

Kelvin Smart II - dual temperature module

[Back to VSCP Modules](#)

Last Updated: 2007-02-27

Current Hardware Version: See Kelvin SHT (Same hardware is used).
Current Firmware version: 0.1.5

Download

Got to <http://www.grodansparadis.com> [<http://www.grodansparadis.com>] for up to date info 

Firmware

MDF File 0.0.1

- http://www.eurosource.se/smart2_001.xml [http://www.eurosource.se/smart2_001.xml]

Version 0.1.5

- **Source:** Available in svn <http://www.vscp.org/download> [<http://www.vscp.org/download>] in subfolder /firmware/pic/kelvin/smart2/project
A nightly snapshot is available at <http://www.dosilos.se/download/> [<http://www.dosilos.se/download/>]
- **Hex files:** <http://www.dosilos.se/download/hardware/kelvin/smart>
[<http://www.dosilos.se/download/hardware/kelvin/smart>]

!!:Use **save as...** to save the hex files.

The **debug** version is built for use with the ICD2, the **release, reloc** version is built to leave room for a bootloader and the **reloc** version is a standard image without a bootloader.

Version 0.0.2

- Source http://www.dosilos.se/download/hardware/kelvin/smart/kelvin_smart2_firmware_src_002.zip
[http://www.dosilos.se/download/hardware/kelvin/smart/kelvin_smart2_firmware_src_002.zip]
- Hex http://www.dosilos.se/download/hardware/kelvin/smart/kelvin_smart2_firmware_hex_002.zip
[http://www.dosilos.se/download/hardware/kelvin/smart/kelvin_smart2_firmware_hex_002.zip]

Schematics

The Kelvin Smart II module use the same hardware as the Kelvin SHT module so the schematics are the same. Please see http://www.vscp.org/wiki/doku.php?id=kelvin_sht_-_temperature_and_humidity_module [http://www.vscp.org/wiki/doku.php?id=kelvin_sht_-_temperature_and_humidity_module]

About



The Kelvin Smart II module is an open hardware and software temperature times two 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) Two sensors can be connected to the module using cables with up to 20 meters length.

The module is built around a Microchip PIC 18F258/PIC18F2580 and the open source code is written in C using the Microchip MCC-18 compiler.

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

Kelvin Smart II 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 at <http://www.eurosource.se> [<http://www.eurosource.se>] In its default configuration the module sends out temperature events for each of its sensors once every 30 seconds. The interval as well as the format for the data can be customized. Currently temperature can be reported in Kelvin, Celsius and Fahrenheit.

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

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

The temperature data is accurate to ± 0.7 degree at 25 degrees Celsius and is typically ± 1.2 over the sensors entire range. A calibration value can be set in the module for both temperature and humidity but is not needed for normal use.

Working temperature range for the module is -45 to +130 degrees Celsius.

Ready built modules and sensors can be bought here <http://www.eurosource.se/catalog> [<http://www.eurosource.se/catalog>]

Description

The Kelvin Smart II module is an open hardware and software temperature 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) The module is built around a Microchip PIC18F258/PIC18F2580 and the code is written in C using the Microchip MCC-18 compiler.

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

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

Sensor 1	Sensor 2
+5V	GND

Attach the

- Power lead of sensor to +5V in the lower left.

- GND lead of sensor to GND in the lower right
- Sensor 1 to upper left screw terminal.
- Sensor 2 to upper right screw terminal.

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 500 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 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 it 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 off 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 125 kbps.

Registers

Register 0(0x00) - Zone
Register 1(0x01) - Subzone
Register 2(0x02) - Current temperature sensor 1, MSB. Read only
Register 3(0x03) - Current temperature sensor 1, LSB. Read only
Register 4(0x04) - Current temperature sensor 2, MSB Read only
Register 5(0x05) - Current temperature sensor 2, LSB Read only
Register 6(0x06) - Absolute high temperature sensor 1, MSB. Read/Write
Register 7(0x07) - Absolute high temperature sensor 1, LSB. Read/Write

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

Register 8(0x08) - Absolute high temperature sensor 2, MSB. Read/Write
Register 9(0x09) - Absolute high temperature sensor 2, LSB. Read/Write

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

Register 10(0x0A) - Absolute low temperature sensor 1, MSB. Read/Write
Register 11(0x0B) - Absolute low temperature sensor 1, LSB. Read/Write

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

Register 12(0x0C) - Absolute low temperature sensor 2, MSB. Read/Write
 Register 13(0x0D) - Absolute low temperature sensor 2, LSB. Read/Write

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

Register 14(0x0E) - Running mean of temperature sensor 1, MSB. Read/Write
 Register 15(0x0F) - Running mean of temperature sensor 1, LSB. Read/Write

Writing any value to register 14 or 15 will set mean value to zero.

