

How to read a MS-TH using the Link12/45

The MS family of sensors is based on the Dallas DS2438Z Smart Battery Monitor.

Think of it as a 1-wire dual channel A/D converter with temperature sensor and a little bit of EEPROM.

Conventions used in this explanation:

Blue denotes characters sent by the Host to the Link12/45. These characters are not echoed by the Link12/45.

Black denotes characters returned by the Link12/45 to the host

Red characters are annotations. They did not actually pass between the host and Link12/45

<CR> means a single ASCII carriage return character.

<LF> means a single ASCII line feed character.

The steps in acquiring the data from the MS-TH are as follows:

1. Tell it to convert its temperature
2. Tell it to convert its voltage readings
3. Read the information back from the registers

In our example, the address of the DS2438 is:

2608846200000083

We acquired the serial number using the “f” command:

```
Link V1.1x Serial Number: 01ABCDEF
rP<CR><LF> ; A link reset command
f-, 8300000062840826<CR><LF> ; A link first command
```

Step 1: Convert the temperature

The commands to convert the temperature are:

```
rP<CR><LF>          ; A reset command  
b55260884620000008344<CR><LF> ; Link byte command "b"  
                                ; OW Match ROM 0x55  
                                ; Address of the DS2438 260884620000083  
                                ; OW Convert T 0x44
```

Now wait for the temperature conversion to complete before sending additional OW commands (10ms).

Step 2: Convert the A/D voltage

The commands to convert the voltage are:

```
rP<CR><LF> ; A reset command  
b552608846200000083B4<CR><LF> ; Link byte command "b"  
; OW Match ROM 0x55  
; Address of the DS2438 260884620000083  
; OW Convert V 0xB4
```

Now wait for the conversion to complete (10ms) before continuing.

Step 3: Read the converted data from registers

The commands to read the temperature and Vdd voltage are:

```
rP<CR><LF> ; A reset command  
b552608846200000083B800<CR><CR><LF> ; Link byte command "b"  
; OW Match ROM 0x55  
; Address of the DS2438 260884620000083  
; OW Recall Memory page 00 0xB800  
  
rP<CR><LF> ; A reset command  
b552608846200000083BE00FFxxFFyyFFzzFFaaFFbb<CR><CR><LF>  
; Link byte command "b"  
; OW Match ROM 0x55  
; Address of the DS2438 260884620000083  
; OW Recall Memory page 00 0xB800  
; xx is the Status Register of the DS2438  
; yy is the LSB of the Temperature Register  
; zz is the MSB of the Temperature Register  
; aa is the LSB of the Voltage Register  
; bb is the MSB of the Voltage Register  
  
rP<CR><LF> ; A reset command
```

Assuming you can type better than me; this is what you would see on your terminal screen.

```
Link V1.1x Serial Number: 01ABCDEF  
P  
55260884620000008344  
P  
552608846200000083B4  
P  
552608846200000083B800  
P  
552608846200000083BE000090168900  
P
```

The 0x9016 is the contents of the temperature register.
The 0x8900 is the contents of the voltage register.

Step 4: Interpret the data

Ok, the temperature was returned to us as 0x1690 (Remember it came back LSB first so we have to invert the bytes).

In binary this is: 0001 0110 1001 0000

We need to shift this right 3 bits yielding 0000 0010 1101 0010 or 0x02D2. This is 722 in decimal. Multiply this by 0.03125 and we get 22.5625 degrees C (or 72.6125 degrees F). Easy huh?

Now it gets a little tricky.

The voltage from the humidity sensor (actually the A/D) was 0x0089 (remember to invert the bytes).

That is 137 decimal and since each bit is 10mV what we really have is 1.37V

The formula from the Honeywell 3610-001 data sheet is:

$$(V_{out} - 0.958) / 0.0307$$

Substituting our observed voltage we get:

$$(1.37 - 0.958) / 0.0307 = 13\% \text{ relative humidity}$$

You can make your readings a bit more accurate by considering the temperature and supply voltage for the DS2438.

One gottcha. The DS2438 initial condition is to convert the Vdd voltage not the A/D voltage. Bit IAD of the status register must be a zero. Back in our example, remember we retrieved the status register and it had a value of 0x00. That means that the bit had been cleared by us at the factory and your readings will be correct. If you are getting consistently wrong readings and they don't change much ... check and see if that bit has been set erroneously. In the LinkTH I don't trust anyone anymore, so the first thing I do is initialize all the MS-TH's so they convert the A/D.