Kelvin SHT

Back to VSCP Modules

```
Current Harware Version: DIN version 0.0.9, SMD version 0.0.2
Current Firmware version: 0.1.0
```

Download

Everything for the project can be downloaded at the following link
http://www.dosilos.se/download/hardware/kelvin/sht [http://www.dosilos.se/download/hardware/kelvin/sht]

Firmware

Version 0.1.0

- Source
  http://www.dosilos.se/download/hardware/kelvin/sht/kelvin_sht11_firmware_src_010.zip
  [http://www.dosilos.se/download/hardware/kelvin/sht/kelvin_sht11_firmware_src_010.zip]
- Hex
  http://www.dosilos.se/download/hardware/kelvin/sht/kelvin_sht11_firmware_hex_010.zip
  [http://www.dosilos.se/download/hardware/kelvin/sht/kelvin_sht11_firmware_hex_010.zip]

Version 0.0.2

- Source
  http://www.dosilos.se/download/hardware/kelvin/sht/kelvin_sht11_firmware_src_002.zip
  [http://www.dosilos.se/download/hardware/kelvin/sht/kelvin_sht11_firmware_src_002.zip]
- Hex
  http://www.dosilos.se/download/hardware/kelvin/sht/kelvin_sht11_firmware_hex_002.zip
  [http://www.dosilos.se/download/hardware/kelvin/sht/kelvin_sht11_firmware_hex_002.zip]

MDF File 0.0.1

- http://www.eurosource.se/sht_001.xml [http://www.eurosource.se/sht_001.xml]

Schematics

Thrue hole version

- BMP for schema: http://www.dosilos.se/download/hardware/kelvin/sht/kelvin_din_sch009.bmp
  [http://www.dosilos.se/download/hardware/kelvin/sht/kelvin_din_sch009.bmp]
- BMP for Board: http://www.dosilos.se/download/hardware/kelvin/sht/kelvin_din_brd009.bmp
  [http://www.dosilos.se/download/hardware/kelvin/sht/kelvin_din_brd009.bmp]

SMD version

- BMP for schema: http://www.dosilos.se/download/hardware/kelvin/sht/kelvin_smd_sch002.bmp
  [http://www.dosilos.se/download/hardware/kelvin/sht/kelvin_smd_sch002.bmp]
- BMP for Board: http://www.dosilos.se/download/hardware/kelvin/sht/kelvin_smd_brd002.bmp
  [http://www.dosilos.se/download/hardware/kelvin/sht/kelvin_smd_brd002.bmp]

About

Control module

Sensor module

The Kelvin SHT module is an open hardware/software temperature and humidity module for the VSCP (Very Simple Control Protocol) for use over CAN (Controller Area Network). The module is mounted in a DIN norm box with standard one unit width (17.5 mm) or in a small case that can be used on or off a DIN rail.

The module is built around a Microchip PIC18F248/PIC18F2480 and the code is written in C using the Microchip MCC-18 compiler.

The Kelvin SHT can be powered with 9-16V DC and draws about 0.15W.

Kelvin SHT is also available in the form of a bare PCB. Firmware, electronic drawings and PCB layout are freely available and can be freely used in both commercial and non commercial projects.

The module, the PCB and components for it can be bought from eurosource at http://www.eurosource.se [http://www.eurosource.se] In its default configuration the module sends out temperature, humidity and dew-point events once every 30 seconds. The interval as well as the format for the data can be customized. Currently temperature and dew-point can be reported in Kelvin, Celsius and Fahrenheit.

The module can be configured to send out alarm events on high/low temperature/humidity.

The module keeps track of absolute min/max values and also has a running mean for all values.

The temperature data is accurate to +- 0.4 degree at 25 degrees Celsius and is typically +-2 over the sensors entire range. The Humidity data is accurate to +- 3% and reported dew-point typically has a +-1 degree accuracy. A calibration value can be set in the module for both temperature and humidity but is not needed for normal use.

Temperature range is -40 to +123 degrees Celsius.

Installing

The module have screw connections on two sides. One side is for CAN and power (marked "CAN + Power") and the other side is for the sensor (marked "sensor").

