

RELAY SHIELD (10 CHANNELS)

(ARDUINO COMPATIBLE)
PRODUCT CODE: M00270061

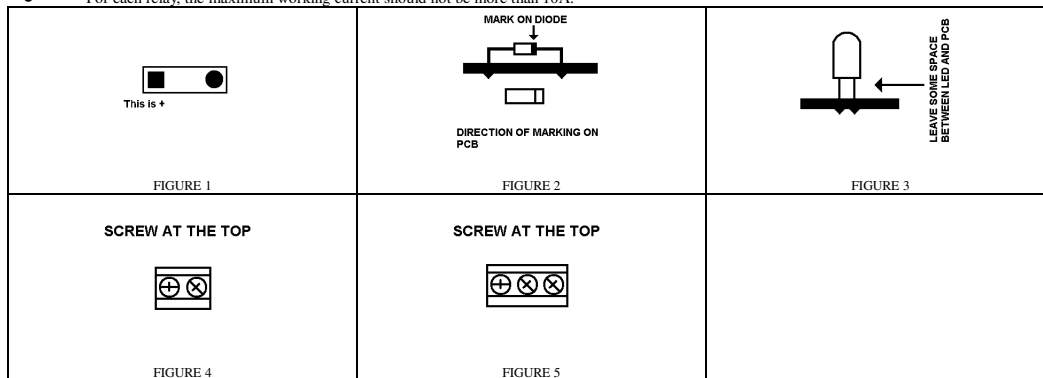
FEATURE:

- 10 channels SPDT relay output
- For each relay output, there is NC, NO and common contact.
- 6 screw type analog input (With resistor and zener diode for overvoltage protection)
- Screw type input/output for Vin, 5V, G and AREF
- One independent push button switch which connected to Interrupt pin of UNO (Propose is set by Sketch writer).
- 10 LED indications for change of state for each relay
- Assembly is needed.
- Arduino Sketch example for showing simplest input and output is attached.
- Requires 1 Arduino UNO (not included).



READ BEFORE INSTALLATION:

- Put the component on the side of screen printing and solder on the back of PCB without printing.
- Placing direction of component.
- 1. On component, longer leg is “+”.
- 2. On PCB marking, square pad as Figure 1 is always “+”.
- 3. For diode, please install as Figure 2.
- Do not put the LED to very bottom, just install as Figure 3.
- For each relay, the maximum working current should not be more than 10A.



DESCRIPTION:

The circuit design is based on the Arduino UNO. Of course, this can be used on any brand of Microcontroller or external circuit if the pin location and voltage is matched. If this is not matched, just route this yourself.

The switches for Interrupt work independent and the propose is totally dependent on Sketch writer.

When connect to the sensor to A0----A5, there must be at least one additional wire from the sensor such as ground of sensor connecting to the ground of this

shield. If not, this shield would not know where (Potential) you are.

INSTALLATION:

Just install the component to the PCB M00260099 according to below table.

