

Programming IoT System

Internet of Things

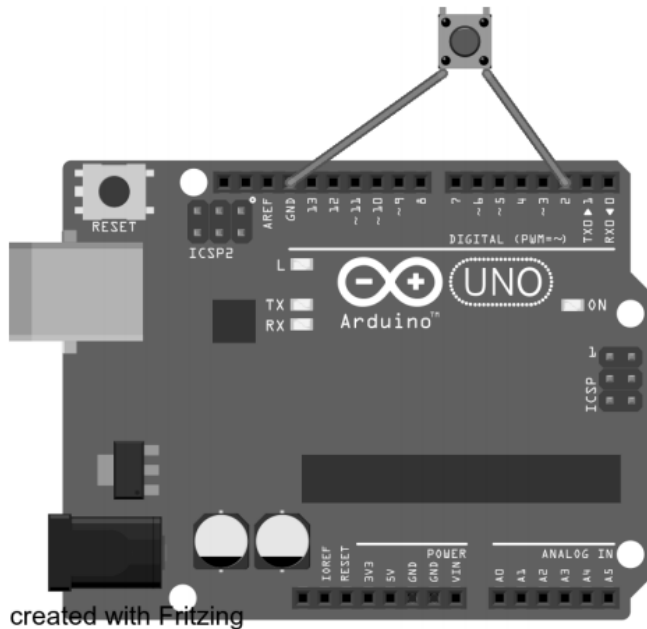
Rahul Shandilya

Digital Write : Blinking LED

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

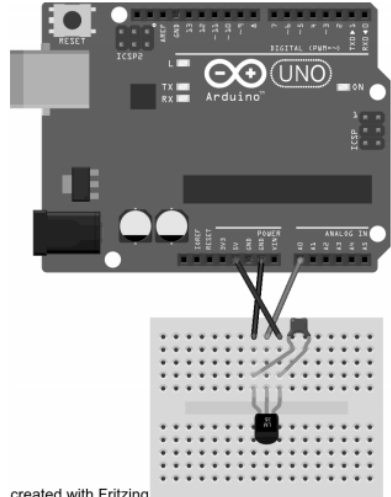
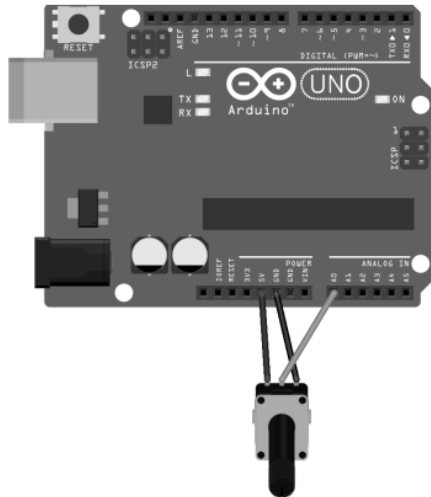
void loop()
{
  digitalWrite(LED_BUILTIN, HIGH); // set the LED on
  delay(2000);                      // wait for two seconds
  digitalWrite(LED_BUILTIN, LOW);  // set the LED off
  delay(2000);                      // wait for two seconds
}
```

Digital Read : Switching LED



```
void setup() {  
  pinMode(2,INPUT_PULLUP);    // button, default=HIGH  
  pinMode(13,OUTPUT);         // LED  
}  
void loop() {  
  if (digitalRead(2)==LOW) {  
    digitalWrite(13,HIGH);  
  } else {  
    digitalWrite(13,LOW);  
  }  
  delay(10);  
}
```

Analog Read : Reading Sensor Value

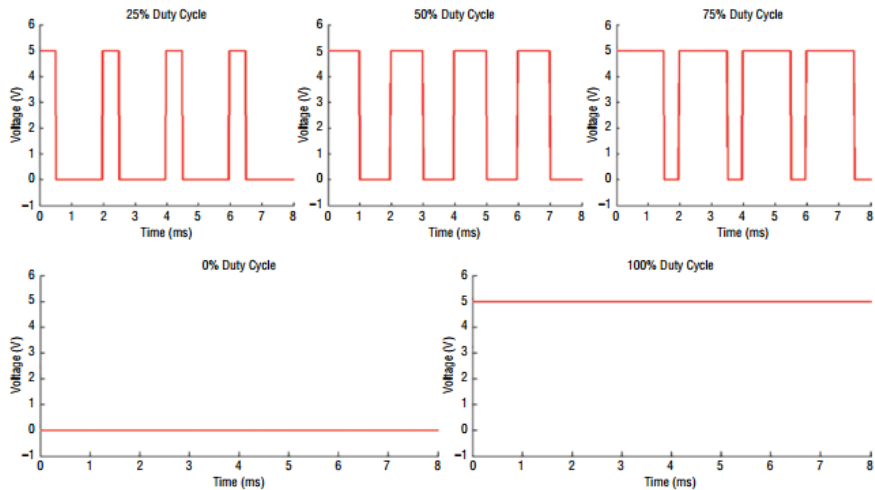


