

## 2.5 Equipment testing

Welcome to the chapter dedicated to the essential task of Equipment Testing for the Mobile Robot Construction Kit. In this chapter, we will guide you through the accurate evaluation of critical components such as the RobDuino controlling module, Battery supply, push button, light sensor, IR distance sensor, and i2c LCD. These elements form the core of your robot construction kit, hence ensuring their proper functioning is crucial for the overall performance of your mobile robot.

We'll delve deep into the intricate details of each component, including their functionality, testing parameters, and what you should look for in terms of optimum performance. By the end of this chapter, you'll have a thorough understanding of how to efficiently test and troubleshoot these key components of your mobile robot.

Don't forget, this is a resource you can always return to when you need guidance or a quick refresher. Ensuring that your equipment is working as it should is fundamental to the success of your mobile robot project, so feel free to revisit this chapter as needed. Whether you're a seasoned roboticist or a novice enthusiast, our detailed guide to equipment testing will prove to be a valuable tool in your robotics journey.

### 2.5.1 RobDuino module

1. Na krmilnik Arduino Uno priključite modul [RobDuino](#) in naložite naslednji program:

**Program 1:** Equipment testing.

```

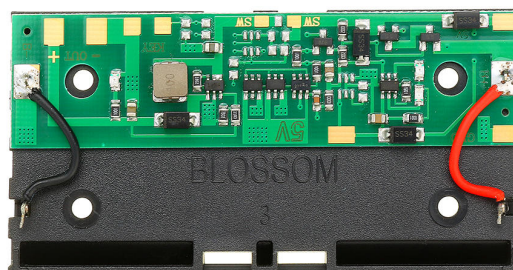
1  bool test_tipk = 1;
2  int l=1;
3
4  void setup() {
5      for (int i = 0; i < 8; i++){
6          pinMode(i, OUTPUT);
7      }
8      pinMode(A4, INPUT_PULLUP);
9      pinMode(A5, INPUT_PULLUP);
10     PORTD=1;
11 }
12
13 void loop() {
14     char tipka_a4_is_pressed = !digitalRead(A4);
15     char tipka_a5_is_pressed = !digitalRead(A5);
16     if (tipka_a4_is_pressed) l = l >> 1;
17     if (tipka_a5_is_pressed) l = l << 1;
18     if (tipka_a4_is_pressed && tipka_a5_is_pressed) test_tipk = !test_tipk
19     ;
20     if (test_tipk){
21         if (l < 1) l = 128;
22         if (l > 255) l = 1;
23         PORTD = l;
24     }else{
25         PORTD = analogRead(A0) >> 2;
26     }
27     delay(200);
28 }

```

2. Nato preverite delovanje obeh tipk (A4 in A5) na modulu in vrednosti izhodnih priključkov D0 .. D7.

**2.5.2 Napajalni modul**

Napajalni modul uporablja 2x Li-ion akumulatorja tipa 18650. Spodnje tiskano vezje je prikazano fig. 1.



**Figure 1:** Napajalni modul.

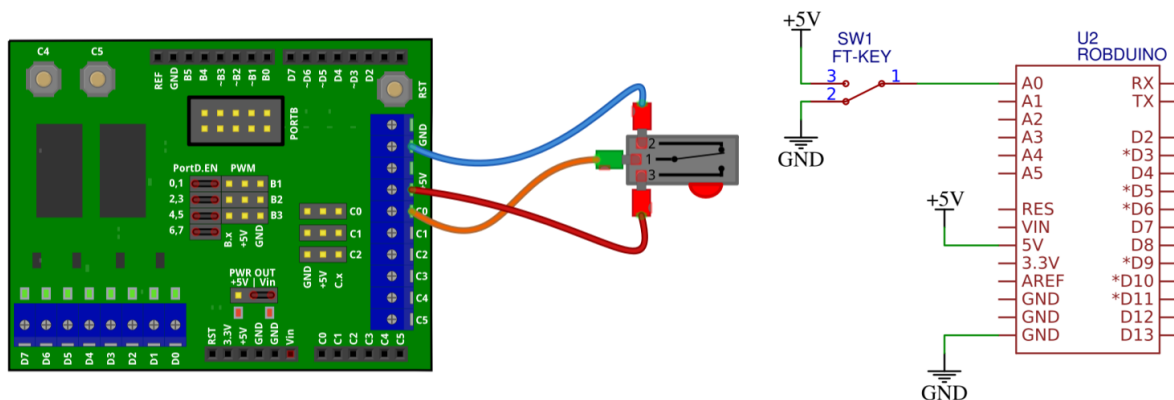
Dodatno smo ga opremili z:

1. 2.5mm jack priključkom za napajanje,
2. 3-pinskim priključkom za napajanje,
3. preklopnim stikalom za izbiranje načina delovanja:
  1. ON - izhod za 9V je kaktiviran
  2. OFF - izključen izhod 9V napajanja in omogočeno je polnenje akumulatorjev preko 3-pinskega priključka (5V).

