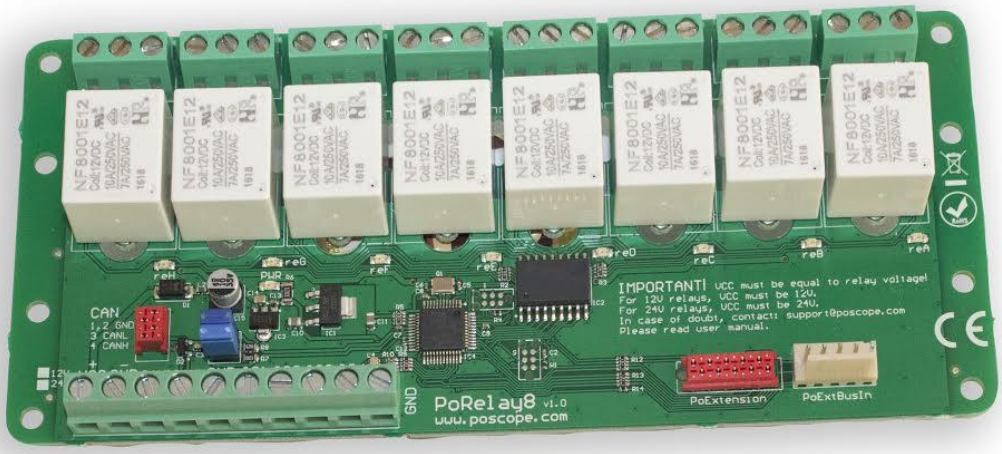
PoLabs

PoRelay8

User's manual



Please read the following notes

1. All information included in this document is current as of the date this document is issued. Such information, however, is subject to change without any prior notice.
2. PoLabs does not assume any liability for infringement of patents, copyrights, or other intellectual property rights of third parties by or arising from the use of PoLabs products or technical information described in this document. No license, express, implied or otherwise, is granted hereby under any patents, copyrights or other intellectual property rights of PoLabs or others. PoLabs claims the copyright of, and retains the rights to, all material (software, documents, etc.) contained in this release. You may copy and distribute the entire release in its original state, but must not copy individual items within the release other than for backup purposes.
3. Descriptions of circuits, software and other related information in this document are provided only to illustrate the operation of the products and application examples. You are fully responsible for the incorporation of these circuits, software, and information in the design of your equipment. PoLabs assumes no responsibility for any losses incurred by you or third parties arising from the use of these circuits, software, or information.
4. PoLabs has used reasonable care in preparing the information included in this document, but PoLabs does not warrant that such information is error free. PoLabs assumes no liability whatsoever for any damages incurred by you resulting from errors in or omissions from the information included herein.
5. PoLabs devices may be used in equipment that does not impose a threat to human life in case of the malfunctioning, such as: computer interfaces, office equipment, communications equipment, test and measurement equipment, audio and visual equipment, home electronic appliances, machine tools, personal electronic equipment, and industrial robots.
6. Measures such as fail-safe function and redundant design should be taken to ensure reliability and safety when PoLabs devices are used for or in connection with equipment that requires higher reliability, for example: traffic control systems, anti-disaster systems, anticrime systems, safety equipment, medical equipment not specifically designed for life support, and other similar applications.
7. PoLabs devices shall not be used for or in connection with equipment that requires an extremely high level of reliability and safety, as for example: aircraft systems, aerospace equipment, nuclear reactor control systems, medical equipment or systems for life support (e.g. artificial life support devices or systems), and any other applications or purposes that pose a direct threat to human life.
8. You should use the PoLabs products described in this document within the range specified by PoLabs, especially with respect to the maximum rating, operating supply voltage range and other product characteristics. PoLabs shall have no liability for malfunctions or damages arising out of the use of PoLabs products beyond such specified ranges.
9. Although PoLabs endeavors to improve the quality and reliability of its products, semiconductor products have specific characteristics such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Further, PoLabs products are not subject to radiation resistance design. Please be sure to implement safety measures to guard them against the possibility of physical injury, and injury or damage caused by fire in the event of the failure of a PoLabs product, such as safety design for hardware and software including but not limited to redundancy, fire control and malfunction prevention, appropriate treatment for aging degradation or any other appropriate measures.
10. Usage: the software in this release is for use only with PoLabs products or with data collected using PoLabs products.
11. Fitness for purpose: no two applications are the same, so PoLabs cannot guarantee that its equipment or software is suitable for a given application. It is therefore the user's responsibility to ensure that the product is suitable for the user's application.
12. Viruses: this software was continuously monitored for viruses during production, however the user is responsible for virus checking the software once it is installed.
13. Upgrades: we provide upgrades, free of charge, from our web site at www.poscope.com. We reserve the right to charge for updates or replacements sent out on physical media.
14. Please contact a PoLabs support for details as to environmental matters such as the environmental compatibility of each PoLabs product. Please use PoLabs products in compliance with all applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive. PoLabs assumes no liability for damages or losses occurring as a result of your noncompliance with applicable laws and regulations.
15. Please contact a PoLabs support at support.poscope.com if you have any questions regarding the information contained in this document or PoLabs products, or if you have any other inquiries.
16. The licensee agrees to allow access to this software only to persons who have been informed of and agree to abide by these conditions.
17. Trademarks: Windows is a registered trademark of Microsoft Corporation. PoKeys, PoKeys55, PoKeys56U, PoKeys56E, PoKeys57U, PoKeys57E, PoKeys57CNC, PoRelay8, PoScope, PoLabs and others are internationally registered trademarks.