```
int inp,val;
void setup() {
  Serial.begin(9600); // baud rate
}
void loop() {
  if (Serial.available()) {
    inp=Serial.read(); // read serial line
    if (inp=='s') {
      val=analogRead(0); // read analog
      Serial.print("A0= "); // and report back
      Serial.println(val);
    } else if (inp=='t') {
      val=analogRead(1);
      Serial.print("A1= "); Serial.println(val);
    } else {
      Serial.println("unknown command");
    }
  }
  delay(50); // wait 50 ms
}
```

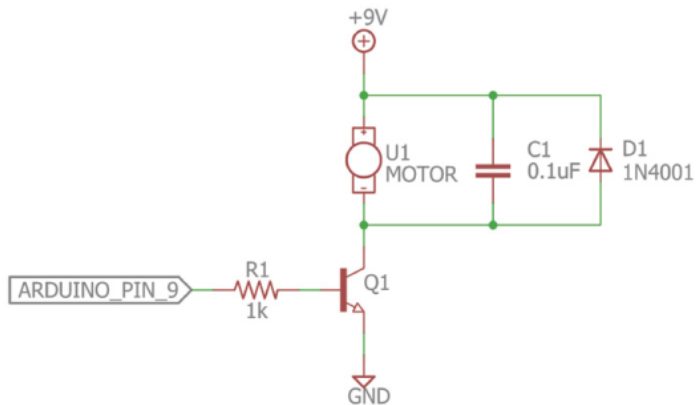
Analog Write : Fading LED

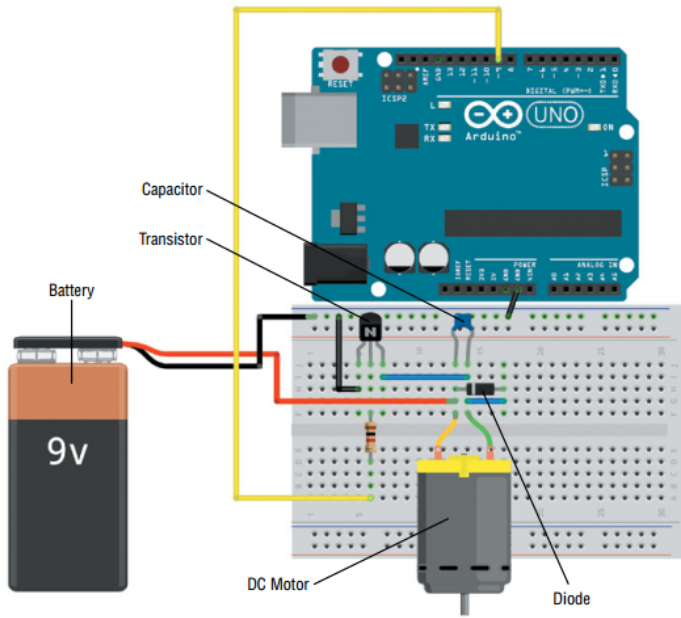
```
const int LED=9;    // Define LED for Pin 9
void setup()
{
  pinMode (LED, OUTPUT);    // Set the LED pin as an output
}

