

Paris - Smart Relay Control

[Back to VSCP Modules](#)

Project Status

Everything for the project can be downloaded at the following link

<http://www.dosilos.se/download/hardware/beijing> [<http://www.dosilos.se/download/hardware/beijing>]

Firmware

- Version 0.0.1: http://www.dosilos.se/download/firmware/paris/paris_firmware001.zip
[http://www.dosilos.se/download/firmware/paris/paris_firmware001.zip]

MDF

- Version 0.0.1: http://www.eurosource.se/paris_001.xml [http://www.eurosource.se/paris_001.xml]

Schematics

Old files are available here <http://www.eurosource.se/download/hardware/paris>
[<http://www.eurosource.se/download/hardware/paris>]

Through hole version

- JPEG for schema: http://www.dosilos.se/download/hardware/paris/paris_sch005.jpg
[http://www.dosilos.se/download/hardware/paris/paris_sch005.jpg]
- JPEG for Board: http://www.dosilos.se/download/hardware/paris/paris_brd005.jpg
[http://www.dosilos.se/download/hardware/paris/paris_brd005.jpg]

SMD version

- JPEG for schema: http://www.dosilos.se/download/hardware/paris/paris_smd_sch002.jpg
[http://www.dosilos.se/download/hardware/paris/paris_smd_sch002.jpg]
- JPEG for Board: http://www.dosilos.se/download/hardware/paris/paris_smd_brd002.jpg
[http://www.dosilos.se/download/hardware/paris/paris_smd_brd002.jpg]

About



The Smart Relay Control module is an open hardware and software intelligent relay control module (1-7 relays) 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 (17.5/32/52.5 mm).

The module is built around a Microchip PIC 18F2580 and the code is written in C using the Microchip MCC-18 compiler.

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

Smart Relay Control 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>]

Other modules at <http://www.vscp.org/vscp/modules/index.html> [<http://www.vscp.org/vscp/modules/index.html>]

Documentation & Data

Power

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

Enclosure

Material for the box: Polycarbonate (UL 94-V1)

Colour

Grey

Enclosure class

IP20

Dimensions for module

B17,7×H89×D59 mm

Smart Relay Control is also available in the form of a bare PCB for home builders.

The module, the PCB and components for it can be bought at <http://www.eurosource.se> [<http://www.eurosource.se>]

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 module to relays

The module comes in different versions with different width for control of 3 or 7 relays.

For a 3 relay module the outputs are arranged as

Relay 1	Relay 2
Common GND	Relay 3

For a 7 relay module the outputs are arranged as

Relay 1	Relay 2	Relay 4	Relay 6
Common GND	Relay 3	Relay 5	Relay 7

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/2A is used.

Your module is now ready for some real work.

For the seven relay variant power/CAN is on the leftmost screw terminals.

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 initialization 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 that 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

Reserved Registers

```
Register 0 (0x00) - Zone.
Register 1 (0x01) - SubZone.
```

Relay Status Registers

```
Register 2 (0x02) - Relay 1 Status register. Read/Write
Register 3 (0x03) - Relay 2 Status register. Read/Write
Register 4 (0x04) - Relay 3 Status register. Read/Write
Register 5 (0x05) - Relay 4 Status register. Read/Write
Register 6 (0x06) - Relay 5 Status register. Read/Write
Register 7 (0x07) - Relay 6 Status register. Read/Write
Register 8 (0x08) - Relay 7 Status register. Read/Write
Register 9 (0x09) - Relay 8 Status register. Read/Write
```

Writing a value to the relay control register will activate/deactivate the relay output.

- 0 - The relay is inactivated.
- 1 - The relay is activated.
- 2 - The relay is pulse activated/deactivated (opposite from rest state).

Reading a value from the relay control register is read as a one if the relay is activated and a 0 if the relay is deactivated.

When the pulse activation/deactivating is used the actual relay state is read. That is, if a 2 is written to relay 1 a one will be read while the relay is activated, after which a zero will be read.

Relay Control Registers

```
Register 10 (0x0A) - Relay 1 Control Register. Read/Write
Register 11 (0x0B) - Relay 2 Control Register. Read/Write
Register 12 (0x0C) - Relay 3 Control Register. Read/Write
Register 13 (0x0D) - Relay 4 Control Register. Read/Write
Register 14 (0x0E) - Relay 5 Control Register. Read/Write
Register 15 (0x0F) - Relay 6 Control Register. Read/Write
Register 16 (0x10) - Relay 7 Control Register. Read/Write
Register 17 (0x11) - Relay 8 Control Register. Read/Write
```

The relay control bits enable/disable intelligent relay functionality:

- Bit 0 - Reserved
- Bit 1 - Alarm sent when protection timer triggers if set.