PoRelay8 user's manual

Description

PoRelay8 is a relay-based 8-channel expansion card for PoKeys devices. It is a PoExtBus Smart device, which features improved robustness against noise on communication lines over existing PoExtBus devices and failsafe configuration options. Each relay channel has common (Vin), NO (normally-open) and NC (normally-closed) connections exposed on the screw terminals.

The device features a 5-pin PoExtBus input connector, PoExtension connector (for future use), 4-pin CAN connector for simplified daisy-chaining of devices and a 10-pin wire terminal for connecting power, CAN signals and additional digital inputs (for future use).

There is one green LED for power supply indication and 8 additional LEDs for relay operation indications. The device can be mounted to a standard DIN rail using the DIN rail mount adapters.

Features

- Up to 10 PoRelay8 boards can be daisy-chained from the single PoExtBus connector
- 8 built-in electromechanical relays with NO, NC and common contacts
- 5 wires board to board ribbon cable to connect to master device or chaining
- Improved reliability over original PoExtBusRe boards
- Daisy-chaining over CAN bus, allowing greater distances between each board in the chain with much improved noise immunity
- Can be mounted to a DIN rail

Electrical specifications

The devices are produced in two versions - with 12 V or 24 V relays. **Note the power supply voltage indicated on the board next to the 10-pin wire terminal. Match the power supply voltage to that value (relay voltage).**

Parameter	Parameter value	
	12 V relays	24 V relays
Power supply voltage	11 - 13 V	23 - 25 V
Power supply current rating	min. 400 mA	
Output current - 250 VAC	max. 7 A	
Output current - 125 VAC	max. 10 A	
Output current - 28 VDC	max. 10 A	

Pinouts of connectors

1. Power and signal connector pinout

The 10 way power and signal connector has the following pinout. Each wire terminal pin is clearly marked on the device itself.

Wire terminal pin	Function
VCC	Positive power supply input - see electrical specifications
GND	Negative (ground) power supply input
CANl	CAN bus L line
CANh	CAN bus H line
GND	CAN bus ground connection (shared with other GND on the PoRelay8)
IO1	General purpose input 1 (for future use)
IO2	General purpose input 2 (for future use)
IO3	General purpose input 3 (for future use)
IO4	General purpose input 4 (for future use)
GND	Ground connection

2. CAN connector pinout

A standard 4-pin Micro-MaTch connector (usually in red color) is used for easy daisy-chaining of PoRelay8 devices. These connectors are IDC-style connectors for flat/ribbon cables that can easily be assembled by the system integrator.

There is a key hole in the PCB for the correct orientation of the connector. Pin 1 is adjacent to the key hole. Note that the CAN bus signals are replicated on the power and signal connector with screw terminals.

Pin number	Function
1	CAN bus ground connection
2	CAN bus ground connection
3	CAN bus L line
4	CAN bus H line

Figure 1: CAN bus connector (red)

3. Relay output connector pinout

The relays are wired as shown below

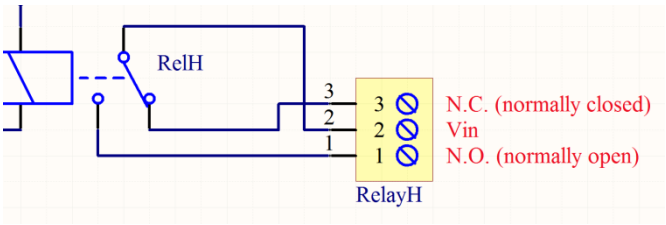


Figure 2: PoRelay8 relay connection schematics

Connect the load between Vin and N.O. or between Vin and N.C.

N.O., means "Normally open" - this contact is closed to Vin if the output is activated. N.C. means "Normally closed" and the contact between N.C. and Vin is opened if the output is activated.

