

Eliot

LoRaWAN sensor for measuring electricity using IR port of electricity meters

Valid for firmware V2.070

Jan 18th, 2018

In this version of FW it is possible to set the period of reading of the electricity meter remotely (+ period of repeated transmitting, if the meter has not changed), and it is possible to set the required registers of electricity meter to be read and sent (max. 10 registers).

If the registers that are not included in the meter data package are selected, or are at the end of the data package, the time when the CPU is in the active state, and the device has a greater consumption (as the communication speed is the 300 bd, and for some electricity meters the entire package DAT reads almost 2 minutes).

All values are stored in the EEPROM. By default, the 60 min period is set .

After the first connection to the meter (the EEPROM does not have any registers set to read), Eliot tries to identify the registers with T1 and T2 and then sets them (high and low tariff).

After resetting the DL message (download), the value in the EEPROM (up to delete with 0x04) remains.

After reset/on (insert battery) three messages are released: version of FW, set registry (at first switched to 0x30 0x00), and period of reading.

Different sleep mode is used in this version: instead of Stop Mode, Stand By Mode (5 uA consumption).

Identification of communication protocol version was added: each uplink message begins with one byte which identifies the version.

Watchdog is active immediately after reset for better robustness of FW.

Command 0x08 is active for on-time reading of particular registers.

Downlink

function	length DL [byte]	format	response
Request for FW version	1	0x01	0x01 A B C D E (6 bytes) – ASCII coded
Request for battery status in data package	1	0x02	0x02 A (2 bytes)
Request for sending registers, previously set	1	0x03	0x03 X (number of requested registers 1-10) A B C .. (according to X)
Setting of requested registers	3-12	0x03 X A B ...	Same principle like above. e.g. 0x03 0x04 0x02 0x03 0x99 A6 – 4 registers are requested (since next measuring) according to the table below 2, 3, 153, 166 (1.8.1, 1.8.2, 31.6.0, 0.3.3)
Clear of all registers previously set	1	0x04	0x04 0x01
Request for sending period of reading data package from electricity meter	1	0x05	0x05 X Y X is period of reading electricity meter in minutes (minimum is 5 minutes), Y is repeating of last value of electricity meter if it has not changed since last reading and they are multiples of X. It means e.g. 0x05 0A 06 = data from electricity meter will be read each 10 minutes and if there is not change 60 minutes (10 minutes * 6), it will send again last value.
Writing of reading period	3	0x05 X Y	See above
Request for list of data registers, supported by electricity meter	1	0x07	0x07, number of registers found, 32 bytes, where each bit represents availability of register according to the table below
Request for one time sending of status of particular registers (without saving into EEPROM)	3-12	0x08 X A B	Same principle like by command 0x03. e.g.: 0x08 0x04 0x08 0x99 0x9A 0xA7 Response: 0x08 0x08 R3 R2 R1 C 0x99 R3 R2 R1 0x9A R3 R2 R1 0xA7 R3 R2 R1 + 16 bit CRC same like by sending data (0xFF)
Request for sending status of automatic electricity meter identification (default on).	1	0x09	0x80 0x09 X where X=0 – automatic identification on X=1 – automatic identification off

Switchin automatic electricity meter identification	2	0x09 X	0x80 0x09 X-see above
Request for el. meter type	1	0x0A	Max. 50 bytes – ASCII – first line of message from el. meter
Request for serial number of el. meter	1	0x0B	4 bytes (uint32_t – not ASCII !)
Request for sending second communication speed with electricity meter	1	0x0C	0x80 0x0C 0-5 0 – 300Bd 1 – 600Bd 2 – 1200Bd 3 – 2400Bd 4 – 4800Bd 5 – 9600Bd
Setting second communication speed with electricity meter	2	0x0C X	X – see above
Request doe sending timeout for communication with electricity meter	1	0x0F	0x80 0x0F 1-20 Time in seconds, during this el. meter must answer to the request for communication
Setting the timeout for communication with el. meter	2	0x0F X	X – see above