ITEM	SYMBOL ON PCB	DESCRIPTION	OUTLOOK	DIRECTION IS IMPORTANT?
1	R1	RESISTOR, 1K ohms	BROWN, BLACK, RED	NO
2	R2	RESISTOR, 1K ohms	BROWN, BLACK, RED	NO
3	R3	RESISTOR, 1K ohms	BROWN, BLACK, RED	NO
4	R4	RESISTOR, 1K ohms	BROWN, BLACK, RED	NO
5	R5	RESISTOR, 1K ohms	BROWN, BLACK, RED	NO
6	R6	RESISTOR, 1K ohms	BROWN, BLACK, RED	NO
7	R7	RESISTOR, 1K ohms	BROWN, BLACK, RED	NO
8	R8	RESISTOR, 1K ohms	BROWN, BLACK, RED	NO
9	R9	RESISTOR, 1K ohms	BROWN, BLACK, RED	NO
10	R10	RESISTOR, 1K ohms	BROWN, BLACK, RED	NO
11	R11	RESISTOR, 4.7K ohms	YELLOW, VIOLET, RED	NO
12	R12	RESISTOR, 4.7K ohms	YELLOW, VIOLET, RED	NO
13	R13	RESISTOR, 4.7K ohms	YELLOW, VIOLET, RED	NO
14	R14	RESISTOR, 4.7K ohms	YELLOW, VIOLET, RED	NO
15	R15	RESISTOR, 4.7K ohms	YELLOW, VIOLET, RED	NO
16	R16	RESISTOR, 4.7K ohms	YELLOW, VIOLET, RED	NO
17	D1	DIODE, IN4001	FIGURE 2	FIGURE 2
18	D2	DIODE, IN4001	FIGURE 2	FIGURE 2
19	D3	DIODE, IN4001	FIGURE 2	FIGURE 2
20	D4	DIODE, IN4001	FIGURE 2	FIGURE 2
21	D5	DIODE, IN4001	FIGURE 2	FIGURE 2
22	D6	DIODE, IN4001	FIGURE 2	FIGURE 2
23	D7	DIODE, IN4001	FIGURE 2	FIGURE 2
24	D8	DIODE, IN4001	FIGURE 2	FIGURE 2
25	D9	DIODE, IN4001	FIGURE 2	FIGURE 2
26	D10	DIODE, IN4001	FIGURE 2	FIGURE 2
27	Z1	ZENER DIODE, 5.1V	FIGURE 2 (MOSTLY TRANSPARAENT RED)	FIGURE 2
28	Z2	ZENER DIODE, 5.1V	FIGURE 2 (MOSTLY TRANSPARAENT RED)	FIGURE 2
29	Z3	ZENER DIODE, 5.1V	FIGURE 2 (MOSTLY TRANSPARAENT RED)	FIGURE 2
30	Z4	ZENER DIODE, 5.1V	FIGURE 2 (MOSTLY TRANSPARAENT RED)	FIGURE 2
31	Z5	ZENER DIODE, 5.1V	FIGURE 2 (MOSTLY TRANSPARAENT RED)	FIGURE 2
32	Z6	ZENER DIODE, 5.1V	FIGURE 2 (MOSTLY TRANSPARAENT RED)	FIGURE 2
33	RELAY1	RELAY	BIG RECTANGLE BOX WITH FIVE LEGS	YES
34	RELAY2	RELAY	BIG RECTANGLE BOX WITH FIVE LEGS	YES
35	RELAY3	RELAY	BIG RECTANGLE BOX WITH FIVE LEGS	YES
36	RELAY4	RELAY	BIG RECTANGLE BOX WITH FIVE LEGS	YES
37	RELAY5	RELAY	BIG RECTANGLE BOX WITH FIVE LEGS	YES
38	RELAY6	RELAY	BIG RECTANGLE BOX WITH FIVE LEGS	YES
39	RELAY7	RELAY	BIG RECTANGLE BOX WITH FIVE LEGS	YES
40	RELAY8	RELAY	BIG RECTANGLE BOX WITH FIVE LEGS	YES
41	RELAY9	RELAY	BIG RECTANGLE BOX WITH FIVE LEGS	YES
42	RELAY10	RELAY	BIG RECTANGLE BOX WITH FIVE LEGS	YES
43	L1	LED	RED	YES
44	L2	LED	RED	YES
45	L3	LED	RED	YES
46	L4	LED	RED	YES
47	L5	LED	RED	YES
48	L6	LED	RED	YES
49	L7	LED	RED	YES
50	L8	LED	RED	YES
51	L9	LED	RED	YES
52	L10	LED	RED	YES
53	S1	STACKABLE HEADER – 8PIN	LONG 8 PIN	NO
54	S2	STACKABLE HEADER – 10PIN	LONG 10 PIN	NO
55	S3	STACKABLE HEADER – 6PIN	LONG 6 PIN	NO
56	S4	STACKABLE HEADER – 8PIN	LONG 8 PIN	NO
57	S5	VIN, 5V OUTPUT	FIGURE 4	NOTE 1
58	S6	G, AREF OUTPUT	FIGURE 4	NOTE 1
59	S7	A0, A1 OUTPUT	FIGURE 4	NOTE 1
60	S8	A2, A3 OUTPUT	FIGURE 4	NOTE 1
61	S9	A4, A5 OUTPUT	FIGURE 4	NOTE 1
62	S10	RELAY1 OUTPUT	FIGURE 5	NOTE 1
63	S11	RELAY2 OUTPUT	FIGURE 5	NOTE 1
64	S12	RELAY3 OUTPUT	FIGURE 5	NOTE 1
65	S13	RELAY4 OUTPUT	FIGURE 5	NOTE 1
66	S14	RELAY5 OUTPUT	FIGURE 5	NOTE 1
67	S15	RELAY6 OUTPUT	FIGURE 5	NOTE 1
68	S16	RELAY7 OUTPUT	FIGURE 5	NOTE 1
69	S17	RELAY8 OUTPUT	FIGURE 5	NOTE 1
70	S18	RELAY9 OUTPUT	FIGURE 5	NOTE 1
71	S19	RELAY10 OUTPUT	FIGURE 5	NOTE 1
72	F1	PUSH BUTTON SWITCH	FOUR LEGS	NO