- Bit 2 - Protection timer enable.
- Bit 3 - Send On event when relay goes to active state.
- Bit 4 - Send Off event when relay goes to inactive state.
- Bit 5 - Send Started event when relay goes to active state.
- Bit 6 - Send Stopped event when relay goes to active state.
- Bit 7 - If set to one the relay is enabled.

Relay Pulse time Registers

```

Register 18(0x12) - On/off pulse time Relay 1 (seconds) MSB. Read/Write
Register 19(0x13) - On/off pulse time Relay 1 (seconds) LSB. Read/Write
Register 20(0x14) - On/off pulse time Relay 2 (seconds) MSB. Read/Write
Register 21(0x15) - On/off pulse time Relay 2 (seconds) LSB. Read/Write
Register 22(0x16) - On/off pulse time Relay 3 (seconds) MSB. Read/Write
Register 23(0x17) - On/off pulse time Relay 3 (seconds) LSB. Read/Write
Register 24(0x18) - On/off pulse time Relay 4 (seconds) MSB. Read/Write
Register 25(0x19) - On/off pulse time Relay 4 (seconds) LSB. Read/Write
Register 26(0x1A) - On/off pulse time Relay 5 (seconds) MSB. Read/Write
Register 27(0x1B) - On/off pulse time Relay 5 (seconds) LSB. Read/Write
Register 28(0x1C) - On/off pulse time Relay 6 (seconds) MSB. Read/Write
Register 29(0x1D) - On/off pulse time Relay 6 (seconds) LSB. Read/Write
Register 30(0x1E) - On/off pulse time Relay 7 (seconds) MSB. Read/Write
Register 31(0x1F) - On/off pulse time Relay 7 (seconds) LSB. Read/Write
Register 32(0x20) - On/off pulse time Relay 8 (seconds) MSB. Read/Write
Register 33(0x21) - On/off pulse time Relay 8 (seconds) LSB. Read/Write

```

This is the pulse time for the each relay expressed in seconds. This can be used to have a relay turn on and off with a certain preset interval. The min pulse time is 1 second and the max time is 65535 seconds which is about 18 hours. Set to zero (default) for no pulse time i.e. the relay will be steady on/off.

To start a pulse sequence first write the pulse time to this register and then write 2 to the relay status register to start the output. The pulse train is terminated by writing on or off (1 or 0) to the relay status register.

Relay Protection time Registers

```

Register 34(0x22) - Protection time Relay 1 (seconds) MSB. Read/Write
Register 35(0x23) - Protection time Relay 1 (seconds) LSB. Read/Write
Register 36(0x24) - Protection time Relay 2 (seconds) MSB. Read/Write
Register 37(0x25) - Protection time Relay 2 (seconds) LSB. Read/Write
Register 38(0x26) - Protection time Relay 3 (seconds) MSB. Read/Write
Register 39(0x27) - Protection time Relay 3 (seconds) LSB. Read/Write
Register 40(0x28) - Protection time Relay 4 (seconds) MSB. Read/Write
Register 41(0x29) - Protection time Relay 4 (seconds) LSB. Read/Write
Register 42(0x2A) - Protection time Relay 5 (seconds) MSB. Read/Write
Register 43(0x2B) - Protection time Relay 5 (seconds) LSB. Read/Write
Register 44(0x2C) - Protection time Relay 6 (seconds) MSB. Read/Write
Register 45(0x2D) - Protection time Relay 6 (seconds) LSB. Read/Write
Register 46(0x2E) - Protection time Relay 7 (seconds) MSB. Read/Write
Register 47(0x2F) - Protection time Relay 7 (seconds) LSB. Read/Write
Register 48(0x30) - Protection time Relay 8 (seconds) MSB. Read/Write
Register 49(0x31) - Protection time Relay 8 (seconds) LSB. Read/Write

```

This is the relay protection time. A relay will be inactivated if not written to before this time has elapsed. Set to zero to disable (default). The max time is 65535 seconds which is about 18 hours.

The registers can be as an example be used as a security feature to ensure that an output is deactivated after a preset time even if the controlling device failed to deactivate the relay.

Relay Zone information

```

Register 50(0x32) - Relay 1 Sub Zone
Register 51(0x33) - Relay 2 Sub Zone
Register 52(0x34) - Relay 3 Sub Zone
Register 53(0x35) - Relay 4 Sub Zone
Register 54(0x36) - Relay 5 Sub Zone
Register 55(0x37) - Relay 6 Sub Zone
Register 56(0x38) - Relay 7 Sub Zone
Register 57(0x39) - Relay 8 Sub Zone

```

This is a subzone value related to a specific relay. If zero the the module Zone/subzone will be used else the modules zone and the relays sub zone will be used.

