DH-Systems Relay Module

- **Product Name:** DH-Systems Relay Module
- **Document Version:** v2.0 (2026)
- **SKU / Product ID:** DHS-MOD-MRLY02
- **Target Audience:** DIY Builders / System Integrators

Important Safety Notice & Disclaimer

⚠ **READ BEFORE INSTALLATION**

- **DIY Product Notice:** This module is sold as a Do-It-Yourself (DIY) component for custom electronics integration. It is not a certified commercial turn-key product. Final system housing, fuse protection, and overall electrical safety are the sole responsibility of the builder.
- **Risk of Damage:** Incorrect wiring, reversed polarity, or short circuits can cause immediate and permanent damage to the module, connected sensors, and the Master Unit. Always double-check all connections with a multimeter before powering up the system for the first time.
- **Qualified Installation:** Electrical systems in vehicles and off-grid environments carry inherent risks of short circuits, fire, or battery damage. It is strongly recommended that the installation, wiring, and integration of these modules are carried out by a qualified individual with a solid understanding of DC electronics.
- **Liability:** DH-Systems cannot be held liable for any hardware damage, data loss, physical injury, or vehicular damage resulting from the incorrect assembly, programming, configuration, or installation of this equipment.
- This product is an **open-component DIY Development Board (sub-assembly)**, not a finished consumer electronics appliance. It is designed exclusively for system integration by DIY builders and qualified individuals.

1. Introduction & Core Function

The **DH-Systems Relay Module** is an industrial-grade, 8-channel digital actuator expansion board designed to seamlessly control high-current DC loads within your central system. To ensure optimal signal integrity, this module communicates via the local I2C bus and must be mounted close to the Master Unit.

Equipped with premium automotive-grade components and hardwired manual overrides, this module guarantees operational safety and diagnostic flexibility directly at the hardware layer, keeping your main control environment reliable and responsive.

Key Features:

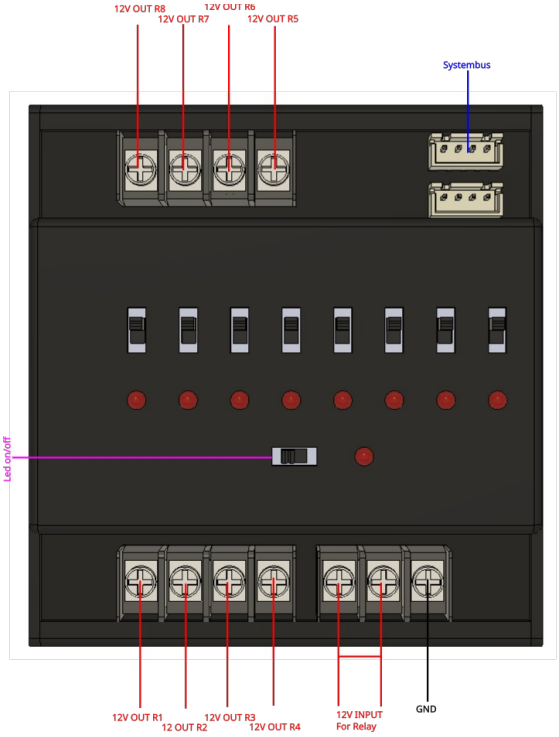
- **High-Current Actuation:** 8 independent automotive-type relays (CMA51H-S-DC12V) capable of switching up to 20A continuous load per channel.
- **Isolated Power Paths:** Volledige fysieke scheiding tussen de lichte stuurstroom (via de SystemBus) en de zware vermogensstroom (via de dedicated 12V voedingsklemmen).
- **Hardware Manual Overrides:** 8 physical slide switches on the top board allow manual activation of each relay, completely independent of the software/microcontroller state.
- **LED Diagnostic Switch:** A dedicated 9th slide switch isolates power to the status LEDs, allowing complete stealth/dark operation.
- **Inductive Load Protection:** Onboard TVS transient voltage suppression diodes (SMBJ18A) on every single output protect the PCB and relays against high-voltage spikes caused by inductive loads (e.g., pumps, fans, compressors).
- **Low Standby Power:** Drastically reduces idle power consumption compared to distributed standalone Wi-Fi relays by utilizing a single central I2C gateway.
- **Industrial Terminals:** Vibration-resistant screw terminals for robust, reliable connections in mobile camper environments.
- **Bus Type:** Local SystemBus (I2C)

