

2448A2 Energy Display Developer Notes

Doc Rev 20110406

Device Name: INSTEON Energy Display

SKU: 2448A2

DevCat SubCat: 0x09, 0x0D

Power Source: 2 AAA Batteries

Description: Displays energy information from up to three iMeter Solo's (current watts used as well as cost and energy consumption per month for each device). It can also display energy info provided by a linked PLM. Built-in sensor displays room temperature on device and sends a broadcast to linked PLM every 3 minutes if temp changes.

Programming Notes:

Group 1: iMeter or PLM (1 device per group. Linking a second device will overwrite the ID in that group)

Group 2: iMeter or PLM

Group 3: iMeter or PLM

Group 4: PLM for temp broadcast

Device sends Get iMeter Status 82 00 command to linked iMeters or PLM when Synchronize button is pressed. After last iMeter response is received, device will stay awake for 3 seconds, then go to sleep mode. Stay Awake command (4 minute duration) must be sent within the 3 seconds to allow for programming of settings and links to the device.

INSTEON Commands

Message Direction	Cmd 1	Cmd 2	Data 1	Notes, Data 2...
Tx	31 (1F)	0		Rx unit will send back bit 0= Plock bit 1 = Beep bit 2 = Devcat check bit 3 = Display in F (Defaults all 0...see set command)
Tx	32 (20)	0		Programming lock On
Tx	32 (20)	1		Programming lock off
Tx	32 (20)	2		Beep On
Tx	32 (20)	3		Beep Off

Tx	32 (20)	4		Display C
Tx	32 (20)	5		Display F
Tx	32 (20)	6		Stayawake On (4 minute duration)
Tx	32 (20)	7		Stayawake Off
Tx	32 (20)	8		Enable - Check Dev Cat for Linking. Only devices with Dev Cat 09 can be linked.
Tx	32 (20)	9		Disable - Check Dev Cat for Linking. Any device can be linked.
Tx	46 (2E)	0	Group/Button # 0-FF	Data 2: 0 is for Read Data (Rx unit will send back a Data 2: 1 below)
				Data 2: 1 return of data Data 3: Main Display Device (Device in group 1, 2, 3) Data 4: Display Mode (01 is "Now Watts", 02 is "Avg \$/mo", 03 is "Avg kWh/mo") Data 5: Temp Format (00 = C, 01 = F) Data 6: Cost Hi Byte (cents/kWh) Data 7: Cost Low Byte Data 8: Temp Calibration Signed Hi Byte Data9: Temp Calibration Signed Low Byte Data10: Current Temp Hi Byte (degree C) Data 11: Current temp Low Byte Data 12: Battery Life (4 = Full, 3, 2, 1 = Empty) Data 13: DB Delta Data 14: Application chip FW version
Rx	46 (2E)	0	Group/Button # 0-FF	
Tx	46 (2E)	0	Group/Button # 0-FF	Data 2: 2 is for setting Main Display Device Data 3: 1 or 2 or 3 (Device in group 1, 2, 3)
Tx	46 (2E)	0	Group/Button # 0-FF	Data 2: 3 is for setting Display Mode (Now Watts, Avg \$/mo, Avg kWh/mo) Data 3: 1 is "Now Watts", 2 is "Avg \$/mo", 3 is "Avg kWh/mo"
Tx	46 (2E)	0	Group/Button # 0-FF	Data 2: 4 is for setting Temp Format Data 3: 0 = C and 1 = F
Tx	46 (2E)	0	Group/Button # 0-FF	Data 2: 5 is for setting cost (cents/kWh) Data 3: Hi Byte Data 4: Lo Byte. Integer 1 = .1 to 9990 = 999 (2706Hex = 9990 = 999)
Tx	46 (2E)	0	Group/Button # 0-FF	Data 2: 6 is for Temp Calibration Data 3: Hi Byte Data 4: Lo Byte. Signed Integer -199 to 199 (FFFF = -1)
Tx	46 (2E)	0	Group/Button # 0-FF	Data 2: 7 is for reset start point of average cost calculation Data 3: Bit Flag for which group (001 = group 1, 010 = group 2, 100 = group 3, 111 = all groups, etc.)
Tx	46 (2E)	1	Group/Button 1-3 (0-	Data 2: 0 is for Read Data (Rx unit will send back a Data 2: 1 below) for the device