Pomembno: Pred prvo uporabo moramo ročno aktivirati napajalni modul tako, da povežemo GND na 3-pinskem priključku in NEGATIVNI terminal akumulatorjev.

### 2.5.3 Tipka

1. Priključite stikalo po shemi na fig. 2.



**Figure 2:** Priključitev tipke.

2. Nato naložite naslednji program.

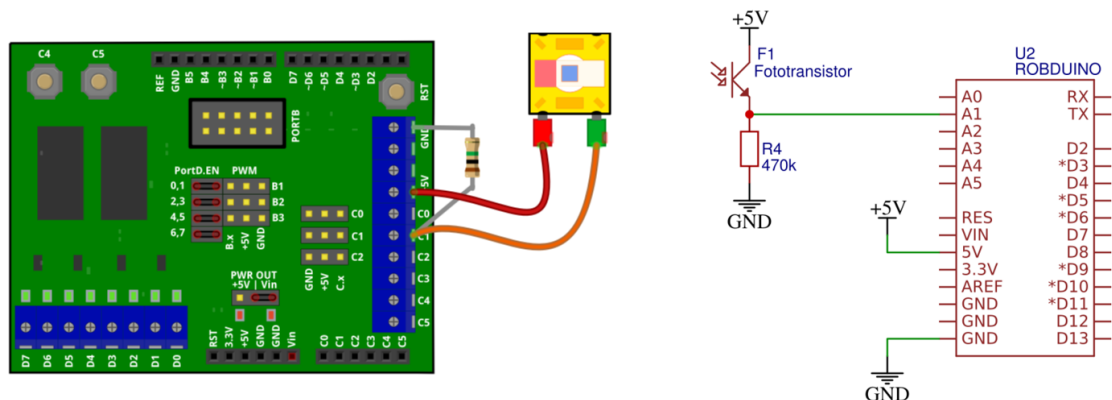
```

1  void setup() {
2      pinMode(A0, INPUT);
3      pinMode(7, OUTPUT);
4  }
5
6  void loop() {
7      char key_a0_is_pressed = digitalRead(A0);
8      if (key_a0_is_pressed){
9          digitalWrite(7, HIGH);
10     } else{
11         digitalWrite(7, LOW);
12     }
13     delay(100);
14 }

```

### 2.5.4 Svetlobni senzor

1. Priključite foto-tranzistor v delilnik napetosti z uporom, kot prikazuje fig. 3.



**Figure 3:** Priključitev foto-tranzistorja kot svetlobni senzor.

- Nato naložite naslednji program in preverite odziv svetlobnega senzorja.

```

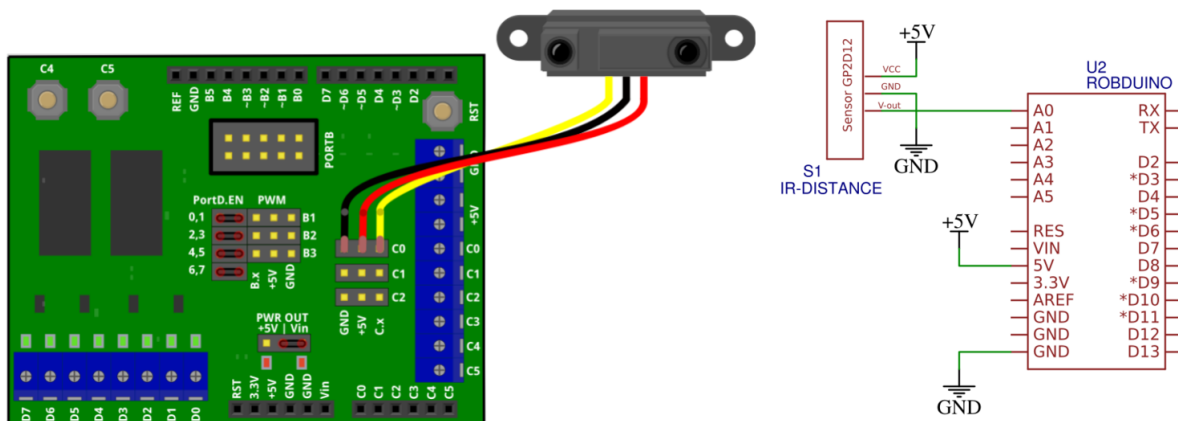
1  void setup() {
2      pinMode(A1, INPUT);
3      Serial.begin(9600);
4  }
5
6  void loop() {
7      int light_senzor_value = analogRead(A1);
8      Serial.println(light_senzor_value);
9      delay(100);
10 }

```

- Odziv senzorja spremljajte v oknu serijske komunikacije.