Figure 3: Relay naming - relay outputs are marked with letters A to H in order as indicated

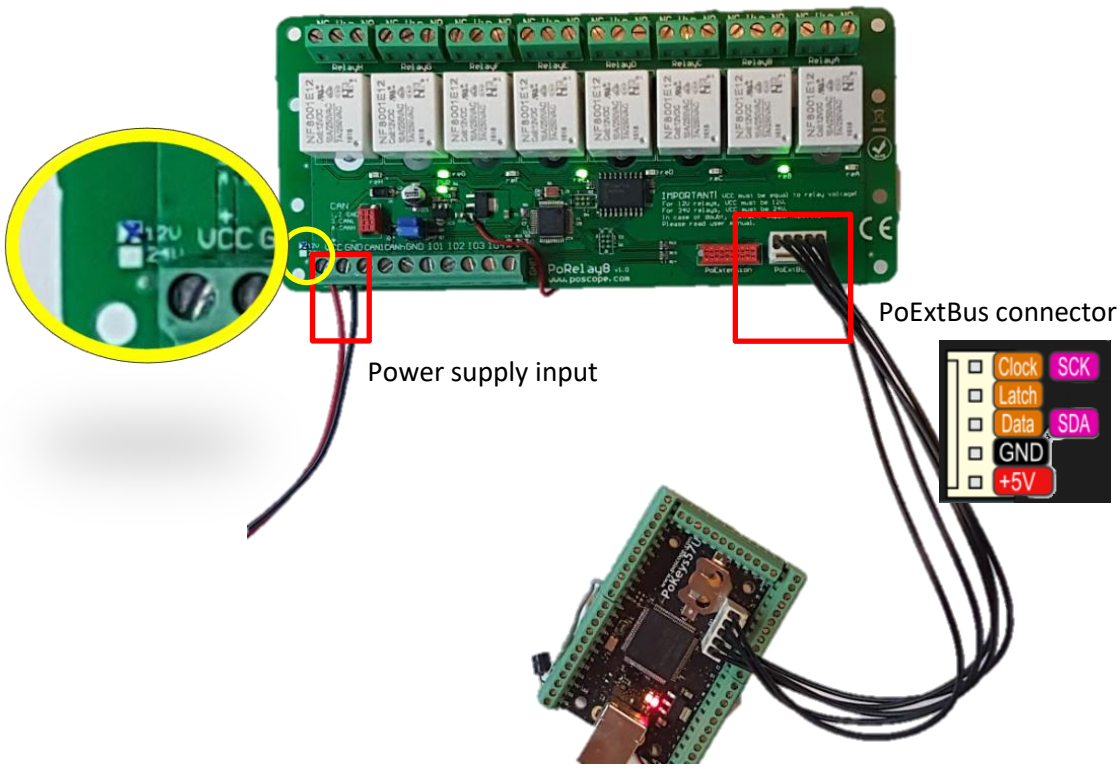
Installation

Installation of PoRelay8 devices is divided into physical device installation and parameter configuration, as the following chapters will describe.

1. Connecting PoRelay8 device to power and PoKeys device

Connect power supply to the wire screw terminal as shown below. Note the relay voltage rating indicated next to the power supply input, as indicated below.

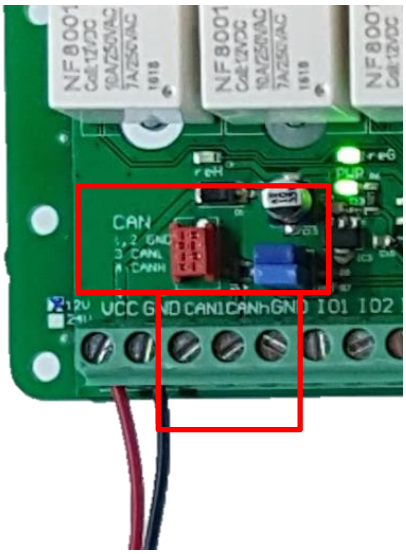
Connect the PoRelay8 to PoKeys device using the provided PoExtBus cable by connecting white 5-pin connector of the PoRelay8 and white 5-pin connector of the PoKeys device. This cable should NOT be extended in length.



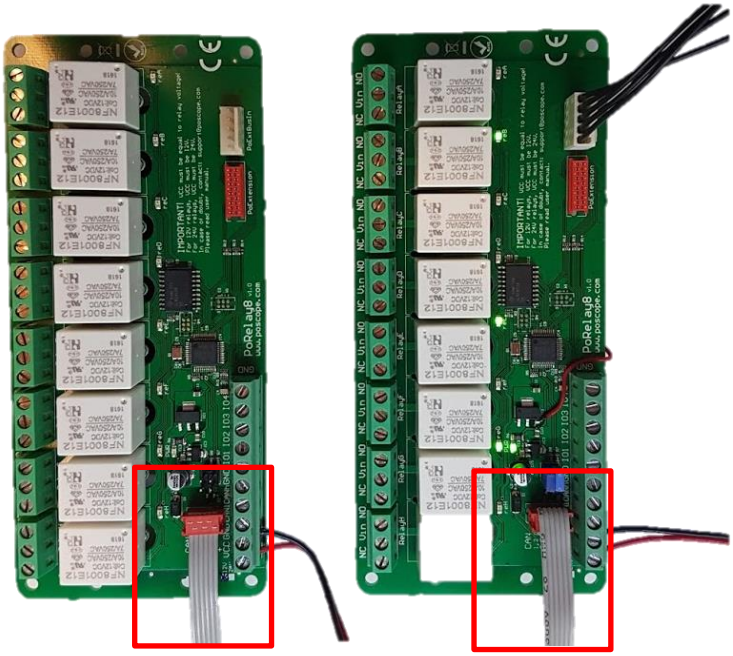
2. Connecting additional PoRelay8 devices

