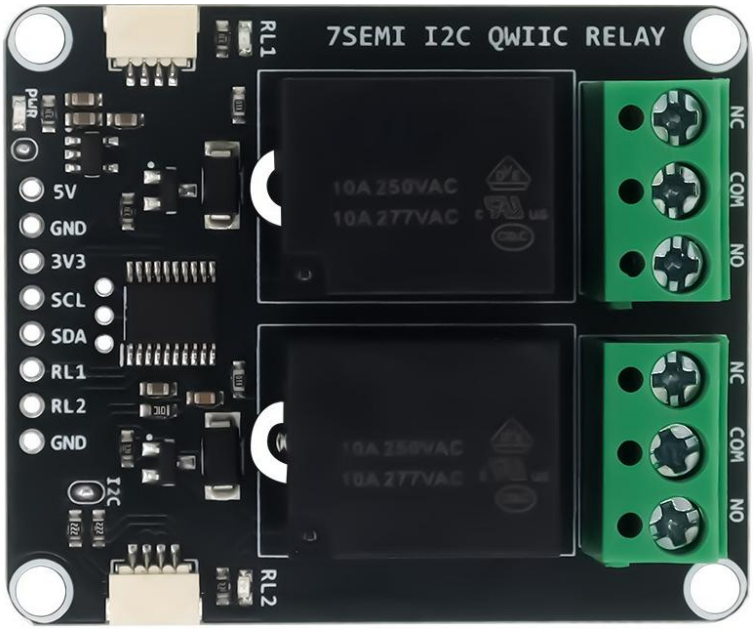
7Semi

Dual Channel I2C (Qwiic) 5V 10A Relay Module

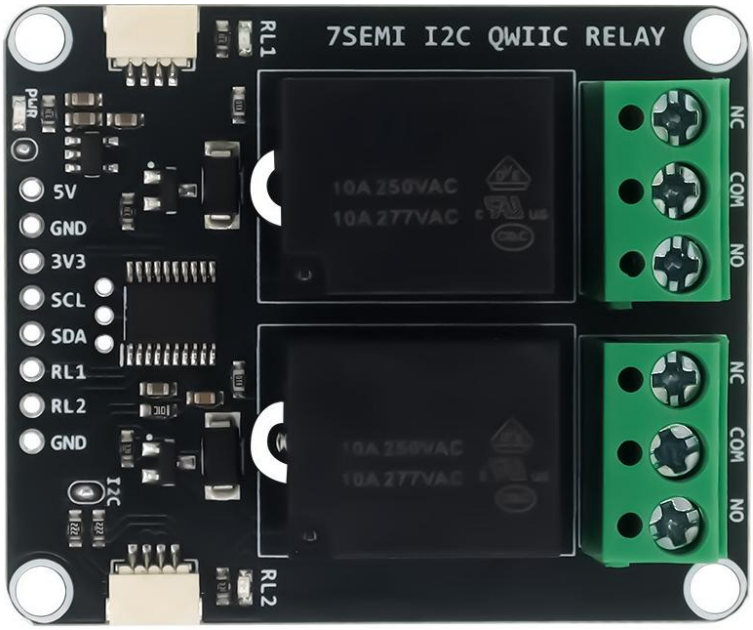
Version 1.1



7Semi +91 8655821342 info@7semi.com www.7semi.com

Table of Contents

1.Introduction



The **7Semi Dual Channel I2C (Qwiic) 5V 10A Relay Module** is a compact and reliable industrial control solution designed for automation and remote switching applications. It features **2 high-performance relay outputs**, enabling efficient control of electrical loads through a robust I2C communication interface.

The module supports **manual control** for direct user interaction, allowing seamless operation alongside remote I2C commands. For multi-device applications, the board includes **hardware address selection via back-side solder jumpers**, enabling flexible integration on a shared I2C bus.

Designed for continuous operation, the module ensures **stable performance with CRC-protected communication, watchdog safety**. Its rugged design and noise-immune communication make it suitable for **industrial environments**, including control panels, embedded systems, and smart automation applications.

1.1 Features

- 2 Industrial-grade Relay Outputs
- Manual Control (Local Operation)
- I2C Slave Interface (7-bit addressing)
- Hardware Address Selection via Back-side Solder Jumpers
- Compact, Rugged & Industrial Design

2.Technical specification

The **7Semi Dual Channel I2C (Qwiic) 5V 10A Relay Module** is designed to deliver reliable and efficient relay control for industrial and automation applications. The module provides **2 independent relay outputs**, allowing flexible control of electrical loads through a robust I2C communication interface.