Connecting the sensor

DATA	CLOCK
+5V	GND

Attach the

- orange lead to +5V in the lower left.
- orange/white lead to GND in the lower right
- blue lead (DATA) to upper left position.
- blue/white lead (CLOCK) to the upper right position.

Connecting CAN and Power

CANH	CANL
9-16V DC	GND

Attach CANL to the upper right position and CANH to the upper left position. If this is the first or the last module in the chain set the 120 ohm terminator (a standard 120 ohms resistor) between CANL and CANH.

CAT5 twisted pair cable should be used and there should be less than 100 meters between the first and the last node. Both ends of the bus should be terminated with a 120 ohm resistor (coupled between CANL and CANH).

Connect the ground cable from the bus to the ground position in the lower right position.

and Power...

First turn of the power source. You should never connect a module with power on.

Secondly you have to decide if the module should be powered from the bus or from a local power source.

If powered from the bus just connect the power line to the lower left screw terminal. Connect ground to lower right position.

If you have a local DC power source is should be between 9-16V. Connect the positive side to the lower left screw terminal and ground to the lower right screw terminal.

Depending on which modules you have installed you can power about 50 modules from the bus if 16V is used.

Your module is now ready for some real work.

Starting it up

When the module is first powered up the green lamp starts to blink to indicate that the module is uninitialized. Press the red button on the device. The address negotiation now starts. The lamp will stop blinking and light steady when initialized or be turned of if an error occurs It is important to understand as VSCP over CAN requires at least two modules on the bus to be working. If you have a computer CAN interface this is counted as one node. The interface should be open and the CAN bitrate set to 500 kbps.

Registers

```
Register 0(0x00) Zone.
Register 1(0x01) Sunzone.
```

```
Register 2(0x02) Current temperature, MSB. Read only.
Register 3(0x03) Current temperature, LSB. Read only.
```

```
Register 4(0x04) Current humidity, MSB Read only.
Register 5(0x05) Current humidity, LSB Read only.
```

```
Register 6(0x06) Current dew point, MSB. Read only.
Register 7(0x07) Current dew point, LSB. Read only.
```

```
Register 8(0x08) Absolute high temperature, MSB. Read/Write.
Register 9(0x09) Absolute high temperature, LSB. Read/Write.
```

Writing any value to register 8 or 9 will reset there content.

```
Register 10(0x0A) Absolute low temperature, MSB. Read/Write.
Register 11(0x0B) Absolute low temperature, LSB. Read/Write.
```

Writing any value to register 10 or 11 will reset there content.

```
Register 12(0x0C) Absolute high humidity, MSB. Read/Write.
Register 13(0x0D) Absolute high humidity, MSB. Read/Write.
```

Writing any value to register 12 or 13 will reset there content.

```
Register 14(0x0E) Absolute low humidity, MSB. Read/Write.
Register 15(0x0F) Absolute low humidity, MSB. Read/Write.
```

Writing any value to register 14 or 15 will reset there content.

```
Register 16(0x10) Absolute high dew point, MSB. Read/Write.
Register 17(0x11) Absolute high dew point, LSB. Read/Write.
```

Writing any value to register 16 or 17 will reset there content.

```
Register 18(0x12) Absolute low dew point, MSB. Read/Write.
Register 19(0x13) Absolute low dew point, LSB. Read/Write.
```

Writing any value to register 18 or 19 will reset there content.

```
Register 20(0x14) Running mean of temperature, MSB. Read/Write.
Register 21(0x15) Running mean of temperature, LSB. Read/Write.
```

Writing any value to register 20 or 21 will set mean value to zero.

```
Register 22(0x16) Running mean of humidity, MSB. Read/Write.
Register 23(0x17) Running mean of humidity, LSB. Read/Write.
```

Writing any value to register 22 or 23 will set mean value to zero.

```
Register 24(0x18) Running mean of dew point, MSB. Read/Write.
Register 25(x019) Running mean of dew point, LSB. Read/Write.
```