You can connect 9 additional PoRelay8 devices to the original PoRelay8 device, connected to the PoKeys device. Two wiring options are available:

- 4-pin CAN connector
- CAN on main screw terminal



Use either one of these to connect two PoRelay8 devices together. Three signals must be connected in parallel on all PoRelay8 boards: GND, CANl and CANh. The following picture shows another PoRelay8 device connected to the first one via the dedicated CAN connector.



CAN bus termination

CAN bus uses termination resistors at both ends of the communication bus. Therefore, first and the last PoRelay8 device in the chain must be equipped with termination resistors, while those in the middle should have those removed.

PoRelay8 devices already come with the termination resistors built-in. The resistors are selected or disabled using a pair of jumpers next to the red 4-pin CAN bus connector. Install both jumpers as indicated below on first and last PoRelay8 device in the chain and remove the jumpers on all other PoRelay8 devices. **Pay attention to the correct orientation of the jumpers.**

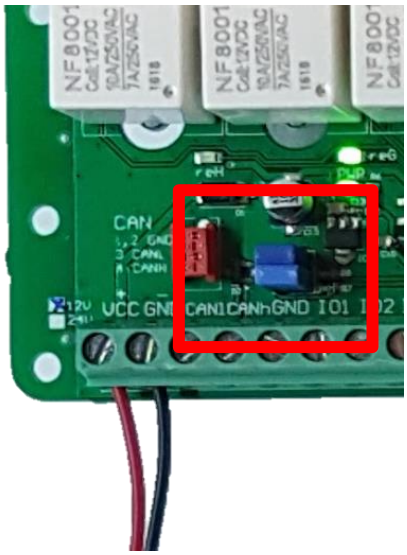


Figure 4: PoRelay8 with CAN termination jumpers installed (termination resistors activated)

3. Configuration of PoRelay8 devices

All PoRelay8 devices will accept PoExtBus device 1 data by default (first PoExtBus device from the PoKeys side, marked as device 1 in PoKeys software, but uses device 9 data in the API). Connect all devices first, then start PoKeys software and open 'Peripherals > PoExtBus Smart' - the following diagram will be displayed.

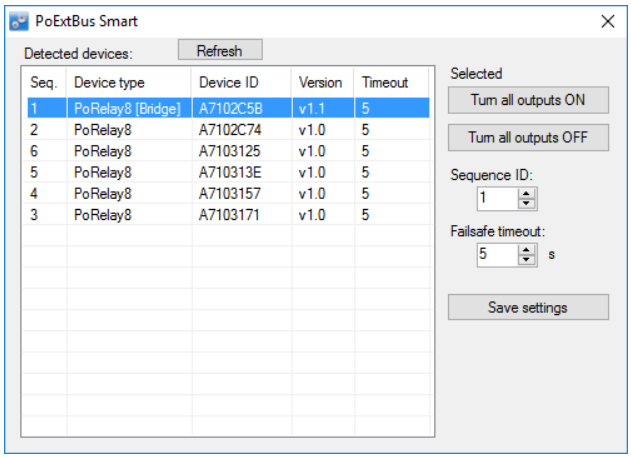
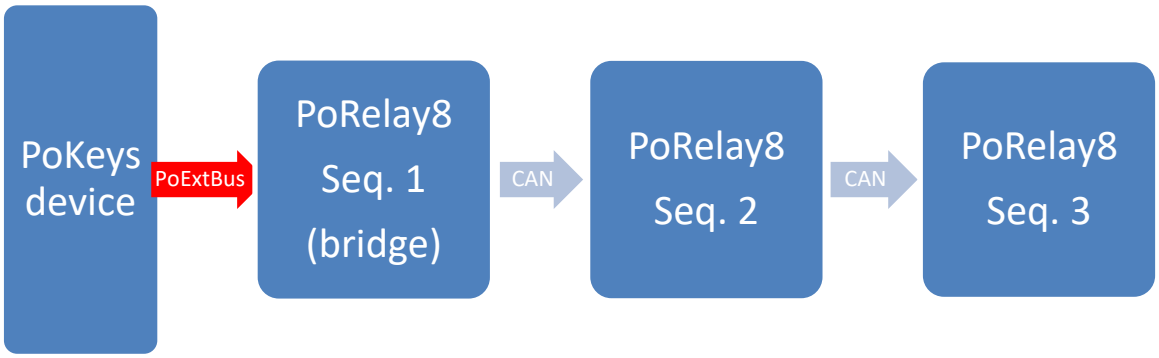


