is also a DIF byte [6] Storage number (not used for EC) [5:4] Function field: **0**: instantaneous value (default for EC) 3: value during error state (e.g. repeating last OCR value due to error detection by EC) x: other codes not used by EC [3:0] **4**: 32 bit integer (means 4 bytes following after VIF). Other codes here not used by EC
21	05	VIF	ValueInformationField for Exponent and OCR value (here energy in Wh) [7] Extension bit **0**: no VIF byte following (i.e. next byte is value) 1: next byte is also a VIF byte, in case of medium UKNOWN two VIF bytes follows, see Table 10: VIF bytes coding for medium UNKNOWN for details [6:3] **0**: Energy [Wh] used for electricity 2: Volume[m^3] used for gas, water and oil x: other codes not used by EC [2:0] Exponent with bias (bias depends on unit: -3 for energy, -6 for volume) 5: Exponent = 5-3 = 2 -> 10^2 factor; 689661*100 Wh = 68966100 Wh -> 68966.1 kWh 5: Exponent = **5**-6 = -1 -> 10^{-1} factor; 689661*0.1 m^3 -> 68966.1 m^3 (used for gas

Byte #	Value [hex]	Name	Description
			and water)
22	FD	Value []	Value 32 bit integer **0x000A85FD**= 689661 (has to be scaled by exponent specified in VIF)
23	85	Value []	
24	0A	Value []	
25	00	Value []	
26	9E	CRC	Check Sum is calculate from the arithmetical sum of data mentioned above, without taking carry digits into account.
27	16	Stop	

Table 9: Detailed description of example response to REQ_UD2

VIF Byte #	Value [hex]	Description
0	FD	Value 0xFD declares an “extension indicator”. This means that VIF coding is done in the following byte
1	BA	Value 0x3A defines that the data value should be interpreted “dimensionless”. MSB is set which means that another VIF byte is following
2	75	ValueInformationField for Exponent and OCR value [7] Extension bit **0**: no VIF byte following (i.e. next byte is value) 1: next byte is also a VIF byte [6:3] **14**: Multiplicative correction factor: $10^{**[6:3]-6}$ [2:0] Exponent with bias (bias -6) 5: Exponent = **5**-6 = -1 -> 10^{-1} factor; 689661*0.1 m^3 -> 68966.1 [dimensionless]

Table 10: VIF bytes coding for medium UNKNOWN

Set SlaveAddress

EnergyCam's slave address can be set with the following telegram:

Byte	1	2	3	4	5	6	7	8	9	10	11	12
Name	Start	Len	Len	Start	C	A				Anew	CRC	Stop
Hex	68	06	06	68	73	01	51	01	7A	02	42	16

Table 11: Telegram to set new slave address

Example shows how to set new slave address to 0x02 (current slave address is 0x01). Response is standard 0xE5.

Set Medium

EnergyCam's medium can be set with the following telegram:

Byte	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
Name	Start	Len	Len	Start	C	A	CI			~~ID1~~	~~ID2~~	~~ID3~~	~~ID4~~	~~MAN1~~	~~MAN2~~	~~VER~~	MED	CRC	Stop
Hex	68	0D	0D	68	73	03	51	07	79	78	56	34	12	FF	FF	FF	07	5F	16

Table 12: Telegram to set new medium (example for slave address 3)

Example shows how to set the medium to water (0x07). Response is standard 0xE5.

Note that only medium can be set, IdentNumber(ID), ManufacturerID (MAN) and Version (VER) are ignored. Valid medium codes are:

0x02: Electricity (Wh)

0x03: Gas (m^3)

0x07: Water (m^3)

Set BaudRate

EnergyCam's baud rate can be set with the following telegram:

Byte	1	2	3	4	5	6	7	8	9
Name	Start	Len	Len	Start	C	A	Baud	CRC	Stop
Hex	68	03	03	68	73	01	BB	2F	16

Table 13: Telegram to set new baud rate (example for slave address 1)

Example shows how to set the baud rate to 2400. Response is standard 0xE5.