The module supports a standard **I2C slave interface with hardware-configurable addressing via back-side solder jumpers**, enabling seamless integration of multiple devices on a shared I2C bus. Each relay channel can be individually controlled or grouped using a simple command protocol with **CRC-based data validation**, ensuring secure and error-free communication.

In addition to remote control, the module features **manual inputs** for local operation, allowing direct user control even without a master controller. Designed for continuous industrial use, with its **robust GPIO-driven relay control and noise-immune design**, the module ensures stable and consistent performance in harsh environments, making it ideal for control panels, embedded systems, and industrial automation applications.

Applications

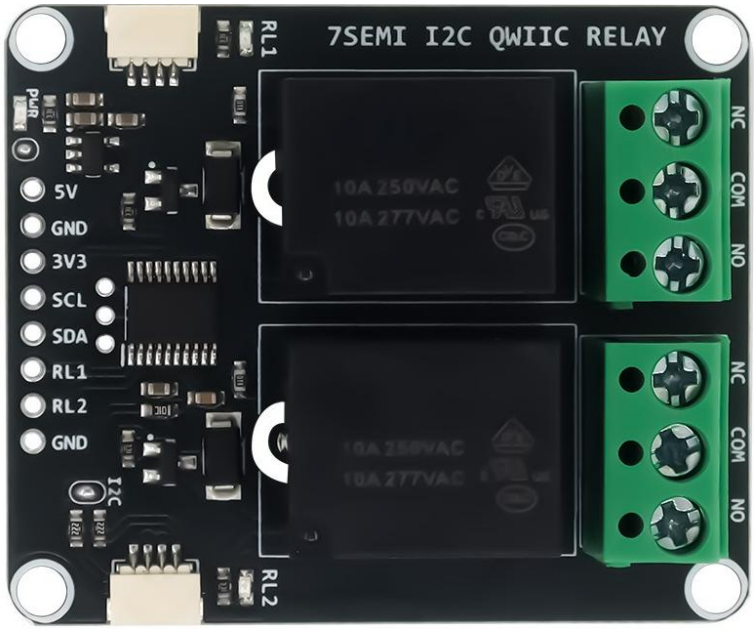
- HVAC Control Systems
- Building Automation Systems
- Industrial Automation & Control Panels
- PLC Expansion Modules
- Remote Relay Switching Systems
- Smart Energy Management Systems
- Process Monitoring & Control
- IoT & Embedded Control Systems

2.1 General specifications

- **Relay Channels:** 2 (Independent Control)
- **Communication Interface:** I2C (7-bit Slave)
- **Address Configuration:** Hardware Selectable via Back-side Solder Jumpers
- **Operating Voltage:** 5V DC
- **Relay Contact Rating:** 10A @ 250VAC / 30VDC
- **Manual Control:** Input Supported
- **Data Integrity:** CRC-based Communication
- **Watchdog Protection:** Supported
- **Operating Temperature:** -20°C to +70°C
- **Storage Temperature:** -40°C to +85°C
- **Operation Mode:** Continuous 24/7 Operation