Uplink

0x06	Error reading data from IR port	0x80 0x06 0x01, 0x06 0x03
0xFF	Sending data package	format 0x80 0xFF A A2 A1 A0 B B2 B1 B0 etc.. HASH where A, B are registers A2, B2 are high bytes A0, B0 are low bytes

Table of conversion codes Eliot - OBIS

Eliot	OBIS	Eliot	OBIS	Eliot	OBIS	Eliot	OBIS	Eliot	OBIS
1	1.8.0	41	9.8.0	81	1.2.4	121	41.7.0	161	13.7.0
2	1.8.1	42	9.8.1	82	2.2.0	122	61.7.0	162	33.7.0
3	1.8.2	43	9.8.2	83	2.2.1	123	2.7.0	163	53.7.0
4	1.8.3	44	9.8.3	84	2.2.2	124	22.7.0	164	73.7.0
5	1.8.4	45	9.8.4	85	2.2.3	125	42.7.0	165	14.7.0
6	2.8.0	46	21.8.0	86	2.2.4	126	62.7.0	166	0.3.3
7	2.8.1	47	41.8.0	87	15.2.0	127	15.7.0	167	21.6.0
8	2.8.2	48	61.8.0	88	15.2.1	128	35.7.0	168	41.6.0
9	2.8.3	49	22.8.0	89	15.2.2	129	55.7.0	169	61.6.0
10	2.8.4	50	42.8.0	90	15.2.3	130	75.7.0	170	0.2.2
11	3.8.0	51	62.8.0	91	15.2.4	131	16.7.0		
12	3.8.1	52	35.8.0	92	3.2.0	132	36.7.0		
13	3.8.2	53	55.8.0	93	4.2.0	133	56.7.0		
14	3.8.3	54	75.8.0	94	5.2.0	134	76.7.0		
15	3.8.4	55	1.6.0	95	6.2.0	135	3.7.0		
16	4.8.0	56	1.6.1	96	7.2.0	136	23.7.0		
17	4.8.1	57	1.6.2	97	8.2.0	137	43.7.0		
18	4.8.2	58	1.6.3	98	9.2.0	138	63.7.0		
19	4.8.3	59	1.6.4	99	1.4.0	139	4.7.0		
20	4.8.4	60	2.6.0	100	2.4.0	140	24.7.0		
21	5.8.0	61	2.6.1	101	15.4.0	141	44.7.0		
22	5.8.1	62	2.6.2	102	3.4.0	142	64.7.0		
23	5.8.2	63	2.6.3	103	4.4.0	143	9.7.0		
24	5.8.3	64	2.6.4	104	5.4.0	144	29.7.0		
25	5.8.4	65	15.6.0	105	6.4.0	145	49.7.0		
26	6.8.0	66	15.6.1	106	7.4.0	146	69.7.0		
27	6.8.1	67	15.6.2	107	8.4.0	147	11.7.0		
28	6.8.2	68	15.6.3	108	9.4.0	148	31.7.0		
29	6.8.3	69	15.6.4	109	1.5.0	149	51.7.0		
30	6.8.4	70	3.6.0	110	2.5.0	150	71.7.0		
31	7.8.0	71	4.6.0	111	15.5.0	151	91.7.0		
32	7.8.1	72	5.6.0	112	3.5.0	152	11.6.0		
33	7.8.2	73	6.6.0	113	4.5.0	153	31.6.0		
34	7.8.3	74	7.6.0	114	5.5.0	154	51.6.0		
35	7.8.4	75	8.6.0	115	6.5.0	155	71.6.0		
36	8.8.0	76	9.6.0	116	7.5.0	156	91.6.0		
37	8.8.1	77	1.2.0	117	8.5.0	157	12.7.0		
38	8.8.2	78	1.2.1	118	9.5.0	158	32.7.0		
39	8.8.3	79	1.2.2	119	1.7.0	159	52.7.0		
40	8.8.4	80	1.2.3	120	21.7.0	160	72.7.0		