I2C-ER1001

I2C-Bus Burst Fire Controller for Solid State Relay

1 Feature

- Burst Fire Control System
- Controls Solid State Relays via I2C bus
- Supports 100Khz and 400Khz I2C bus frequencies
- Onboard pull-up resistors for SCL and SDA lines
- 10-Step Solid State Relay Control
- Operating Voltage: 3Vdc – 5Vdc
- Addressing up to 4 addresses via jumpers
- PCB size optimized for mounting on Solid State Relays
- Suitable for Thermal Management Applications
- LED indicator included
- Control up to 4 SSRs on a single I2C bus.
- Period time: 200ms

2 Description

The I2C-ER1001 is a burst fire controller board designed for use with zero-crossing solid state relays (SSRs). The board mounts directly onto the SSRs and operates by varying their duty cycle. The power delivered to the load is proportional to the I2C commands received.

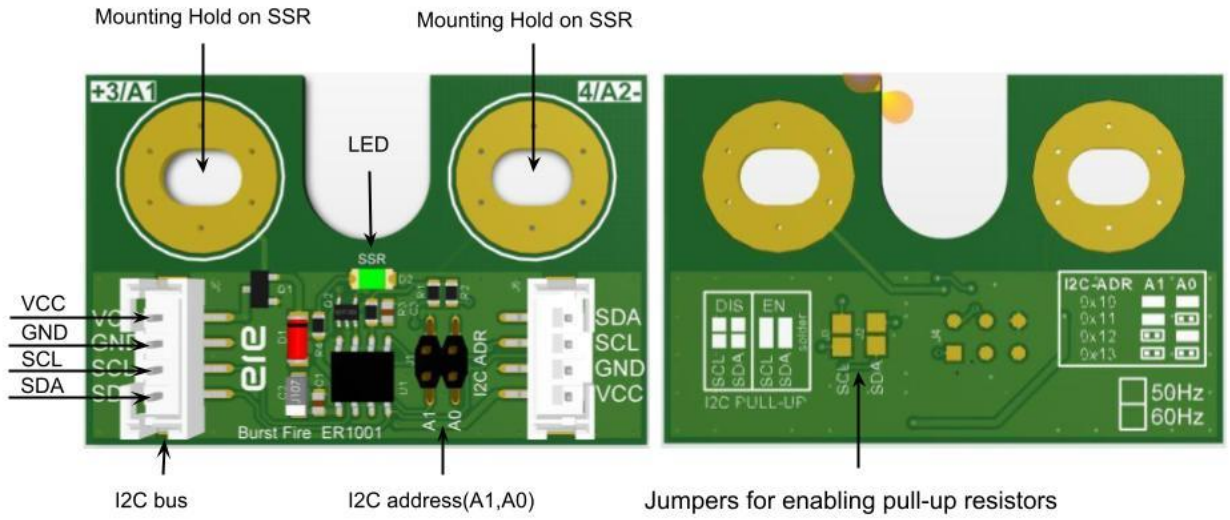


Figure 1: Board Diagram

3 Output Cycles

The table below shows an example of the output cycles for each control value. The burst fire board adjusts the SSR duty cycles based on the input value. The period time is 200ms

Table 1: Example of output cycles in a 50Hz system

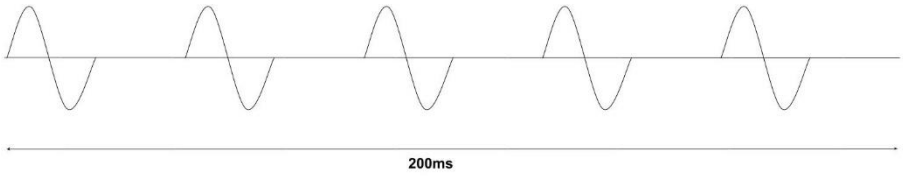
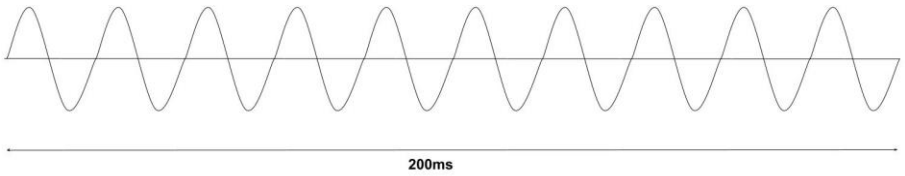
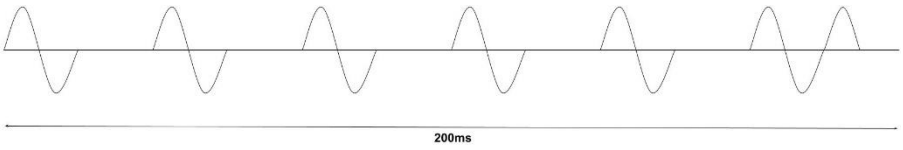
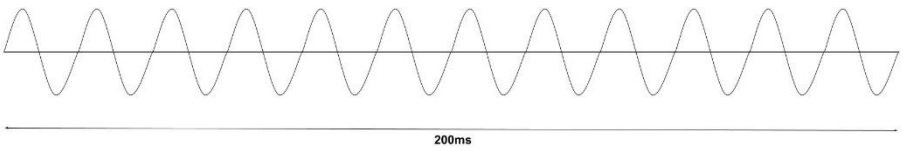
Values	Duty Cycles	Graphs
0	0.00%	
5	50.00%	
10	100.00%	