3. Hardware setup

3.1 Pinouts

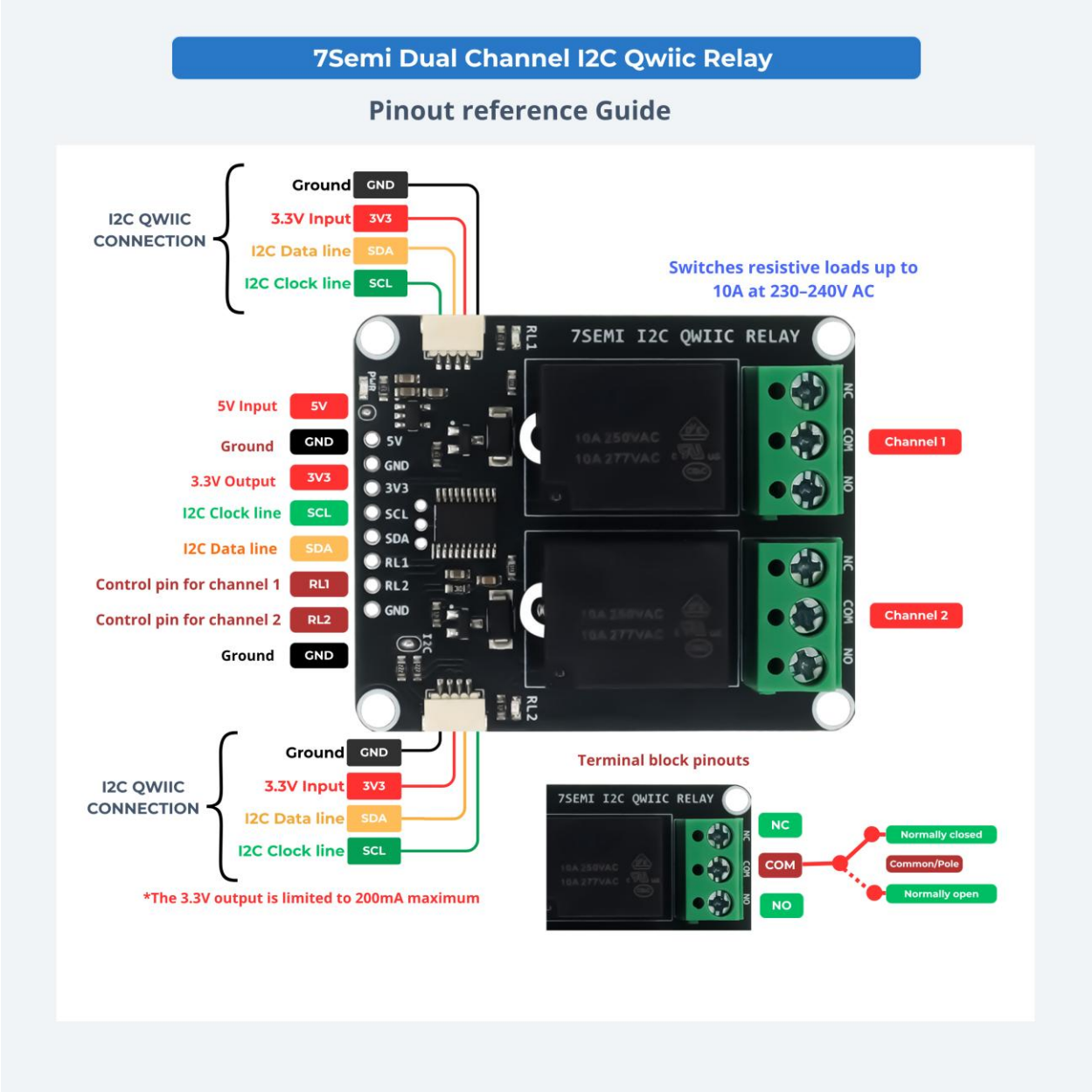


Wire Color	Function
Yellow	I2C SCL
Green	I2C SDA
Red	VCC
Black	GND

Connection Guidelines

- Power Supply: Connect the red wire to a 5V DC power source and the black wire to ground (GND). Ensure stable power for reliable operation.
- **I2C Communication:** Connect the **yellow wire** to **SCL** and the **green wire** to **SDA** of the master device.
- **Wiring Recommendation:** Use short and properly routed wires for I2C communication to maintain signal integrity and avoid noise issues.

3.2 Hardware connection



4. I2C Communication Protocol

The **7Semi Dual Channel I2C (Qwiic) 5V 10A Relay Module** utilizes a simple and efficient **I2C communication protocol** for device control and status monitoring. The module operates as an **I2C slave device**, responding to commands from a master such as microcontrollers, embedded systems, or control units.

Communication is based on a compact frame structure consisting of **register, data, and CRC fields**, ensuring reliable data exchange. A built-in **CRC (Cyclic Redundancy Check)** mechanism provides error detection for robust and noise-immune communication in industrial environments.

The module supports both **relay control commands and status read operations**, enabling flexible integration into automation systems. Local manual inputs operate independently and are internally combined with I2C commands for reliable hybrid control.