0xBA: 1200 baud
0xBB: 2400 baud
0xBD: 9600 baud
0xBE: 19200 baud (*)
0xBF: 38400 baud (*)

(*) Note: Baud rates greater than 9600 are not supported by FAST FORWARD M-Bus communication interface. These baud rate can only be used by other physical layers like RS232/USB/RS485.

Set back to Modbus protocol

EnergyCam's protocol can be changed to Modbus with the following telegram:

Byte	1	2	3	4	5	6	7	8	9	10
Name	Start	Len	Len	Start	C	A			CRC	Stop
Hex	68	04	04	68	73	03	50	F1	B7	16

Table 14: Telegram to set back to Modbus protocol (example for slave address 3)

This sets EnergyCam to Modbus protocol with default settings (slave address 1, 115200 baud, 8E1). Response is standard 0xE5.

Note that due to the high baud rate EnergyCam cannot communicate with the FAST FORWARD M-Bus communication interface anymore. Use e.g. the FAST FORWARD USB communication interface.

Manufacture specific data block

The EN 1434 standard allows to transfer vendor-specific data in a SND_UD message. The MDH (Manufacturer Data Header) consists of the character 0x0F (DIF) and indicates that all following data are manufacture specific. The length of this data is calculated from the L-Field minus the length of the so-called standard data (C-Field, A-Field, CI-Field and the data up to and including the data block 0x0F).

The manufacture specific data consists of standard Modbus commands for read input registers and read/write holding registers. Please have a closer look at our FAST FORWARD Modbus Slave Protocol documentation (FAST_EnergyCam-Protocol-MODBUS-Slave.pdf) for more details. In this case of tunneled Modbus commands the slave address of the Modbus packet is not evaluated, only the CRC must be correct. For this reason, the tunneled Modbus slave address is set to the reserved address 255 (0xFF) by default.

The following telegrams are examples for standard Modbus commands wrapped in an M-Bus telegram. As a user you don't have to change the orange marked Modbus data.

This allows access to all Modbus registers of EnergyCam via M-Bus, even a fimware update is possible.

Get AppFirmwareBuildNumber

Here is an example for a multiple reading of a Modbus input register starting at 0x000B (refers to AppBuildnumber register 0x000C), which returns EnergyCam's current firmware build number 0x2CF5 (Build 11509).

Byte	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
Name	Start	Len	Len	Start	C	A	CI	DIF	Slave	FC	Add. (H)	Add. (L)	No. of Reg. (H)	No. of Reg. (L)	CRC (H)	CRC (L)	CRC	Stop
Hex	68	0C	0C	68	73	01	51	0F	FF	04	00	0B	00	02	15	D7	D0	16

Table 15: *Telegram to get firmware build number with tunneled Modbus telegram (TX)*

Byte	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
Name	Start	Len	Len	Start	C	A	CI	DIF	Slave	FC	Byte Cnt.	Data 1 (H)	Data 1 (L)	Data 2 (H)	Data 2 (L)	CRC (H)	CRC (L)	CRC	Stop
Hex	68	0D	0D	68	08	01	51	0F	FF	04	04	00	00	2C	F5	38	CC	95	16

Table 16: *EnergyCam response telegram with firmware build number (RX)*

Set ActionOCRInstallation

Here is an example for a multiple writing of a Modbus holding register starting at 0x001F (refers to ActionOCRInstallation address 0x0020). This command will trigger an OCR installation (timeout set to 100 =0x64 seconds).