void loop()
{
  for (int i=0; i<256; i++)
  {
    analogWrite(LED, i);
    delay(10);
  }
  for (int i=255; i>=0; i--)
  {
    analogWrite(LED, i);
    delay(10);
  }
}
```



Motor





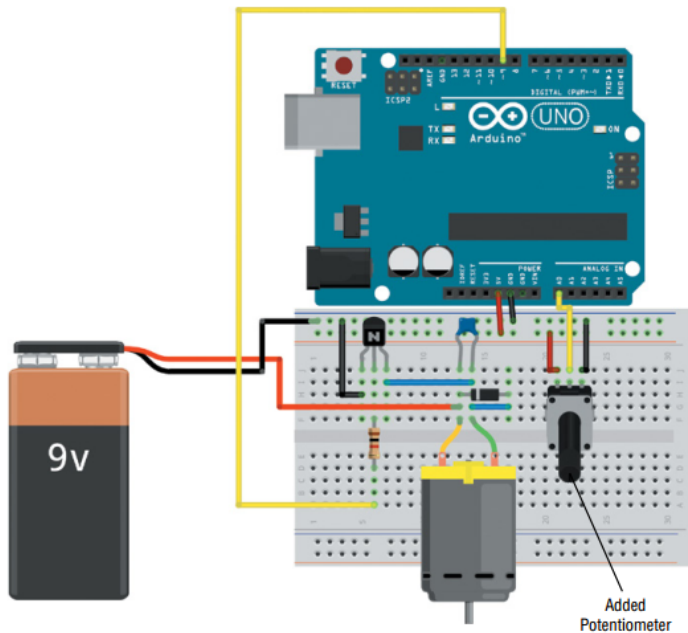
```
//Simple Motor Speed Control Program

const int MOTOR=9;    //Motor on Digital Pin 9

void setup()
{
    pinMode (MOTOR, OUTPUT);
}

void loop()
{
    for (int i=0; i<256; i++)
    {
        analogWrite(MOTOR, i);
        delay(10);
    }
    delay(2000);
    for (int i=255; i>=0; i--)
    {
        analogWrite(MOTOR, i);
        delay(10);
    }
    delay(2000);
}
```

Controlling Motor using Potentiometer



//Motor Speed Control with a Pot

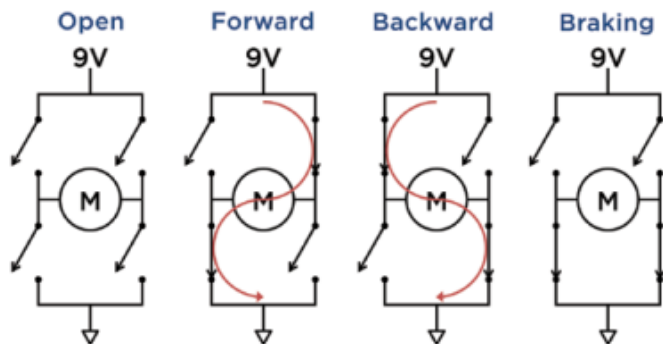
```
const int MOTOR=9; //Motor on Digital Pin 9  
const int POT=0; //POT on Analog Pin 0
```

```
int val = 0;
```

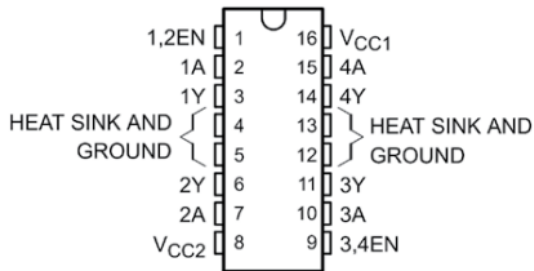
```
void setup()  
{  
    pinMode (MOTOR, OUTPUT);  
}
```