Supported Command Registers

Register	Description
0x01	Set Relay State (All Relays)
0x02	Turn ON Selected Relay(s)
0x03	Turn OFF Selected Relay(s)
0x00	Read Current Relay State

Command Description

- **Register 0x01:** Used to set the full relay state using a bitmask (e.g., 0x03 = all ON).
- **Register 0x02:** Sets (turns ON) specific relay(s) without affecting others.
- **Register 0x03:** Clears (turns OFF) specific relay(s) without affecting others.
- **Register 0x00:** Returns current relay status as a bitmask value.

Communication Notes

- The device operates as an **I2C slave with hardware-selectable address via back-side solder jumpers**.
- All commands include a **CRC byte for data validation**.
- Manual input and I2C control are internally combined for **reliable hybrid operation**.
- Supports **continuous communication for real-time control**.

Register Address Mapping - Address quick reference

Register Address	Purpose	Description
0x00	Read Relay State	Returns current relay status (bitmask)
0x01	Write Full State	Set all relays using bitmask
0x02	Set Relay ON	Turn ON selected relay(s)
0x03	Set Relay OFF	Turn OFF selected relay(s)
0x10	Device Address	Current I2C slave address (read-only)
0x11	New Address	Set new slave address (optional)
0x20	Saved State	Last stored relay state (flash)
0x30	System Status	Device status / health info

Relay Bit Mapping

Bit Position	Relay
Bit 0	Relay 1
Bit 1	Relay 2

➢ **Notes**

• Relay control uses **bitmask format**
Example: 0x03 → Relay 1 & Relay 2 ON

• All write commands include **CRC validation**

• Device operates as an **I2C slave** with address configurable via **back-side solder jumpers**

• Supports **individual and group relay control**

• Manual input and I2C commands are internally combined for **hybrid operation**

➢ Testing Steps & Standard I2C Frame

1) Basic Relay Control Test

➢ **Test Steps (Host Controller / Arduino / MCU)**

1. **Initialize I2C Master**
2. **Set Slave Address (Default: 0x20)**
3. **Send command to:**
    - **Command:** ALL ON
    - **Command:** ALL OFF
    - **Command:** R1 ON / R1 OFF
    - **Command:** R2 ON / R2 OFF
    - **Command:** READ

- Expected :
    - Relay outputs respond **immediately** to commands
    - Each relay operates **independently**
    - READ command returns correct **bitmask status**
    - Manual button and I2C commands work **simultaneously (hybrid control)**

➢ **Test Steps (Low-Level I2C Frame)**

Write Frame Format: [Slave Address] [Register] [Data] [CRC]
Example (All Relay ON): 20 01 03 02

Field	Bytes	Value	Remarks
Slave address	1	0x20	Device address
Register	1	0x01	Write full state
Data	1	0x03	All relays ON
CRC code	1	02 (XOR)	Data integrity

- **Relay Control Examples**

Operation	Command (HEX)	CRC	Full Frame	Description
All ON	20 01 03	02	20 01 03 02	Turn ON all relays
All OFF	20 01 00	01	20 01 00 01	Turn OFF all relays
Relay 1 ON	20 02 01	03	20 02 01 03	Set Relay 1
Relay 2 ON	20 02 02	00	20 02 02 00	Set Relay 2
Relay 1 OFF	20 03 01	02	20 03 01 02	Clear Relay 1
Relay 2 OFF	20 03 02	01	20 03 02 01	Clear Relay 2
READ	20 00	---		