Register 16(0x10) - Running mean of temperature sensor 2, MSB. Read/Write
 Register 17(0x11) - Running mean of temperature sensor 2, LSB. Read/Write

Writing any value to register 16 or 17 will set mean value to zero.

Register 18(0x12) - Temperature low alarm point for sensor 1. 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 19(0x13) - Temperature low alarm point for sensor 2. 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 20(0x14) - Temperature high alarm point for sensor 1. 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 21(0x15) - Temperature high alarm point for sensor 2. 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 22(0x16) - Temperature sensor 1 read error counter. Read/Write

A write of any value set the register to zero.

Register 23(0x17) - Temperature sensor 2 read error counter. Read/Write

A write of any value set the register to zero.

Register 24(0x18) - Sensor Status register read/write. Read/Write

Register 25(0x19) - Time in seconds between temperature events for sensor 1.
 (0 is no temperature event, 30 is default). Read/Write

Register 26(0x1A) - Time in seconds between temperature events for sensor 2.
 (0 is no temperature event, 30 is default). Read/Write

Register 27(0x1B) - Temperature format. 0x00 - Kelvin, 0x01 Celsius (default), 0x02 Fahrenheit. Read/Write

Register 28(0x1C) - Temperature calibration sensor 1 MSB. (default is zero). Read/Write
 Register 29(0x1D) - Temperature calibration sensor 1 LSB. (default is zero). Read/Write

Register 30(0x1E) - Temperature calibration sensor 2 MSB. (default is zero). Read/Write
 Register 31(0x1F) - Temperature calibration sensor 2 LSB. (default is zero). Read/Write

Register 32(0x20) - Hysteresis for temperature sensor 1 (default is five degrees Celsius). Read/Write
 Register 33(0x21) - Hysteresis for temperature sensor 2 (default is five degrees Celsius). 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 (0x80) 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.

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

Events

The following events are generated by this device.

Temperature

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

The temperature is reported as a normalized integer value by this event. The event frequency for sensor 1 is set in register 25 (0x19) and for sensor 2 in register 26 (0x1A) (default is one event per 30 seconds). Set register to zero to disable the event for one or both sensors.

Temperature can be reported in one of three units. Kelvin, Celsius (default) and Fahrenheit as set in register 36 (0x24).

For most other 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 4 (MSB) and byte 5 (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: 0x06

Package:

Byte	Description
Byte 0	format.
Byte 1	0x02 (exponent).
Byte 2	LSB of normalized integer.
Byte 3	MSB of normalized integer.

Format:

See below.

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

Bits	Description
Bits 5,6,7	0x04 Normalized integer format.
Bits 3,4	0x00 == Kelvin Celsius == 0x01 Fahrenheit == 0x02
Bits 0,1,2	Index for sensor. 0x00 or 0x01 for sensor 1 or sensor 2.

Package examples

+40.00 degrees Celsius from sensor 1 is reported as:

Byte	Description
Byte 0	0x88

Byte 1	0x02
Byte 2	0x0F
Byte 3	0xA0

+120.00 degrees Celsius for sensor 2 is reported as:

Byte	Description
Byte 0	0x89
Byte 1	0x02
Byte 2	0x2E
Byte 3	0xE0

0.00 degrees Celsius for sensor 2 is reported as:

Byte	Description
Byte 0	0x89
Byte 1	0x02
Byte 2	0x00
Byte 3	0x00

-40 degrees Celsius for sensor 1 is reported as:

Byte	Description
Byte 0	0x88
Byte 1	0x02
Byte 2	0xFE
Byte 3	0x70

1.38 degrees Celsius for sensor 1 is reported as:

Byte	Description
Byte 0	0x88
Byte 1	0x02
Byte 2	0x00
Byte 3	0x8A

Output when no sensors are connected or sensor failure

If no sensors is connected the following data is received from the module

```
89 82 80 00
```

this value will never show in any other case (represents minus 128 degrees).

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).

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:

none

Package:

Byte	Description
Byte 0	code for alarm type.
Byte 1	Alarm register.
Byte 2	Sensor 1 = 0 Sensor 2 = 1.

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 sensor.

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

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

Format:

none

Package:

Byte	Description
Byte 0	error code.
Byte 1	Zone.
Byte 2	Subzone.

module, can, climate

modules/kelvin_smart_ii.txt · Last modified: 2011/12/01 14:28 by admin