

Experiment-01:

/*

AnalogReadSerial

Reads an analog input on pin 0, prints the result to the Serial Monitor.

Graphical representation is available using Serial Plotter (Tools > Serial Plotter menu).

Attach the center pin of a potentiometer to pin A0, and the outside pins to +5V and ground.

This example code is in the public domain.

<https://www.arduino.cc/en/Tutorial/BuiltInExamples/AnalogReadSerial>

*/

// the setup routine runs once when you press reset:

void setup() {

// initialize serial communication at 9600 bits per second:

Serial.begin(9600);

}

// the loop routine runs over and over again forever:

void loop() {

// read the input on analog pin 0:

int sensorValue = analogRead(A0);

// print out the value you read:

Serial.println(sensorValue);

delay(1); // delay in between reads for stability

}

Experiment-02:

/*

DigitalReadSerial

Reads a digital input on pin 2, prints the result to the Serial Monitor

This example code is in the public domain.

<https://www.arduino.cc/en/Tutorial/BuiltInExamples/DigitalReadSerial>

*/

// digital pin 2 has a pushbutton attached to it. Give it a name:

int pushButton = 2;

// the setup routine runs once when you press reset:

void setup() {

 // initialize serial communication at 9600 bits per second:

 Serial.begin(9600);

 // make the pushbutton's pin an input:

 pinMode(pushButton, INPUT);

}

// the loop routine runs over and over again forever:

void loop() {

 // read the input pin:

 int buttonState = digitalRead(pushButton);

 // print out the state of the button:

 Serial.println(buttonState);

 delay(1); // delay in between reads for stability

```
}
```

Experiment-03:

```
/*
```

Blink

Turns an LED on for one second, then off for one second, repeatedly.

Most Arduinos have an on-board LED you can control. On the UNO, MEGA and ZERO it is attached to digital pin 13, on MKR1000 on pin 6. LED_BUILTIN is set to the correct LED pin independent of which board is used.

If you want to know what pin the on-board LED is connected to on your Arduino model, check the Technical Specs of your board at:

<https://www.arduino.cc/en/Main/Products>

modified 8 May 2014

by Scott Fitzgerald

modified 2 Sep 2016

by Arturo Guadalupi

modified 8 Sep 2016

by Colby Newman

This example code is in the public domain.

<https://www.arduino.cc/en/Tutorial/BuiltInExamples/Blink>

```
*/
```

```
// the setup function runs once when you press reset or power the board
```

```
void setup() {
```

```

// initialize digital pin LED_BUILTIN as an output.
pinMode(LED_BUILTIN, OUTPUT);
}

// the loop function runs over and over again forever
void loop() {
  digitalWrite(LED_BUILTIN, HIGH); // turn the LED on (HIGH is the voltage level)
  delay(1000);                    // wait for a second
  digitalWrite(LED_BUILTIN, LOW);  // turn the LED off by making the voltage LOW
  delay(1000);                    // wait for a second
}

```

Experiment-04:

```

/*
Controlling a servo position using a potentiometer (variable resistor)
by Michal Rinott <http://people.interaction-ivrea.it/m.rinott>

```

modified on 8 Nov 2013

by Scott Fitzgerald

<http://www.arduino.cc/en/Tutorial/Knob>

```

*/

```

```

#include <Servo.h>

```

```

Servo myservo; // create servo object to control a servo

```

```

int potpin = A0; // analog pin used to connect the potentiometer

```

```

int val; // variable to read the value from the analog pin

```

```
void setup() {  
  myservo.attach(9); // attaches the servo on pin 9 to the servo object  
}  
  
void loop() {  
  val = analogRead(potpin);      // reads the value of the potentiometer (value between 0 and 1023)  
  val = map(val, 0, 1023, 0, 180); // scale it for use with the servo (value between 0 and 180)  
  myservo.write(val);            // sets the servo position according to the scaled value  
  delay(15);                     // waits for the servo to get there  
}
```

Experiment-05: Control a led pattern with button

Experiment-06: Implement a counter using 7 segment LED and push button switch

Experiment-07: object counter using IR sensor.

Experiment-08: arduino thermometer using LM35 and LCD display