Table 2: Example of output cycles in a 60Hz system

Values	Duty Cycles	Graphs
0	0.00%	
5	54.16%	
10	100.00%	

4 Mounting Instructions

The Solid-State Relay (SSR) used with the burst fire controller board must be a zero-crossing type.

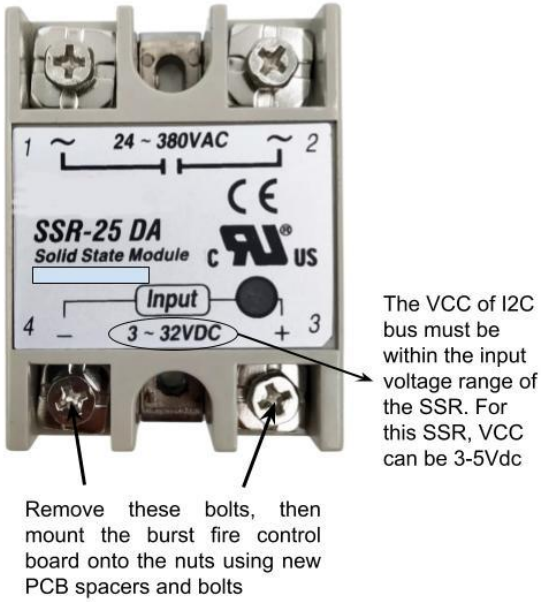


Figure 2: Example of a Solid-State Relay (SSR) used with the controller board

Figure 3: Installation diagram

The nuts are M4 with a length of 16mm, and the PCB spacers are 4mm in hold diameter and 10mm in length. In case your SSR uses different nut sizes, you will need to prepare nuts for this case.

5 I2C bus Commands

5.1 Write Mode

In writing mode, the master writes data to the slave (I2C-ER1001). The first byte is I2C slave address with RW=0, and the second byte is the data to be written to the slave. For the board, the data value must range from 0 to 10. A value of zero means no output signal (0% of duty cycle), while the value of ten means full output signal (100% duty cycle). If a value greater than ten is written to the slave, the slave will automatically reduce it to ten.

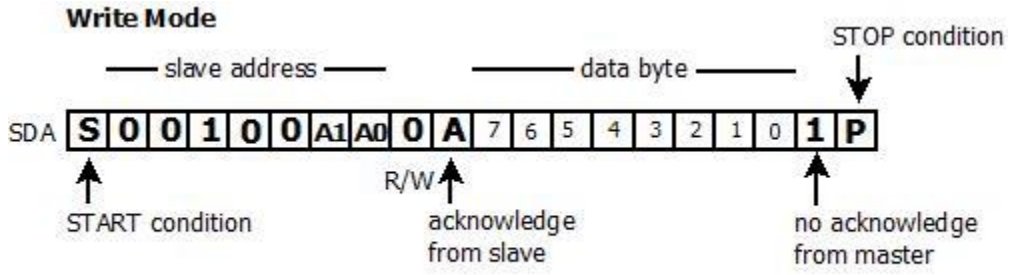


Figure 4: Write data to the board

5.2 Read mode

The master can determine the current duty cycle value by reading data via the I2C bus. The master first sends the address byte with RW=1, then reads the data from the slave. The value of data is between 0 to 10.

