Programs

1) Temperature sensor (lastminuteengineers - lm35 temp)

```
#define sensorPin A0
void setup() {
  Serial.begin(9600);
}
void loop() {
  int reading = analogRead(sensorPin);
  float voltage = reading * (5.0/1024.0);
  float temperatureC = voltage * 100;
  Serial.print("Temperature: ");
  Serial.print(temperatureC);
  Serial.print("\xC2\xB0");
  Serial.print("C | ");
  float temperatureF = (temperatureC * 9.0/5.0) + 32.0;
  Serial.print(temperatureF);
  Serial.print("\xC2\xB0");
  Serial.println("F");
  delay(1000);
}
```

---

2) Fire Sensor

```
int buz_pin = 13;
int flame_sensor_pin = 10;
int flame_pin = HIGH;
void setup() {
pinMode(buz_pin, OUTPUT);
pinMode(flame_sensor_pin, INPUT);
Serial.begin(9600);
}
```

```
void loop() {
flame_pin = digitalRead ( flamesensor_pin);
 if (flame_pin == LOW)
 {
 Serial.println(" Alert: Flame detected");
 digitalwrite(buz_pin, HIGH);
 }
 else
 {
  Serial.println(" No flame detected");
  digitalwrite ( buz_pin, LOW);
 }
}
```

Humidity sensor

```
#include dht.h
#define dht_apin A0
 dht DHT;
  void setup() {
   Serial.begin (9600);
   delay(500);
   Serial.println (" Humidity ");
   delay(1000);
  }
  void loop() {
     DHT.read11(dht_apin);
     Serial.print("current humidity=");
     Serial.print(DHT.humidity);
     Serial.print(" %  temperature = ");
     Serial.print(DHT.temperature);
     Serial.println(" C ");
     delay(5000);
  }
```

Big Sound / Small Sound ckt

```
void setup() {
  Serial.begin(9600);
}
void loop() {
  int sensorValue = AnalogRead(A0);
  Serial.println(SensorValue);
}
```

Touch Sensor:

```
void setup()
{
// pinMode(13, OUTPUT);
  pinMode(08, Input);
  Serial.begin(9600);
}
void loop()
{
  if (digitalRead(08) == HIGH) {
    Serial.println("touched");
    delay(500);
  }
else {
    Serial.println("Sensor is not touched");
    delay(500);
  }
}
```

Tracking Sen

```
void setup() {
  Serial.begin(9600);
}
void loop() {
  Serial.print(digitalRead(8));
  delay(500);
}
```

Mercury tilt Switch

```
void Setup() {
    pinMode(3,OUTPUT)
    pinMode(4, INPUT)
    Serial.begin(9600);
}
void loop() {
    if (digitalRead(4)==1)
    { Serial.print("tilted\n");
      digitalwrite(3, HIGH);
      delay(300);
      digitalwrite(3,LOW);
      delay(300);
    }
    else
    { Serial.print("not tilted");
      delay(300);
    }
}
```

Ball Switch

Same as Mercury

Button :

```
const int BUTTON_PIN = 6;
int lastState = HIGH;
int currentState)

void setup() {
   Serial.begin(9600);
   pinMode(BUTTON_PIN, INPUT_PULLUP);
}

void loop() {
   currentState = digitalRead(BUTTON_PIN);
if ((lastState == low) && (currentState = HIGH))
{  Serial.println("Button is pressed");
   delay(1000);
}
else
{  Serial.println("Button not pressed");
   delay(1000);
}
lastState = currentState)
}
```

RGB Led:

```
int red_light_pin = 11;
int green_light_pin = 10;
int blue_light_pin = 9;
void setup() {
   pinMode(red_light_pin, OUTPUT);
   pinMode(green_light_pin, OUTPUT);
   pinMode(blue_light_pin, OUTPUT);
}
void loop() {
   RGB_color(255,0,0);      //Red
   delay(1000);
   RGB_color(0,255,0);      //Green
   delay(1000);
   RGB_color(0,0,255);      //Blue
   delay(1000);
```

```
RGB_color(255, 255, 125);        //Raspberry
delay(1000);
RGB_color(0, 255, 255);          //Cyan
delay(1000);
RGB_color(255, 0, 255);          //Magneta
delay(1000);
RGB_color(255, 255, 0);          //Yellow
delay(1000);
RGB_color(255, 255, 255);        //White
delay(1000);
}
void RGB_color(int red_light_value, int green_light_value, int blue_light_value)
{
analogWrite(red_light_pin, red_light_value);
analogWrite(green_light_pin, green_light_value);
analogWrite(blue_light_pin, blue_light_value);
}
```

7 segment display — B C D E F G

```
#define segmentA 7 6 5 4 3 2
void show_num(int num)
{
switch(num){
  case 0: digitalWrite(segmentA, LOW);
  1 2 3 4 5 6 7 8 9 default
void setup(){
  pinMode(segmentA, OUTPUT);      B C D PIN
void loop(){
  for int i;
  for(i=0; i<=9; i++)
  {
    show_num(i);
    delay(1000);
  }
  for(i=9; i>=0; i--)
  {
    show_num(i);
    delay(100);
  }
```

Rotary encoder

```
#define encoderOPinA 2
#define encoderOPinB 3
#define encoderOBtn 4
int encoderOPos = 0;
void setup() {
  Serial.begin(9600);
  pinMode(encoderOPinA, INPUT_PULLUP);
  —"— PinB —"—
  —"— OBtn —"—
  attachInterrupt(0, doEncoder, CHANGE);
}
int valRotary, lastValRotary;
void loop() {
  int btn = digitalRead(encoderOBtn);
  Serial.print(btn);
  Serial.print(" ");
  Serial.print(valRotary);
  if (valRotary > lastValRotary)
  {
    Serial.print(" CW");
  }
  if (valRotary) {
    Serial.print(" CCW");
  }
  lastValRotary = valRotary;
  Serial.println(" ");
  delay(250);
}
void doEncoder()
{ if (digitalRead(encoderOPinA) == digitalRead(encoderOPinB))
  { encoderOPos++;
  }
  else { encoderOPos--;
  }
  valRotary = encoderOPos/2.5;
}
```

7-color LED

```
int Led = 4;
void setup()
 { pinMode(LED, OUTPUT);
   Serial.begin(9600);
 }
void loop() {
  Serial.println("LED is On");
   digitalWrite(Led, HIGH);
   delay(10000);
  Serial.println("LED Off");
   digitalWrite(Led, LOW);
   delay(2000);
 }
```

Heart beat

```
void setup() {
   pinMode(A0, INPUT);
   Serial.begin(9600);
 }
void loop() {
   float pulse;
   int sum=0;
  for(int i=0; i<20; i++) {
       sum += analogRead(A0);
   }
  pulse = sum/20000;
  Serial.println(pulse);
   delay(1000);
 }
```