Figure 5: PoExtBus Smart configuration (showing 6 PoRelay8 devices that were detected)

The list will contain all detected PoExtBus Smart devices. Click on the device you want to configure (you can identify the device by clicking 'Turn all outputs ON', which will turn all outputs ON for approximately 1 second). Select the sequence ID for the device by changing the value in the 'Sequence ID'. Use the following diagram as an illustration of how the sequences can be configured.



Note: the PoRelay8 devices allow the user to specify the logical order of the devices that is different to the physical one, based on the target application.

Click 'Save settings' after changing the configuration in order to store the configuration of devices in their non-volatile memory.

Failsafe configuration

By default, PoRelay8 devices will enter the failsafe mode after 5 seconds of no valid incoming data signal (over PoExtBus or CAN interfaces). In failsafe mode, all outputs are deactivated.

The failsafe timeout value can be adjusted between 0 (failsafe disabled) and 60 seconds.

Note: setup failsafe timeout to 0 if using PoRelay8 devices with PoKeys56 series device.

Interoperability with other (non-smart) PoExtBus and PoNET devices

Although PoExtBus, PoExtBusSmart and PoNET devices all use the same 5-pin connector of the PoKeys device, some rules must be followed to guarantee a correct operation of all devices.

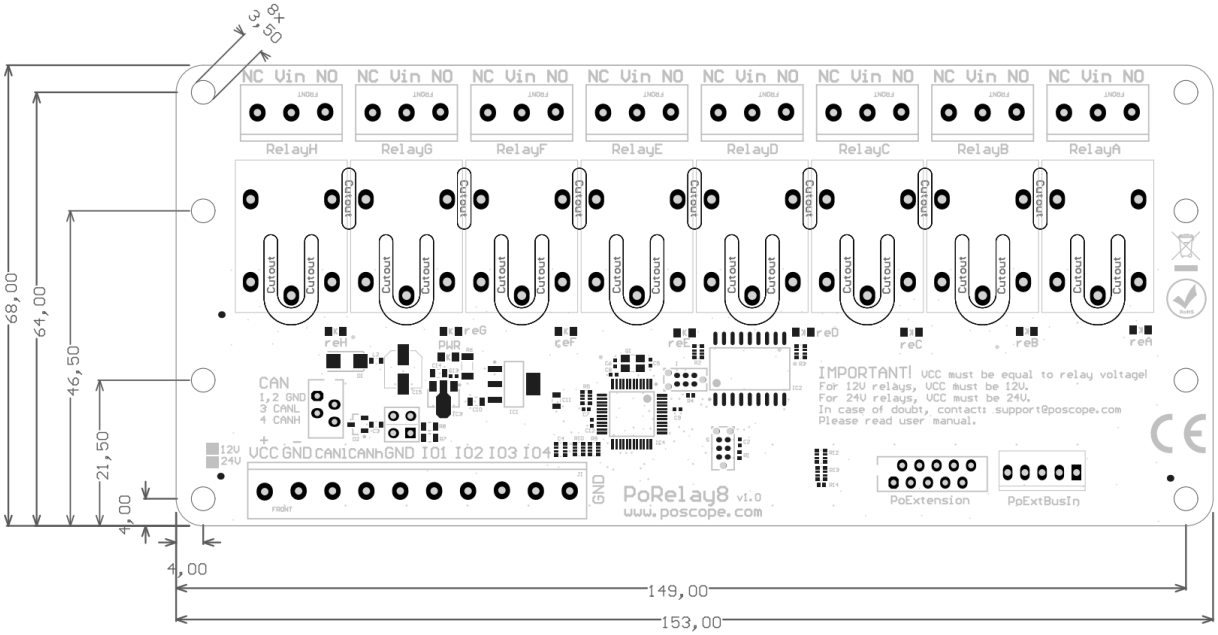
The following table provides information on how different devices should be connected to PoExtBus connector of the PoKeys device.