Writing any value to register 24 or 25 will set mean value to zero.

```
Register 26(0x1A) Temperature low alarm point. 255 mean no alarm and is the default value.
This value is read and written degrees Celsius  Read/Write  Note that the value stored in this register
is the real value not the 100* value used for other registers.
```

```
Register 27(0x1B) Temperature high alarm point.  255 mean no alarm and is the default value.
This value is read and written degrees Celsius  Read/Write  Note that the value stored in this register
is the real value not the 100* value used for other registers.
```

```
Register 28(0x1C) Humidity low alarm point.  255 mean no alarm and is the default value.
Read/Write.  Note that the value stored in this register is the real value not the 100* value used for
other registers.
```

```
Register 29(0x1D) Humidity high alarm point.  255 mean no alarm and is the default value.
Read/Write. Note that the value stored in this register is the real value not the 100* value used for
other registers.
```

```
Register 30(0x1E) Temperature sensor read error counter. Read/Write.
A write of any value set the register to zero.
```

```
Register 31(0x1F) Humidity sensor read error counter. Read/Write.
A write of any value set the regsiter to zero.
```

```
Register 32(0x20) Sensor Status register read/write. Read/Write.
```

```
Register 33(0x21) Time in seconds between temperature events.
(0 is no temperature event, 30 is default). Read/Write
```

```
Register 34(0x22) Time in seconds between humidity event.
(0 is no temperature event. Set to 30 by default). Read/Write.
```

```
Register 35(0x23) Time in seconds between dew point event. (
0 is no temperature event. Set to 30 by default). Read/Write.
```

```
Register 36(0x24) Temperature format.
    0x00 – Kelvin
    0x01 Celsius (default)
    0x02 Fahrenheit. Read/Write
```

```
Register 37(0x25) Temperature calibration MSB. (default is zero). Read/Write.
Register 38(0x26) Temperature calibration LSB. (default is zero). Read/Write.
```

```
Register 39(0x27) Humidity calibration MSB (default is zero). Read/Write.
Register 40(0x28) Humidity calibration LSB (default is zero). Read/Write.
```

```
Register 41(0x29) Hysterisis for temperature (default is five degrees Celsius). Read/Write.
```

```
Register 42(0x2A) Hysteresis for humidity (default is five percent). Read/Write.
```

All values, temperature, humidity, dew point are stored as 100 * the real value. This is to preserve two decimals as they are stored as integers. This means that you should divide a read value with 100 to get the real value. This is true for all registers except for the alarm points.

Alarm register

You can read standard VSCP register 128 (0×80) to get alarm status for the device. When an alarm is generated the corresponding bit in the alarm register will be set. The bit will be cleared by a read of the alarm register .You can therefore pick up alarm conditions without looking for alarm events.

This can also be used as a safeguard against missed alarm events.

Bit	Description
Bit 7	Temperature high alarm.
Bit 6	Temperature low alarm.
Bit 5	Humidity high alarm.
Bit 4	Humidity low alarm.
Bit 3	Reserved.
Bit 2	Reserved.
Bit 1	Reserved.
Bit 0	Reserved.

Events

The following events are generated by this device.

Temperature Event

```
canid = 0x80a06xx where xx is node nickname id.
```

The temperature is reported as a normalized integer value by this event. The event frequency is set in register 33 (0×21) (default is one event per 30 seconds). Set this register to zero to disable the event.

Temperature can be reported in one of three units. Kelvin, Celsius (default) or Fahrenheit as set in register 36 (0×24).

```
Class: 0x00A
Type: 0x06
```

Format:

See below.

Package:

Byte	Description
Byte 0	format
Byte 1	0×02 (exponent)
Byte 2	MSB of normalized integer
Byte 3	LSB of normalized integer

format is the measurement data coding format described in the VSCP specification

Bits	Description
5,6,7	0×08 - Normalized integer format.
3,4	0×00 - Kelvin 0×01 – Celsius 0×02 – Fahrenheit.
0,1,2	Sendor Index: 0×00 or 0×01 for sensor 1 or sensor 2.