			FF valid)	group selected in Data1
Rx			-	Data 2: 1 return of data Data 3: Now Watts (Main Power) Hi Byte Data 4: Now Watts (Main Power) Lo Byte Data 5: Avg \$/mo (4 bytes) Data 6: Avg \$/mo Data 7: Avg \$/mo Data 8: Avg \$/mo Data9: Avg KWH(4 bytes) Data10: Avg kWh/mo Data11: Avg kWh/mo Data12: Avg kWh/mo Data13: Avg kWh/mo
Tx	47 (2F)	1	Group/Button # 0-FF	Data 2: 0 is for Read Data (Rx unit will send back a Data 2: 1 below)
Rx	47 (2F)	1	Group/Button # 0-FF	Data 2: 1 Return of data. Read device ID for each group. Data3: id1MSb Hi Byte (Device ID in group 1) Data4: id1 Data5: id1 Lo Byte Data6: id2MSB Hi Byte (Device ID in group 2) Data7: id2 Data8: id2 Data9: id3MSB Hi Byte (Device ID in group 3) DataA: id3 DataB: id3
Tx	47 (2F)	1	Group/Button # 01	Data 2: 2 Write an ID to group 1 Data3: idMSb Data4: id Data5: id
Tx	47 (2F)	1	Group/Button # 02	Data 2: 2 Write an ID to group 2 Data3: idMSb Data4: id Data5: id
Tx	47 (2F)	1	Group/Button # 03	Data 2: 2 Write an ID to group 3 Data3: idMSb Data4: id Data5: id

Temperature Reporting:

Link the energy display group 4 to a PLM. Temperature will be checked every 3 minutes. If it changes more than 1 degree C, a broadcast will be sent out with the temperature (Example below).

Energy Display ID: AA BB CC

Broadcast: AA BB CC 00 00 04 8F 6E 40 1A

40 hex → 64 dec, $64 * 0.5 \text{ degree C} = 32 \text{ degree C}$

Sending Energy Data to Display from a PLM:

Write a PLM-Energy Display link in group 1, 2, or 3 using 2F 01 01 AA BB CC (for INSTEON ID "AA.BB.CC", group 1)

Energy Display will send iMeter Get Status command 82 00 to the linked PLM.

Send data from the linked PLM to the Energy Display as in the examples below:

PLM → 0 watts → Energy Display. Send 02 62 AA BB CC 1F 82 00 00 00 00 FA 00 91 00 00 00 00 4C 7D ED 14

PLM → 99 watts → Energy Display. Send 02 62 AA BB CC 1F 82 00 00 94 00 F7 00 91 00 63 00 00 4C 83 50 9F

NOTE: You must calculate the CRC in Data 13 and Data 14.

CRC Calculation

16 bit CRC calculation of payload for checking data involves data bytes from command 1 to Data 12 byte

1.1.1 Calculation

Sample Source Code:

```
unsigned int MyFrame::crc16(unsigned char *msgbuf,int count)
{
    int loop,bit;
    unsigned int fb;
    unsigned char byte;
    unsigned int crc;

    crc = 0;

    for(loop = 0;loop < count;loop++)
    {
        byte = msgbuf[loop];
        for(bit = 0;bit < 8;bit++)
        {
            fb = byte & 1;
            fb = (crc & 0x8000) ? fb ^ 1 : fb;
            fb = (crc & 0x4000) ? fb ^ 1 : fb;
            fb = (crc & 0x1000) ? fb ^ 1 : fb;
            fb = (crc & 0x0008) ? fb ^ 1 : fb;
            crc = (crc << 1) | fb ;
            byte = byte >> 1;
        }
    }
    return crc;
}
```