```
void loop()  
{  
    val = analogRead(POT);  
    val = map(val, 0, 1023, 0, 255);  
    analogWrite(MOTOR, val);  
}
```

Driving Motor using H-bridge



H-bridge



FUNCTION TABLE
(each driver)

INPUTS†		OUTPUT Y
A	EN	
H	H	H
L	H	L
X	L	Z

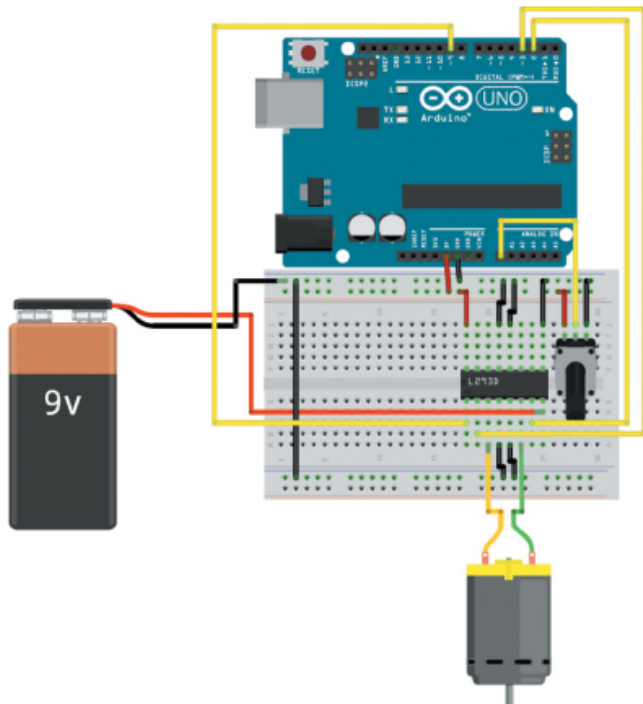
H = high-level, L = low-level

X = irrelevant

Z = high-impedance (off)

† In the thermal shutdown mode, the output is in a high-impedance state regardless of the input levels.

Motor using H-bridge




```
//Motor goes forward at given rate (from 0-255)
void forward (int rate)
{
    digitalWrite(EN, LOW);
    digitalWrite(MC1, HIGH);
    digitalWrite(MC2, LOW);
    analogWrite(EN, rate);
}

//Motor goes backward at given rate (from 0-255)
void reverse (int rate)
{
    digitalWrite(EN, LOW);
    digitalWrite(MC1, LOW);
    digitalWrite(MC2, HIGH);
    analogWrite(EN, rate);
}

//Stops motor
void brake ()
{
    digitalWrite(EN, LOW);
    digitalWrite(MC1, LOW);
    digitalWrite(MC2, LOW);
    digitalWrite(EN, HIGH);
}
```

```
void loop()  
{  
    val = analogRead(P0T);  
  
    //go forward  
    if (val > 562)  
    {  
        velocity = map(val, 563, 1023, 0, 255);  
        forward(velocity);  
    }  
  
    //go backward  
    else if (val < 462)  
    {  
        velocity = map(val, 461, 0, 0, 255);  
        reverse(velocity);  
    }  
  
    //brake  
    else  
    {  
        brake();  
    }  
}
```

```
//H-bridge Motor Control
const int EN=9;    //Half Bridge 1 Enable
const int MC1=3;   //Motor Control 1
const int MC2=2;   //Motor Control 2
const int POT=0;   //POT on Analog Pin 0

int val = 0;       //for storing the reading from the POT
int velocity = 0;  //For storing the desired velocity (from 0-255)

void setup()
{
    pinMode(EN, OUTPUT);
    pinMode(MC1, OUTPUT);
    pinMode(MC2, OUTPUT);
    brake(); //Initialize with motor stopped
}

void loop()
{
    val = analogRead(POT);

    //go forward
    if (val > 562)
    {
        velocity = map(val, 563, 1023, 0, 255);
        forward(velocity);
    }

    //go backward
    else if (val < 462)
```

```
{
    velocity = map(val, 461, 0, 0, 255);
    reverse(velocity);
}

//brake
else
{
    brake();
}
}

//Motor goes forward at given rate (from 0-255)
void forward (int rate)
{
    digitalWrite(EN, LOW);
    digitalWrite(MC1, HIGH);
    digitalWrite(MC2, LOW);
    analogWrite(EN, rate);
}
```

```
//Motor goes backward at given rate (from 0-255)
void reverse (int rate)
{
    digitalWrite(EN, LOW);
    digitalWrite(MC1, LOW);
    digitalWrite(MC2, HIGH);
    analogWrite(EN, rate);
}

//Stops motor
void brake ()
{
    digitalWrite(EN, LOW);
    digitalWrite(MC1, LOW);
    digitalWrite(MC2, LOW);
    digitalWrite(EN, HIGH);
}
```