2. Technical Specifications

Main Controller	PCF8574 (8-Bit I2C I/O Expander)
Operating Voltage	12V DC Nominal (Cleaned via onboard L78L33 providing 3.3V for logic circuitry)
Relay Specifications	CMA51H-S-DC12V (Automotive Type, Up to 20A Switching)
Contact Type	Normally Open (NO) Only *(NC behavior must be simulated in software)*
Transient Protection	SMBJ18A TVS Diodes on all outputs (Ideal for inductive loads)

Main Controller	PCF8574 (8-Bit I2C I/O Expander)
Connectivity / Bus	Local SystemBus via JST connector
Default I2C Address	0x20 (Configurable via onboard solder pads A0, A1, A2)
Terminal Type	Industrial-grade vibration-resistant screw terminals

3. Pinout & Wiring Diagram



Hardware Interfaces & Connections:

SystemBus Input (4-Pin JST Connector):

This port links the Relay Module to the central Master Unit or adjacent local modules, carrying primary power and data lines:

- **12V:** Main power input from the fused/protected system rail
- **GND:** Common System Ground
- **SDA:** I2C Data Line
- **SCL:** I2C Clock Line

Main Power Terminals (Screw Terminals)

- **12V IN / GND:**
  Connect the main power supply (12V) intended for the loads here. This power circuit must be correctly fused and is completely isolated from the SystemBus power supply to prevent overloading the bus cabling.

Top Board Controls & Diagnostic Switches:

- **Switches 1 to 8:**
  Individual hardwired slide switches. Moving a switch to 'ON' forces the corresponding relay to close, regardless of software state.

  ⚠ **OPERATIONAL NOTE:**
  If a relay refuses to turn off via software commands, check the physical top board switch; it will override any 'OFF' command from the controller.

- **9th Switch (LED Power):**
  Toggles power to the status indicator LEDs. When set to 'OFF', the relays function normally, but the diagnostic LEDs remain unlit to conserve power or maintain a dark environment.

4. Hardware Configuration & Solder Jumpers

I2C Address Selection

By default, the Relay Module comes pre-configured with the hardware I2C address **0x20** (all address pads connected to GND by default/open). If your system configuration requires multiple relay expanders, you can run multiple modules on the same local SystemBus.

To avoid address conflicts, alter the I2C address by bridging the dedicated solder pads (A0, A1, and A2) on the PCB to either GND or 3.3V.

(Note: Below is a standard selection matrix for the PCF8574. Ensure your physical PCB traces match this layout before assigning addresses in software).

A2 Pad	**A1 Pad**	**A0 Pad**	**Resulting I2C Address**
GND	GND	GND	**0x20 (Primary Module / Default)**
GND	GND	3.3V	0x21
GND	3.3V	GND	0x22
GND	3.3V	3.3V	0x23

A2 Pad	A1 Pad	A0 Pad	Resulting I2C Address
3.3V	GND	GND	0x24
3.3V	GND	3.3V	0x25
3.3V	3.3V	GND	0x26
3.3V	3.3V	3.3V	0x27

⚠ **CRITICAL:**
Always ensure that every module connected to the same Master Unit has a completely unique I2C address.

I2C Pull-Up Resistors (R3, R4)

The PCB includes dedicated solder pads for onboard I2C pull-up resistors (**R3** and **R4**).