Registers for Decision Matrix

```

Register 72(0x48) - 127(0x7f) - Decision Matrix with seven rows.

```

Decision Matrix

The module has a decision matrix with seven entries.

Available Actions

0 (0x00)	- NOOP, No action.
1 (0x01)	- Activate relay(s) given by argument. The argument is a bit array where bit 0 is relay 1 and so on. Byte 1 is Zone and byte 2 is zone page and must be equal to register content to trigger action.
2 (0x02)	- Deactivate relay(s) given by argument. The argument is a bit array where bit 0 is relay 1 and so on. Byte 1 is Zone and byte 2 is zone page and must be equal to register content to trigger action.
3 (0x03)	- Pulse relay(s) given by argument. The argument is a bit array where bit 0 is relay 1 and so on. Byte 1 is Zone and byte 2 is zone page and must be equal to register content to trigger action.
4 (0x04)	- Reserved.
5 (0x05)	- Reserved.
6 (0x06)	- Reserved.
7 (0x07)	- Reserved.
8 (0x08)	- Reserved.
9 (0x09)	- Reserved.
10 (0x0a)	- Send relay status. The argument is a bit array where bit 0 is relay 1 and so on.
11 (0x0b)	- Reserved.
12 (0x0c)	- Reserved.
13 (0x0d)	- Reserved.
14 (0x0e)	- Reserved.
15 (0x0f)	- Reserved.
16 (0x10)	- Disable relay(s) given by argument. The argument is a bit array where bit 0 is relay 1 and so on. Byte 1 is Zone and byte 2 is zone page and must be equal to register content to trigger action.

Alarm register

Bit	Description
Bit 0	Relay 1 protection timer has caused a relay action.
Bit 1	Relay 2 protection timer has caused a relay action.
Bit 2	Relay 3 protection timer has caused a relay action.
Bit 3	Relay 4 protection timer has caused a relay action.
Bit 4	Relay 5 protection timer has caused a relay action.
Bit 5	Relay 6 protection timer has caused a relay action.
Bit 6	Relay 7 protection timer has caused a relay action.
Bit 7	Relay 8 protection timer has caused a relay action.

Read the register to clear alarm bits.

Events

On

If enabled the event is sent when a relay goes to its active state.

```
Class: 0x014
Type: 0x03
```

Package:

```
Byte 0: Index.
Byte 1: Zone
Byte 2: Subzone
```

Index is 0 for relay 1, 1 for relay 2 and so on. Zon and subzone set accordingly. Subzone for relay is used if it's not zero.

Off

If enabled the event is sent when a relay goes to its inactive state.

```
Class: 0x014
Type: 0x04
```

Package:

```
Byte 0: index.
Byte 1: Zone
Byte 2: Subzone
```

Index is 0 for relay 1, 1 for relay 2 and so on. Zon and subzone set accordingly. Subzone for relay is used if it's not zero.

Stopped

If enabled the event is sent when a relay goes to its inactive state.

```
Class: 0x014
Type: 0x18
```

Package:

```
Byte 0: index.
Byte 1: Zone
Byte 2: Subzone
```

Index is 0 for relay 1, 1 for relay 2 and so on. Zon and subzone set accordingly. Subzone for relay is used if it's not zero.

Started

If enabled the event is sent when a relay goes to its active state.

```
Class: 0x014
Type: 0x19
```

Package:

```
Byte 0: index.
Byte 1: Zone
Byte 2: Subzone
```

Index is 0 for relay 1, 1 for relay 2 and so on. Zon and subzone set accordingly. Subzone for relay is used if it's not zero.

Alarm

If enabled the event is sent when a relay goes to its inactive state after a protection timer have timed out.

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

Package:

```
Byte 0: index.
Byte 1: Zone
Byte 2: Subzone
```

Index is 0 for relay 1, 1 for relay 2 and so on.

VSCP Smart Relay - Simple Test program



This is a simple VB test application for the smart relay module. On this screen dump two decision matrix elements has been set. One that activate relay eight when an ON event is received and another that deactivate the relay when an OFF event is received. Note that Zone and subzone also has to match for the ON/OFF events for this to happen.

Note that there are some random data in the other decision matrix positions but as long as the enable flag (bit 7 in flags) is not set this does not matter.

Contact Information

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

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

Company:

```
eurosource
Brattbergavagen 17
820 50 IOS
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](#), [relay](#)

modules/paris.txt · Last modified: 2011/12/05 16:35 by admin