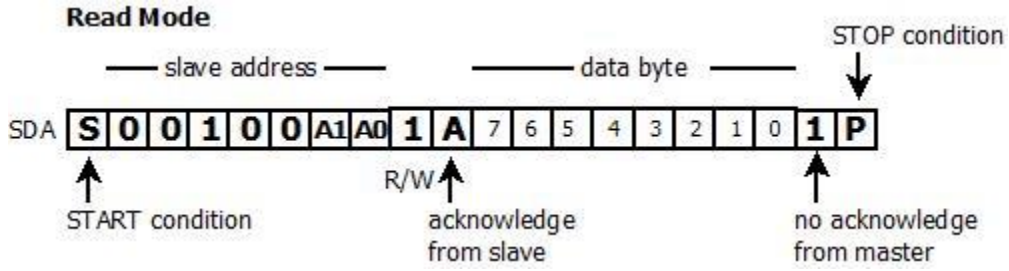
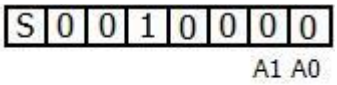
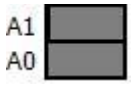
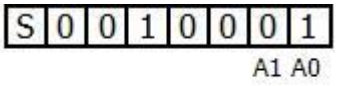
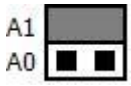
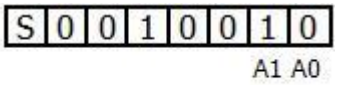
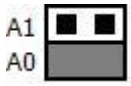
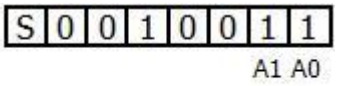
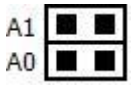


Figure 5: Read data from the board

6 I2C bus Addressing

Table 3: Jumper Setting for I2C addresses

I2C Addresses	Address Byte	Jumper Setting
0x10		
0x11		
0x12		
0x13		

7 Data Byte Values

Table 4: Output duty cycles for each input value

Values	% Duty Cycle in a 50Hz System	% Duty Cycle in a 60Hz System
0	0.00%	0.00%
1	10.00%	12.50%
2	20.00%	25.00%
3	30.00%	29.16%
4	40.00%	45.83%
5	50.00%	54.16%
6	60.00%	66.00%
7	70.00%	75.00%
8	80.00%	83.33%
9	90.00%	91.66%
10	100.00%	100.00%

8 Interfacing

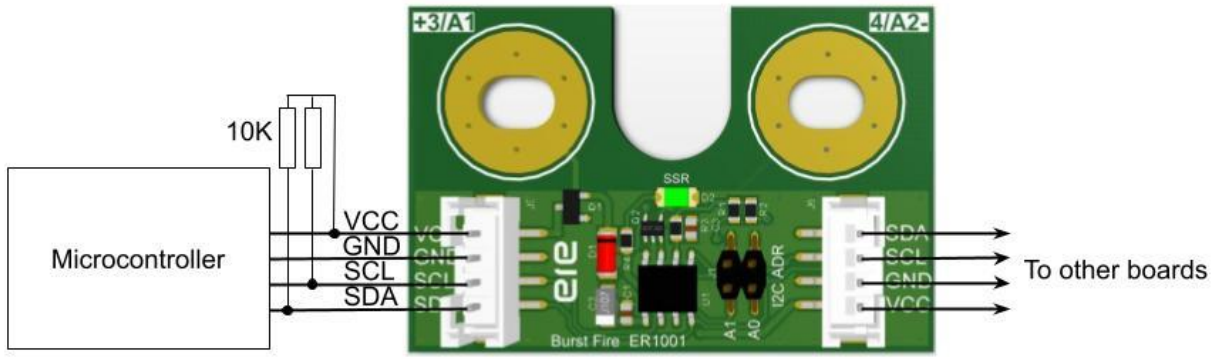


Figure 6: Interfacing microcontroller with the board via I2C bus

The I2C bus requires pull-up resistors for the SCL and SDA lines. Therefore, two resisters must be connected externally to pull up these lines, as shown in the picture. However, by soldering the jumpers (shorting the jumpers) underneath the board, the pull-up resistors are enabled, eliminating the need for external resistors.

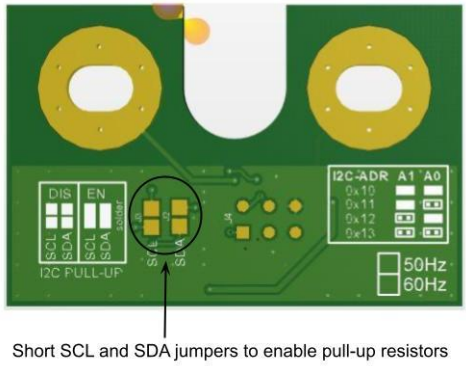


Figure 7: Jumper settings for pull-up resistors

9 Multiple Units

Up to four boards can be connected to a single I2C bus. Each board must have a unique I2C address.