- **Do Not Populate (Default):**
  In standard DH-Systems setups, the Master Unit provides the necessary bus pull-ups. These resistors should be left unpopulated.
- **When to Populate:**
  Only install these resistors if you are troubleshooting a custom bus layout where the overall resistance on the I2C data lines is too high, leading to signal degradation over multiple chained modules.

5. Software Integration & ESPHome Configuration

To make deployment clean and modular, the DH-Systems ecosystem relies on a package-based software architecture. Your central Master Unit runs the primary core configuration, while each connected physical hardware module is represented by its own plug-and-play code package.

Configuration Example (Relay Module Package)

To integrate the Relay Module into your system, include the code block below into your ESPHome configuration file.

Note: The default address in this example is set to 0x20. If you adapted your address via the physical solder pads (Section 4), update the address: line accordingly.

```
pcf8574:
  - id: 'Relay_card_1'
    address: 0x20
```

```
    pcf8575: false

output:
  - platform: gpio
    pin:
      pcf8574: Relay_card_1
      number: 0
      mode:
        output: true
    id: Relay0
    inverted: True

  - platform: gpio
    pin:
      pcf8574: Relay_card_1
      number: 1
      mode:
        output: true
    id: Relay1
    inverted: True

  - platform: gpio
    pin:
      pcf8574: Relay_card_1
      number: 2
      mode:
        output: true
    id: Relay2
    inverted: True

  - platform: gpio
    pin:
      pcf8574: Relay_card_1
      number: 3
      mode:
        output: true
    id: Relay3
    inverted: True
```

```
  - platform: gpio
    pin:
      pcf8574: Relay_card_1
      number: 4
      mode:
        output: true
    id: Relay4
    inverted: True

  - platform: gpio
    pin:
      pcf8574: Relay_card_1
      number: 5
      mode:
        output: true
    id: Relay5
    inverted: True

  - platform: gpio
    pin:
      pcf8574: Relay_card_1
      number: 6
      mode:
        output: true
    id: Relay6
    inverted: True

  - platform: gpio
    pin:
      pcf8574: Relay_card_1
      number: 7
      mode:
        output: true
    id: Relay7
    inverted: True

light:
```

```
  - platform: binary
    name: "relay_1"
    id: relay_1
    output: Relay0
  - platform: binary
    name: "relay_2"
    id: relay_2
    output: Relay1
  - platform: binary
    name: "relay_3"
    id: relay_3
    output: Relay2
  - platform: binary
    name: "relay_4"
    id: relay_4
    output: Relay3
  - platform: binary
    name: "relay_5"
    id: relay_5
    output: Relay4
  - platform: binary
    name: "relay_6"
    id: relay_6
    output: Relay5
  - platform: binary
    name: "relay_7"
    id: relay_7
    output: Relay6
  - platform: binary
    name: "relay_8"
    id: relay_8
    output: Relay7
```

Exposed Home Assistant Entities

Once deployed, ESPHome automatically pushes the following light entities to Home Assistant via the Native API:

- **relay_1** t/m **relay_8** (light.relay_x):
  Control entities to toggle each individual channel on or off.

  💡 **SOFTWARE NOTE (NC Behavior):**
  This board physically breaks out the Normally Open (NO) paths. If your use-case requires Normally Closed (NC) behavior, invert the logic within your automations or template entities in Home Assistant.

6. Troubleshooting & Safety Warnings

- ⚠ **WARNING:** Always disconnect your solar panels and the main 12V camper battery system before wiring or making changes to the local I2C network bus to prevent accidental short circuits or ground loops.

Common Issue - Entity showing NaN or Unavailable / Relays Not Responding:

- Verify that the SDA and SCL lines are not swapped in the JST connection.
- Ensure your central Master Unit is powered on and actively supplying stable power to the module.
- Check if the I2C address assigned via the solder pads (A0, A1, A2) exactly matches the value in your YAML configuration file (Default: 0x20).
- **Hardware Override Check:** If a relay remains energized even when turned off in Home Assistant, physically check the top board slide switch for that channel to ensure it isn't set to manual 'ON'.