Package examples

+40.00 degrees Celsius is reported as:

Byte	Description
Byte 0	0×88
Byte 1	0×02
Byte 2	0x0F
Byte 3	0xA0

+120.00 degrees Celsius is reported as:

Byte	Description
Byte 0	0×88
Byte 1	0×02
Byte 2	0x2E
Byte 3	0xE0

0.00 degrees Celsius is reported as:

Byte	Description
Byte 0	0×88
Byte 1	0×02
Byte 2	0×00
Byte 3	0×00

-40.00 degrees Celsius is reported as:

Byte	Description
Byte 0	0×88
Byte 1	0×02
Byte 2	0xF0

Byte 3 0×60

1.38 degrees Celsius is reported as:

Byte	Description
Byte 0	0×88
Byte 1	0×02
Byte 2	0×00
Byte 3	0x8A

Humidity

```
canid = 0x80a23xx where xx is node nickname id.
```

The humidity is reported as a floating point value by this event. The event frequency is set in register 34 (0×22) (default one event per 30 seconds). Set this register to zero to disable the event.

Humidity is always reported in percent (0-100%).

For most nodes which are low level devices floating point is not the preferred data format. This is not a problem as the data can be read in byte 2 (MSB) and byte (3(LSB) as a value 100 times the measured value. The sign for this value can be fetched from bit 7 of byte 2.

```
Class 0x00A
Type 0x23
```

Format:

See below.

Package:

Byte	Description
Byte 0	0×80
Byte 1	0×02 (exponent)
Byte 2	MSB of normalized integer
Byte 3	LSB of normalized integer

format is the measurement data coding format described in the VSCP specification

Bits	Description
Bits 5,6,7 0×08	Normalized integer format.
Bits 3,4 0×00	Realative humidity.
Bits 1,2,3	0×00 or 0×01 for sensor 1 or sensor 2.

Package examples

47.54% humidity is reported as:

Byte	Description
Byte 0	0×80
Byte 1	0×02
Byte 2	0×12
Byte 3	0×92

Dew point

```
canid = 0x80a31xx where xx is node nickname id.
```

The calculated dew point is reported as a normalized integer value by this event. The event frequency is set in register 35 (0×23) (default one event per 30 seconds). Set this register to zero to disable the event. Dew point can be reported in one of three units. Kelvin, Celsius (default) and Fahrenheit as set in register 36

(0×24).

```
Class 0x00A
Type 0x31
```

Format:

See below.

Package:

Byte	Description
Byte 0	format
Byte 1	0×02 (exponent)
Byte 2	MSB of normalized integer
Byte 3	LSB of normalized integer

format is the measurement data coding format described in the VSCP specification

Bits	Description
Bits 5,6,7	0×08 - Normalized integer format.
Bits 3,4	0×00 - Kelvin 0×01 – Celsius 0×02 – Fahrenheit.
Bits 1,2,3	0×00 or 0×01 for sensor 1 or sensor 2.

Alarm

```
canid = 0xe0102xx where xx is node nickname id.
```

The module can send alarm events on a number of conditions:

- Temperature over limit (code = 1).
- Temperature under limit (code = 2).
- Humidity over limit (code = 3).
- Humidity under limit (code = 4).

Code = 0 is reserved for a general alarm and is not used at the moment.

One event will be sent for each alarm type generated. When the corresponding alarm flag is set in the alarm register set after an alarm no new alarms will be generated until the bit is cleared and a new alarm condition occurs.

```
Class 0x001
Type 0x02
```

Format:

Package:

Byte	Description
Byte 0	code for alarm type.
Byte 1	Alarm register.

Error

```
canid = 0xe140dxx where xx is node nickname id.
```

An error message is sent out if a failure is detected with the module or the SHT-xx sensor.

- Error code = 0 – Reserved
- Error code = 1 – Sensor fault.
- Error code = 2 – Module fault.

```
Class: 0x014
Type: 0x0d
```

Format:

Package:

Byte	Description
Byte 0	error code.

Contact Information

This is open hardware / software that is made by Ake Hedman, eurosource, Sweden. It is free to use for commecial or non commercial use.

Email: vscp@eurosource.se [mailto:vscp@eurosource.se]

Company:

```
eurosource
Brattbergavagen 17
820 50 LOS
SWEDEN

info@eurosource.se
http://www.eurosource.se

Phone: +46 657 413430
Fax: +46 657 413503
```

Modules and components are sold at http://www.eurosource.se/catalog [http://www.eurosource.se/catalog]

More info about the VSCP and CANAL protocol can be found at http://www.vscp.org [http://www.vscp.org]

module, can, climate

modules/kelvin_sht.txt · Last modified: 2011/01/03 08:44 by admin