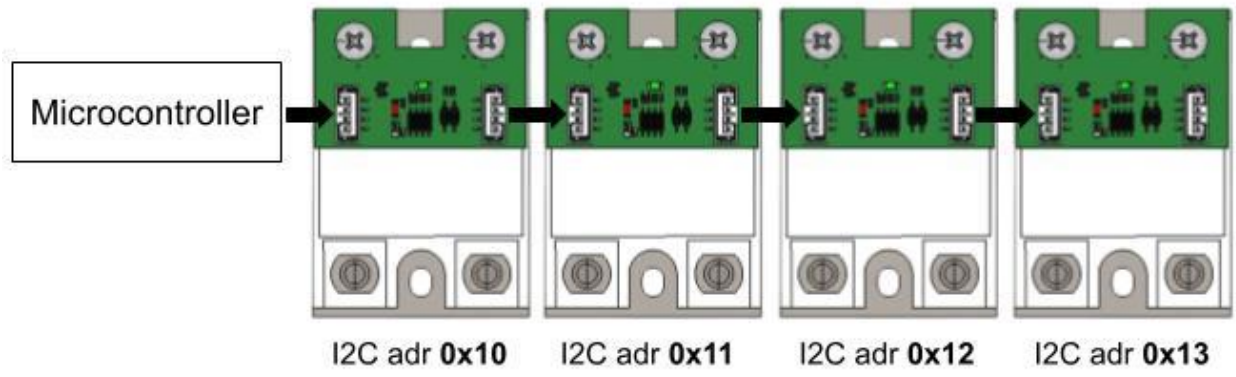


Figure 8: Connecting four boards on a single I2C bus

10 Dimensions

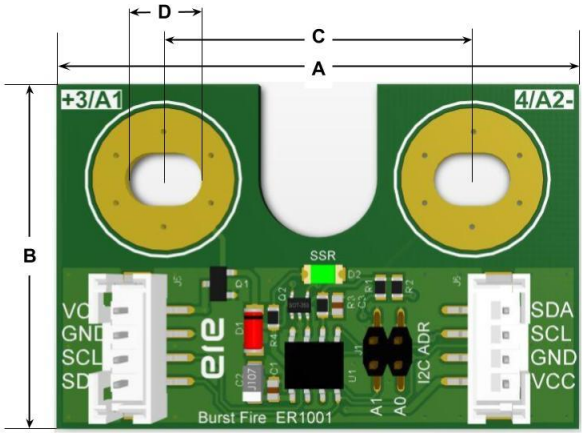


Figure 9: Board dimensions

Units	mm	inch
A	44.00	1.7323
B	29.00	1.1417
C	26.00	1.0236
D	6.50	0.2559

11 Application

The I2C burst fire board is designed to control thermal systems, such as heater control, through a microcontroller via the I2C bus.

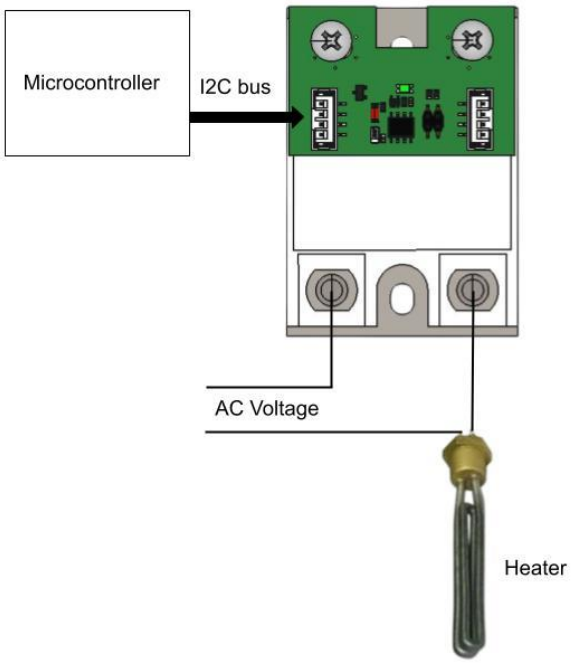


Figure 10: Heater control system