- **Read Relay State**

  Step 1: Send register

  ```
  20 00
  ```

  Step 2: Read 1 byte response

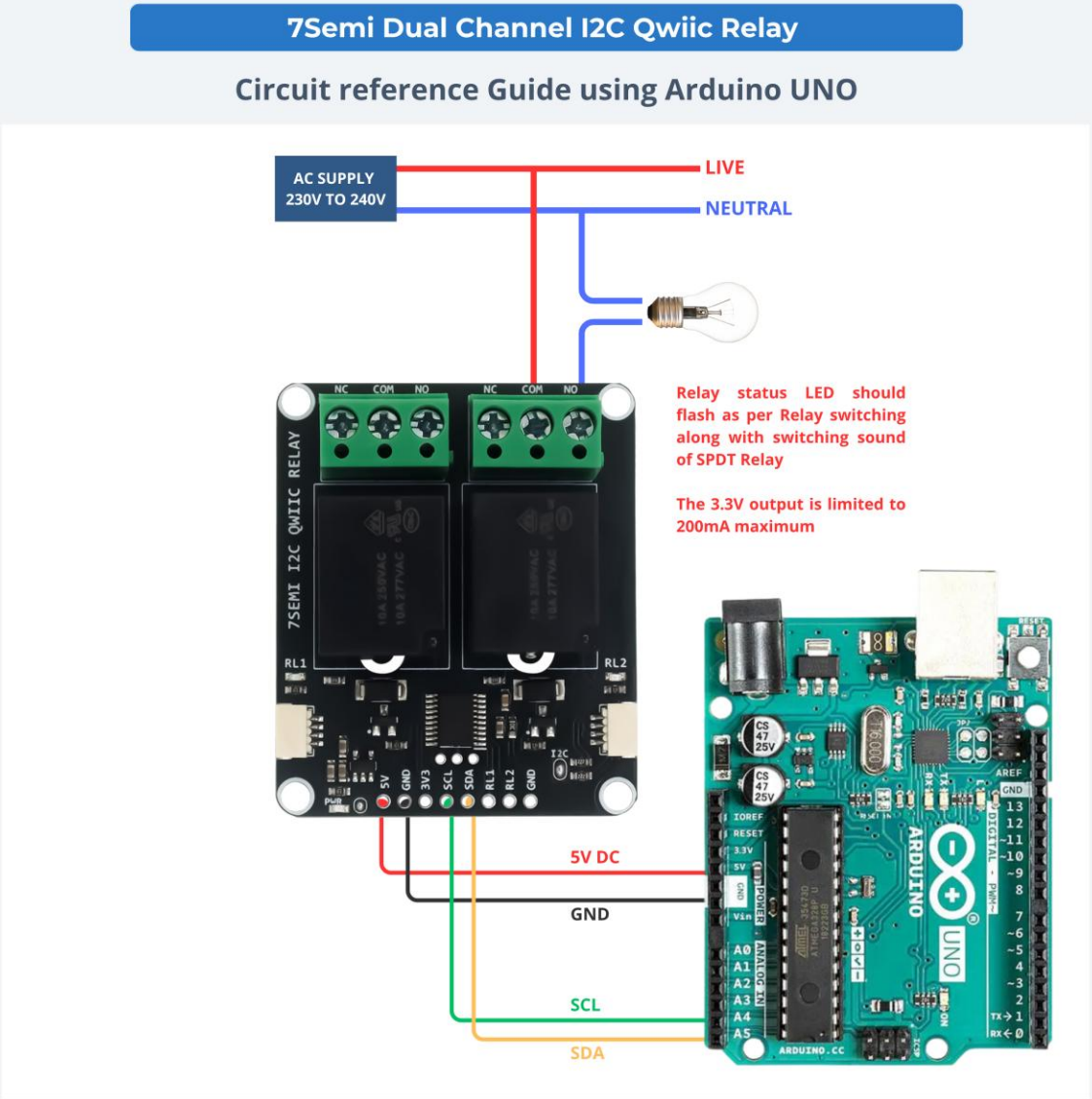
- **Response Format**

Bit	Relay	Description
Bit 0	Relay 1	ON = 1
Bit 1	Relay 2	ON = 1

- **Example**

Response	Binary	Active Relays
0x00	0000	All ON
0x01	0001	Relay 1 ON
0x02	0010	Relay 2 ON
0x03	0011	Relay 1 & Relay 2 ON

4.1 Connection and sample code for Arduino



4.2 How to Install 7Semi Arduino Library

- **Install the Arduino IDE** on your computer.
- Once installed, **open the Arduino IDE**.
- Go to **Library Manager** and search for **"7Semi I2C Relay."**
- **Install** the library from the search results.
- After installation, navigate to **File → Examples → Basic → IndustrialRelayMaster**, example.
- **Upload** the example code to your Arduino board.
- Set the **baud rate to 115200** in the Serial Monitor.
- You should now **see the output** displayed on the Serial Monitor Use that command to drive relay like R1 ON => Relay 1 ON , R1 OFF => Relay 1 OFF .

Sample code link for manual download:- *"7Semi I2C Relay "*

Sample output image:-

```
==== INDUSTRIAL RELAY MASTER ====
ON / OFF
R1 ON / R1 OFF
R2 ON / R2 OFF
R3 ON / R3 OFF
R4 ON / R4 OFF
READ
```

5.Mechanical specification