### 2.5.5 IR senzor razdalje

- IR senzor razdalje priključite na tri-pinski priključek kot je prikazano na fig. 4.



**Figure 4:** Priključitev IR senzorja razdalje.

2. Delovanje senzorja preskusite z naslednjim programom, njegov odziv pa spremljajte v oknu za serijsko komunikacijo.

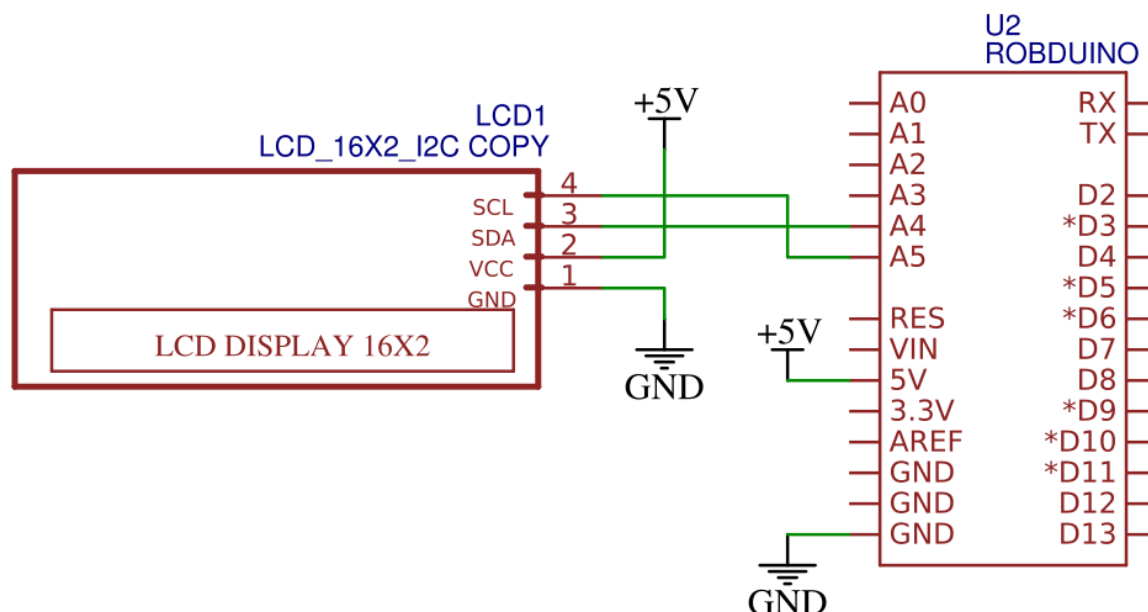
```

1  void setup() {
2      pinMode(A0, INPUT);
3      Serial.begin(9600);
4  }
5
6  void loop() {
7      int distance_senzor_value = analogRead(A0);
8      Serial.println(distance_senzor_value);
9      delay(100);
10 }

```

### 2.5.6 LCD (I2C)

1. Priključite LCD na I2C vodilo kot prikazuje



**Figure 5:** Povezava LCD na I2C vodilo krmilnika.

2. Priskrbite si knjižnico `LiquidCrystal-I2C` iz naslova:  
<https://www.arduino.cc/reference/en/libraries/liquidcrystal-i2c/>
3. Knjižnico dodajte v Arduino IDE okolje tako, da dodate ZIP datoteko v :  
`Sketch >> Include Library >> Add .ZIP Library`
4. V VSC in PlatformIO vtičniku si lahko knjižnico naložite tako, da v terminalno okno vpišete ukaz  
`pio lib install "marcoschwartz/LiquidCrystal_I2C@1.1.4"`
5. Nato preskusite naslednji program:

```
1  #include <Wire.h>
2  #include <LiquidCrystal_I2C.h>
3  LiquidCrystal_I2C Lcd(0x27, 16, 2);
4
5  void setup() {
6      Lcd.init();
7
8      Lcd.clear();
9      Lcd.backlight();
10
11     Lcd.setCursor(3,0);
12     Lcd.print("Hello");
13     Lcd.setCursor(6,1);
14     Lcd.print("World");
15 }
16
17 void loop() {
18 }
```

Če niste prepričani kateri i2c naslov uporablja naprava na LCD-ju le tega lahko preverite s programom **I2C scanner** (<https://playground.arduino.cc/Main/I2cScanner/>). Običajno I2C LCD-ji, ki jih naredijo kitajski proizvajalci uporabljajo I2C naslov 0x27, 0x3F ali manj pogosto 0x38.