	PoExtBus	**PoExtBusSmart**	**PoNET**
PoExtBus	daisy-chain	parallel	parallel
PoExtBusSmart	parallel	use CAN for chaining	parallel
PoNET	parallel	parallel	parallel

There are three wiring possibilities:

- **daisy-chain**: uses the pairs of PoExtBus connectors (or PoExtension connectors on newer devices) on PoExtBus devices to daisy-chain devices one after the other with the provided cables. Devices that only have a pair of white PoExtBus connectors must be positioned first in the chain (connected to PoKeys devices on one end)
- **parallel**: both devices must be connected in parallel to the PoExtBus connector on the PoKeys device. A 1-to-2 cable splice is needed (is not currently provided by PoLabs).
- **daisy-chain** over CAN: PoExtBusSmart devices are virtually chained using the CAN bus - all PoExtBusSmart devices are connected in parallel to CAN bus connector as described in this manual.

4. PoRelay8 board dimensions



PoExtBus Smart and PoRelay8 protocols

PoExtBus is a five pin extension bus which is used to connect various peripheral devices to some PoLabs products. It is used to transfer signals (and optionally power) to the connected peripheral.

PoExtBus Smart devices feature backwards-compatibility with PoExtBus devices and add additional features to improve reliability and offer more configuration options. Additional communication uses I2C bus that is already available on the same PoExtBus connector. PoExtBus Smart devices allow configuration and firmware updates via the additional communication interface.

Detailed explanation of both PoExtBus and PoNET can be found in the PoKeys user manual. Users should pay attention to the way how PoExtBus and PoNET must be connected to PoKeys device if both types are used at the same time.

Master devices supporting PoExtBus Smart

- PoKeys56 and PoKeys57 series: PoExtBus and PoNET on a dedicated 5-pin connector

Features

- Uses only 3 signal wires to control up to 80 digital outputs
- Additional communication interface for configuration, firmware upgrades
- PoRelay8 devices can be daisy-chained (up to 10 devices) over CAN bus
- Standardized connectors
- Simple and affordable additional output devices for PoKeys master devices

Requirements

- Master device with available PoExtBus port

PoRelay8 communication protocol

PoRelay8 extends the original PoExtBus data frame of 80 bits to 88 bits, including an 8-bit CRC value as the first byte that is sent from the PoKeys device. The support for additional CRC value is implemented in PoKeys57 series devices starting with firmware release 4.2.35. Implemented CRC uses 0x8C polynomial with seed value of 0 - 1's complement of the calculated CRC value over 10 data bytes is used as the first byte sent from PoKeys device (ignored by older PoExtBus devices that accept only the last 80 bits).

PoRelay8 devices will still accept 80 bit frames in order to support operation with older PoKeys devices or third-party PoExtBus masters.

CAN data frames are used to transfer data between the bridge device (only PoRelay8 device that is connected directly to the PoKeys device via PoExtBus cable) and other PoRelay8 devices in the chain.

All multi-byte values are transferred as LSB (least significant byte) first.

PoExtBus Smart – protocol definition

PoExtBus Smart uses standard I2C communication protocol over SDA and SCL lines on the PoExtBus and PoExtension connectors. By default, I2C address of 0x7B (7-bit representation) is used.

I2C register map/commands

The following table lists i2C register map, which combines both the memory and system commands.

Name	Type	Address
Device identification	R	0x10
Configuration read	R	0x11
Configuration write	W	0x12
Configuration save	W	0x13
Set outputs	W	0x20
Set outputs x10	W	0x21
Re-enable PoExtBus	W	0x2F
PoIL command	R/W	0x30
Send CAN message	W	0x40
Read CAN message queue	R	0x41
Start bootloader	W	0xF4

Detailed command description

Device identification (0x10)

Device identification returns information on the PoRelay8 device, including device type (10/1), firmware version and 32-bit device identifier.

Request

Address 0x7B (W)	0x10

Response

Address 0x7B (R)	STATUS (0x1A)	TYPE_1 (10)	TYPE_2 (1)	FW_ver_1	FW_ver_2	Device ID (32-bit)

Configuration read (0x11)

This command is used to access the settings of the PoRelay8 device.

Request

Address 0x7B (W)	0x11	Index	Checksum

Response

Address 0x7B (R)	STATUS (0x1A)	Index	32-bit value

The following parameters are available (by index):

Table 1: Parameter indexes

Index	Description	Default value
0	Device's I2C address	0x7B
1	PoExtBus daisy-chain position (data index)	0
2	CAN daisy-chain position (data index)	0
3	Number of PoRelay8 devices on CAN bus (additional CAN frames are sent if more than 8 PoRelay8 devices present)	10
4	Failsafe timeout (in ms)	5000
5	Disable CRC check on PoExtBus	0
6	CAN bus timing option 0 – default CAN bitrate of 250 kbit/s 125 – CAN bitrate of 125 kbit/s 250 – CAN bitrate of 250 kbit/s 500 – CAN bitrate of 500 kbit/s 1000 – CAN bitrate of 1000 kbit/s	0
7	CAN bus message ID	0x108
8	PoIL master enable switch and PoIL core ID	0

Configuration write (0x12)

This command is used to access the settings of the PoRelay8 device.