NOTE 1. Metal terminal is facing outside the PCB.

[illegible]

```

/*
 * This program would test if each relay work normally.
 * Also, we have added an interrupt on this
 */

/* Pin declaration for each relay */
int relay1 = 4; int relay2 = 5; int relay3 = 6;
int relay4 = 7; int relay5 = 8; int relay6 = 9;
int relay7 = 10; int relay8 = 11; int relay9 = 12;
int relay10 = 13;

/* Interrupt variable */
const byte interruptPin = 3;
volatile byte count = 0;

void setup() {

    pinMode(relay1, OUTPUT); pinMode(relay2, OUTPUT);
    pinMode(relay3, OUTPUT); pinMode(relay4, OUTPUT);
    pinMode(relay5, OUTPUT); pinMode(relay6, OUTPUT);
    pinMode(relay7, OUTPUT); pinMode(relay8, OUTPUT);
    pinMode(relay9, OUTPUT); pinMode(relay10, OUTPUT);
    digitalWrite(relay1, LOW); digitalWrite(relay2, LOW);
    digitalWrite(relay3, LOW); digitalWrite(relay4, LOW);
    digitalWrite(relay5, LOW); digitalWrite(relay6, LOW);
    digitalWrite(relay7, LOW); digitalWrite(relay8, LOW);
    digitalWrite(relay9, LOW); digitalWrite(relay10, LOW);

/* Interrupt setting */
    pinMode(interruptPin, INPUT_PULLUP);
    attachInterrupt(digitalPinToInterrupt(interruptPin), INT, FALLING);

}

void loop() {

    digitalWrite(relay1, HIGH); delay(500); digitalWrite(relay1, LOW);
    digitalWrite(relay2, HIGH); delay(500); digitalWrite(relay2, LOW);
    digitalWrite(relay3, HIGH); delay(500); digitalWrite(relay3, LOW);
    digitalWrite(relay4, HIGH); delay(500); digitalWrite(relay4, LOW);
    digitalWrite(relay5, HIGH); delay(500); digitalWrite(relay5, LOW);
    if (count == 0) {
        digitalWrite(relay6, HIGH); delay(500); digitalWrite(relay6, LOW);
        digitalWrite(relay7, HIGH); delay(500); digitalWrite(relay7, LOW);
        digitalWrite(relay8, HIGH); delay(500); digitalWrite(relay8, LOW);
        digitalWrite(relay9, HIGH); delay(500); digitalWrite(relay9, LOW);
        digitalWrite(relay10, HIGH); delay(500); digitalWrite(relay10, LOW);
    }

}

/* Interrupt code */
void INTO {

    if (count == 0) {
        count = 1;
    }
    else {
        count = 0;
    }

}

```