Byte	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
Name	Start	Len	Len	Start	C	A	CI	DIF	Slave	FC	Add. (H)	Add. (L)	No. of Reg. (H)	No. of Reg. (L)	Byte Cnt.	Data 1 (H)	Data 1 (L)	Data 2 (H)	Data 2 (L)	CRC (H)	CRC (L)	CRC	Stop
Hex	68	11	11	68	73	01	51	0F	FF	10	00	1F	00	02	04	00	64	00	01	05	17	89	16

Table 17: *Trigger an OCR installation with tunneled Modbus telegram (TX)*

Byte	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
Name	Start	Len	Len	Start	C	A	CI	DIF	Slave	FC	Add. (H)	Add. (L)	No. of Reg. (H)	No. of Reg. (L)	CRC (H)	CRC (L)	CRC	Stop
Hex	68	0C	0C	68	08	01	51	0F	FF	10	00	1F	00	02	65	D0	CE	16

Table 18: *EnergyCam response for successful write to holding register and trigger OCR installation (RX)*

Set ActionOCR

Here is an another example for a single writing of a Modbus holding register starting at 0x0021 (refers to ActionOCR address 0x0022). This command will trigger an OCR reading.

Byte	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
Name	Start	Len	Len	Start	C	A	CI	DIF	Slave	FC	Add. (H)	Add. (L)	Data (H)	Data (L)	CRC (H)	CRC (L)	CRC	Stop
Hex	68	0C	0C	68	73	01	51	0F	FF	06	00	21	00	01	0D	DE	E6	16

Table 19: Trigger an OCR reading with tunneled Modbus telegram (TX)

Byte	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
Name	Start	Len	Len	Start	C	A	CI	DIF	Slave	FC	Add. (H)	Add. (L)	Data (H)	Data (L)	CRC (H)	CRC (L)	CRC	Stop
Hex	68	0C	0C	68	08	01	51	0F	FF	06	00	21	00	01	0D	DE	7B	16

Table 20: *EnergyCam response for successful write to holding register and trigger OCR reading (RX)*

Set ActionPowerDown

Here is an example for a single writing of a Modbus holding register at 0x0024 (refers to ActionPowerDown address 0x0025). This command will send EnergyCam in power down.

Byte	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
Name	Start	Len	Len	Start	C	A	CI	DIF	Slave	FC	Add. (H)	Add. (L)	Data (H)	Data (L)	CRC (H)	CRC (L)	CRC	Stop
Hex	68	0C	0C	68	73	01	51	0F	FF	06	00	24	00	01	1D	DF	FA	16

Table 21: *Trigger a power down with tunneled Modbus telegram (TX)*

Byte	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
Name	Start	Len	Len	Start	C	A	CI	DIF	Slave	FC	Add. (H)	Add. (L)	Data (H)	Data (L)	CRC (H)	CRC (L)	CRC	Stop
Hex	68	0C	0C	68	08	01	51	0F	FF	06	00	24	00	01	1D	DF	8F	16

Table 22: *EnergyCam response for successful write to holding register and trigger of a power down (RX)*

History

Date	Author	Version	Changes
17th May 13	SPR	0.1	Initial
28th Nov 13	FBL	0.2	Baud rate telegram corrected
19th Feb 14	CHZ	0.3	Add manufacture specific data
26th Feb 14	SPR	0.4	Add tx data for SND_NKE and REQ_UD2, more explanations for DIF in REQ_UD2
04th Mar 14	CHZ	0.5	Add M-Bus tunneled Modbus commands
14th Mar 14	SPR	0.6	Add M-Bus tunneled Modbus command ActionOCR
17th Mar 14	SPR	0.7	Cosmetic
1st Apr 14	SPR	0.8	Added comments for: baud rate usage, set back to Modbus
3rd Apr 14	SPR	0.9	Added link to official M-Bus web page
28th Apr 14	SPR	1.0	Unification of naming
4th Jun 2014	SPR	1.1	Added many details in UD2 response
20th Jun 2014	SPR	1.2	Fixed typo: OCR value 869661 instead of 689661
28th Oct 2014	SPR	1.3	Added comment to STAT field in response of REQ_UD2 VIF
29th Jan 2015	SPR	1.4	Added medium Oil and UNKNOWN. Added VIF bytes coding for medium UNKNOWN (VIF dimensionless)
25th Mar 2015	SPR	1.5	Corrected Filenames of referenced .pdf

Table 23: History