Request

Address 0x7B (W)	0x12	Index	32-bit value	Checksum

Response

Address 0x7B (R)	STATUS (0x1A)	Index	32-bit value

Configuration save (0x13)

This command is used to save the settings of the PoRelay8 device to non-volatile memory.

Request

Address 0x7B (W)	0x13	0xA5	Checksum

Response

Address 0x7B (R)	STATUS (0x1A)

Set outputs (0x20)

This command is used to control the outputs state of the PoRelay8 device. The output states are encoded in the form of bit-mapped field 'Output state', where bit 0 corresponds to output H. Once this command is received, data provided over PoExtBus protocol will be ignored until 'Re-enable PoExtBus' command is executed.

Request

Address 0x7B (W)	0x20	Output state	Checksum

Response

Address 0x7B (R)	STATUS (0x1A)

Set outputs x10 (0x21)

This command is used to control the outputs state of the PoRelay8 device and other PoRelay8 devices connected to the CAN bus. The output states are encoded in the form of bit-mapped field 'Output state', where bit 0 corresponds to output H. CAN-bus daisy chain position is used by each device to determine what data is targeting each device. Once this command is received, data provided over PoExtBus protocol will be ignored until 'Re-enable PoExtBus' command is executed.

Request

Address 0x7B (W)	0x21	Output state 1	…	Output state 10	Checksum

Response

Address 0x7B (R)	STATUS (0x1A)

Re-enable PoExtBus (0x2F)

This command restores the PoExtBus decoding functionality of the PoRelay8 device.

Request

Address 0x7B (W)	0x2F

Response

Address 0x7B (R)	STATUS (0x1A)

PoIL command (0x30)

The command is used to access PoIL core functionality of the PoRelay8 device. The command uses a simulation of CAN message for sending the request and reading the response.

Request

Address 0x7B (W)	0x30	CAN message ID	CAN message data

Response

Address 0x7B (R)	STATUS (0x1A)	CAN message ID	CAN message data

Send message to CAN bus (0x40)

The command is used to access CAN bus via the PoRelay8 device in order to communicate with other PoRelay8 devices, connected via the CAN bus.

Request

Address 0x7B (W)	0x40	Flags	CAN message ID	CAN message data

7	6	5	4	3	2	1	0
Ex	R	-	-	Length			

Ex: extended frame bit
R: remote frame request
Length: message data length

Response

Address 0x7B (R)	STATUS (0x1A)

Read CAN message queue (0x41)

The command is used to access CAN bus via the PoRelay8 device in order to communicate with other PoRelay8 devices, connected via the CAN bus. The command reads from the CAN message queue.

Request

Address 0x7B (W)	0x41

Response – no CAN frame available

Address 0x7B (R)	NO_DATA (0xF2)

Request

Address 0x7B (W)	0x41

Response

Address 0x7B (R)	STATUS (0x1A)	Flags	CAN message ID	CAN message data	Chksm

7	6	5	4	3	2	1	0
Ex	R	-	-	Length			

PoCAN – protocol definition for PoRelay8 devices

PoRelay8 uses standard CAN messages with 11-bit IDs and 1-8 byte length (at 250 kbit/s). By default, the following CAN message IDs are used by the PoRelay8 device.

Name	ID	Description
CAN_PORELAY8_MSGID_STATE_UPDATE	0x112	Message with output states for up to 8 PoRelay8 devices. The appropriate one is selected by each PoRelay8 device based on the value of the parameter 2 (CAN daisy-chain position). - Bytes 0-7: output state (chain positions 0-7)
CAN_PORELAY8_MSGID_STATE_UPDATE2	0x113	Message with output states for two additional PoRelay8 devices (up to 10 devices supported). The appropriate one is selected by each PoRelay8 device based on the value of the parameter 2 (CAN daisy-chain position). - Bytes 0-2: output state (chain positions 8-9)
CAN_PORELAY8_MSGID_STATE_UPDATE_SINGLE	0x114	Single device can be addressed with this command by including target device's 32-bit ID - Bytes 0-3: target device 32-bit ID (LSB first) - Byte 4: output state
CAN_PORELAY8_MSGID_GENERAL	0x108	Command interface - this message contains general command and data layout that is used for general purpose command execution on the device. **First data byte is interpreted as command by the device.**

Supported commands by the command interface

Device identification (0x10)

Device identification returns information on the PoRelay8 device, including device type (1), firmware version and 32-bit device identifier.

Request

ID 0x108	Command 0x10

Response

ID 0x108	Command 0x10	TYPE_2 (1)	FW_ver_1	FW_ver_2	Device ID (32-bit)

Configuration read (0x11)

This command is used to access the settings of the PoRelay8 device. Only lower 16 bits of the parameter values can be accessed via CAN.

