

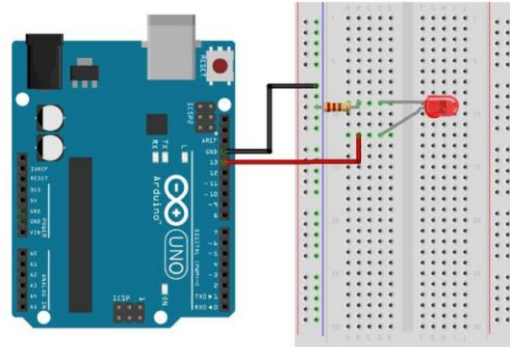
Basic Coding of Arduino Uno - Blinking LED:

Description:

This circuit is a simple blinking LED that turns on and off every 1 second.

Materials:

1 Arduino Uno R3
1 Breadboard
1 Pushbutton
1 10 k Ω Resistor
1 400 Ω Resistor
1 LED
Wires



Codes:

```
void setup() {  
  pinMode(11, OUTPUT);  
}  
  
void loop() {  
  digitalWrite(11,HIGH);  
  delay(1000);  
  digitalWrite(11,LOW);  
  delay(1000);  
}
```

Explanation on codes:

```
void setup() {  
  pinMode(11, OUTPUT);  
}  
  
pinMode(11, OUTPUT);
```

11: the Arduino pin number to set the mode of.

OUTPUT: mode of the pin that can be: INPUT, OUTPUT, or INPUT_PULLUP.

- This sets up the pin 11 as an OUTPUT pin because the led is our output and it is attached to pin 11.
- Meaning that the Arduino is controlling “writing” to a pin instead of reading from it.

```
void loop() {  
  digitalWrite(11,HIGH);  
  delay(1000);  
  digitalWrite(11,LOW);  
  delay(1000);  
}
```

digitalWrite(11,HIGH);

11: the Arduino pin number.

HIGH: digitally write on the pin. It can be either HIGH or LOW.

- These lines are where the action is.
- We start by writing HIGH out on the pin connected to the pin 11 which will turn the LED on. (HIGH means putting 5V out on the pin. The other choice is LOW which means putting 0V out on the pin.)
- We then call delay() which delays the number of milliseconds (1/1000th of a second) sent to it. Since we send the number 1000, it will delay for 1 second.
- We then turn the LED off by writing LOW out on the pin. We delay for 1000 milliseconds (1 second) This will continue until power is removed from the Arduino.

Remember:

If you are having any problems with the projects we did, make sure the following has been checked.

1. Verify the LED is actually functional. Use a 3v coin cell battery and connect the LONG leg of the LED to the (+) and SHORT leg to the (-) of the battery.
2. Verify the correct leg of the LED is connected properly. LONG leg to positive and SHORT leg to negative.
3. Make sure the Arduino IDE shows the correct board. Go to Tools > Board then select Arduino Uno.
4. Make sure the Arduino IDE shows the correct port. Go to Tools > Port then select the port that says Arduino.
5. Verify all component connections are secure with the Arduino board and breadboard.
6. Check your codes if there are any errors.