Arduino for Schools

G V V Sharma*

Abstract—The objective of this manual is to introduce beginners to computers and programming through an electronics game.

1 Components

Component	Value	Quantity
Breadboard		1
Resistor	$\geq 220\Omega$	1
Arduino	Uno	1
Seven Segment Display	Common Anode	1
Jumper Wires		20

TABLE 1.0

2 Display Control through Hardware

2.1 Powering the Display

The breadboard can be divided into 5 segments. In each of the green segements, the pins are internally connected so as to have the same voltage. Similarly, in the central segments, the pins in each column are internally connected in the same fashion as the blue columns.

*The author is with the Department of Electrical Engineering, Indian Institute of Technology, Hyderabad 502285 India e-mail: gadepall@iith.ac.in. All content in this manual is released under GNU GPL. Free and open source.

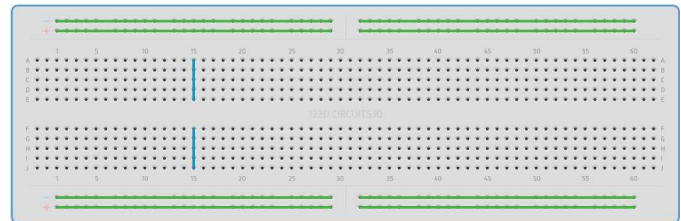**Problem 2.1.** Plug the display to the breadboard in Fig. 2.1

Fig. 2.1

The seven segment display in Fig. 2.2 has eight pins, a, b, c, d, e, f, g and *dot* that take an active LOW input, i.e. the LED will glow only if the input is connected to ground. Each of these pins is connected to an LED segment. The *dot* pin is reserved for the · LED.

Problem 2.2. Connect one end of the resistor to the COM pin of the display and the other end to an extreme pin of the breadboard.

The Arduino Uno has some ground pins, analog input pins A0-A3 and digital pins D1-D13 that can be used for both input as well as output. It also has two power pins that can generate $3.3V$ and $5V$. In the following exercises, only the GND, $5V$ and digital pins will be used.

Problem 2.3. Connect the 5V pin of the arduino to an extreme pin that is in the same segment as the 1K resistor pin.

Problem 2.4. Connect the GND pin of the arduino to the opposite extreme pin of the breadboard

Problem 2.5. Connect the Arduino to the computer.

Problem 2.6. Connect the *dot* pin of the display to a pin in the same segment as the GND pin. What do you observe?
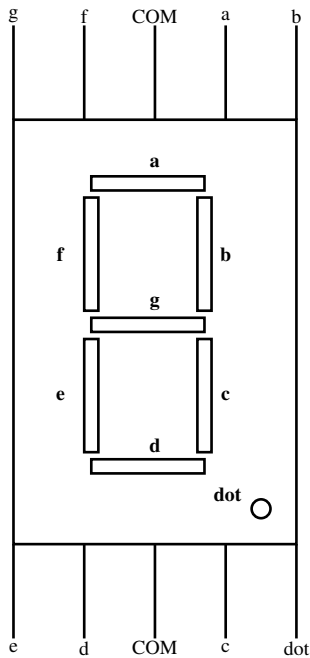
Fig. 2.2

2.2 Controlling the Display

Fig. 2.9 explains how to get decimal digits using the seven segment display.

Problem 2.7. Generate the number 1 on the display by connecting only the pins *b* and *c* to GND.

Problem 2.8. Repeat the above exercise to generate the number 2 on the display.

Problem 2.9. Table 2.9 summarizes the process of generating the decimal digits. 0 means connecting to ground and 1 means not connecting. Complete Table 2.9 for all numbers between 0-9.

a	b	c	d	e	f	g	decimal
1	0	0	1	1	1	1	1
0	0	1	0	0	1	0	2

TABLE 2.9

Problem 2.10. Now generate all numbers between 0-9 on the display using the above table.

3 Display Control through Arduino Software

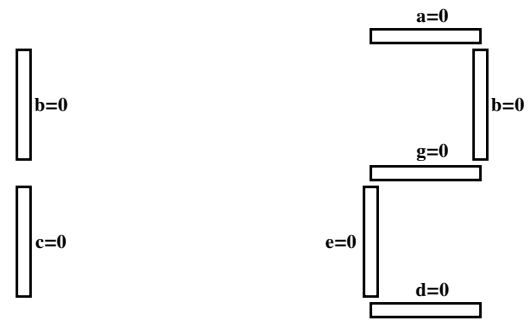**Problem 3.1.** Connect the $a - g$ pins of the display to the pins D2-D8 of the Arduino.

Fig. 2.9

Problem 3.2. Open the arduino software. Check if the ports show Arduino Uno and click the appropriate button.

Problem 3.3. Type the following code and execute. What do you observe?

```
void sevenseg(int a, int b, int c,
   int d, int e, int f, int g)
{
  digitalWrite(2, a);
  digitalWrite(3, b);
  digitalWrite(4, c);
  digitalWrite(5, d);
  digitalWrite(6, e);
  digitalWrite(7, f);
  digitalWrite(8, g);
}
void setup()
{
    pinMode(2, OUTPUT);
    pinMode(3, OUTPUT);
    pinMode(4, OUTPUT);
    pinMode(5, OUTPUT);
    pinMode(6, OUTPUT);
    pinMode(7, OUTPUT);
    pinMode(8, OUTPUT);
}
void loop()
{
sevenseg(1,0,0,1,1,1,1);
}
```

Problem 3.4. Now generate the numbers 0-9 by modifying the above program.

Problem 3.5. Suitably modify the above program to obtain a decade counter.