Request

ID 0x108	Command 0x11	Device ID (32-bit)	Index

Response

ID 0x108	Command 0x11	Device ID (32-bit)	Index	Value (16-bit)

See Parameter indexes table on page 15.

Configuration write (0x12)

This command is used to access the settings of the PoRelay8 device. Only lower 16 bits of the parameter values can be accessed via CAN.

Request

ID 0x108	Command 0x12	Device ID (32-bit)	Index	Value (16-bit)

Configuration save (0x13)

This command is used to save the settings of the PoRelay8 device to non-volatile memory.

Request

ID 0x108	Command 0x13	0xA5

Set outputs (0x20)

This command is used to control the outputs state of the PoRelay8 device. The output states are encoded in the form of bit-mapped field 'Output state', where bit 0 corresponds to output H.

Request

ID 0x108	Command 0x20	Device ID (32-bit)	Output state

PoIL command set

PoIL CAN commands can be encapsulated into I2C frames with data fields extended to up to 17 bytes (command COMMAND_POIL) – PoIL command (0x30) is used in I2C register map.

The value of PoIL core ID corresponds to parameter 8 of the PoRelay8 device.

Name	ID	Description
CAN_PORELAY8_POIL_CMD	0x200	PoIL command interface # Byte 0: bits 5-7: PoIL core ID bits 4-0: reserved # Byte 1: command ID ## 0x00 Report status ## 0x01 Set PoIL core state bytes 2-3: core state ## 0x02 Reset PoIL core ## 0x10 Read memory byte 2: target memory bytes 3-4: memory pointer bytes 5-6: data len ## 0x16 Erase memory byte 2: target memory ## 0x20 Get task status byte 2: task ID
CAN_PORELAY8_POIL_STATUS	0x201	PoIL status reporting interface # Byte 0: bits 5-7: PoIL core ID bits 4-0: reserved # Bytes 1-2: CoreState # Byte 3: CoreType (8) # Byte 4: Version (1)
CAN_PORELAY8_POIL_WRITE	0x205	host > PoRelay8 write interface # Byte 0: bits 5-7: PoIL core ID bits 4-3: reserved bits 2-0: target memory # Bytes 1-2: memory pointer # Bytes 3-7: data
CAN_PORELAY8_POIL_READ	0x206	PoRelay8 > host read interface # Byte 0: bits 5-7: PoIL core ID bits 4-3: reserved bits 2-0: target memory # Bytes 1-2: memory pointer # Bytes 3-7: data

CAN_PORELAY8_POIL_TASK_STATUS	0x202	PoIL task status reporting interface # Byte 0: bits 5-7: PoIL core ID bits 4-0: task ID # Byte 1: Task status # Byte 2: Task load # Bytes 3-4: Configured task period # Bytes 5-6: Real task period - filtered

Document versions

Date	Changes
10.8.2017	First release
29.5.2018	The manual was updated with information on interoperability with other devices using PoExtBus connector
30.5.2018	Extended the installation chapter with clarification of certain terms, added description that PoExtBus cable should not be extended.
7.9.2021	Added protocol descriptions

Grant of license

The material contained in this release is licensed, not sold. PoLabs grants a license to the person who is using any compatible PoLabs product, subject to the conditions listed below.

1. Access

The licensee agrees to allow access to this manual only to persons who have been informed of and agree to abide by these conditions.

2. Usage

The informations in this release is for use only with PoLabs products.

3. Copyright

PoLabs claims the copyright of, and retains the rights to, all material (software, documents etc) contained in this release. You may copy and distribute the entire release in its original state, but must not copy individual items within the release other than for backup purposes.

4. Liability

PoLabs and its agents shall not be liable for any loss or damage, howsoever caused, related to the use of PoLabs equipment or software, unless excluded by statute.

5. Fitness for purpose

No two applications are the same, so PoLabs cannot guarantee that its equipment or software is suitable for a given application. It is therefore the user's responsibility to ensure that the product is suitable for the user's application.

6. Mission Critical applications

Because the software runs on a computer that may be running other software products, and may be subject to interference from these other products, this license specifically excludes usage in 'mission critical' applications, for example life support systems.

7. Errors

This manual was continuously monitored for errors during production, however the user is responsible for error checking the manual once it is used.

8. Support

There could be errors in these manual, but if you found some, please contact our technical support staff, who will try to fix the problem within a reasonable time.

9. Upgrades

We provide upgrades, free of charge, from our web site at www.poscope.com. We reserve the right to charge for updates or replacements sent out on physical media.

10. Trademarks

Windows is a registered trademark of Microsoft Corporation. PoKeys, PoKeys55, PoKeys56U, PoKeys56E, PoKeys57U, PoKeys57E, PoKeys57CNC, PoScope, PoLabs, PoExtBus, PoExtBus Smart, PoRelay8 and others are internationally registered trademarks.

support: www.poscope.com