

## 5. LM35

Temperature measurement: The core temperature measures the validity by the thermistor. In type of resistor based on its resistance change in temperature. The temp Based on this the data is converted into the voltage and sent temperature value.



*[Signature]*

```
const int lm35-pin = A1;
```

```
void setup()
```

```
{
```

```
  serial.begin(9600);
```

```
}
```

```
void loop()
```

```
{
```

```
  int tmp_adc_val;
```

```
  float tmp_val;
```

```
  tmp_adc_val = analogRead(lm35-pin) /* Read Temperature */
```

```
  tmp_val = (tmp_adc_val * 4.88); /* convert adc value to equivalent voltage */
```

```
  tmp_val = (tmp_val / 10); /* LM35 gives output of 10mV/°C */
```

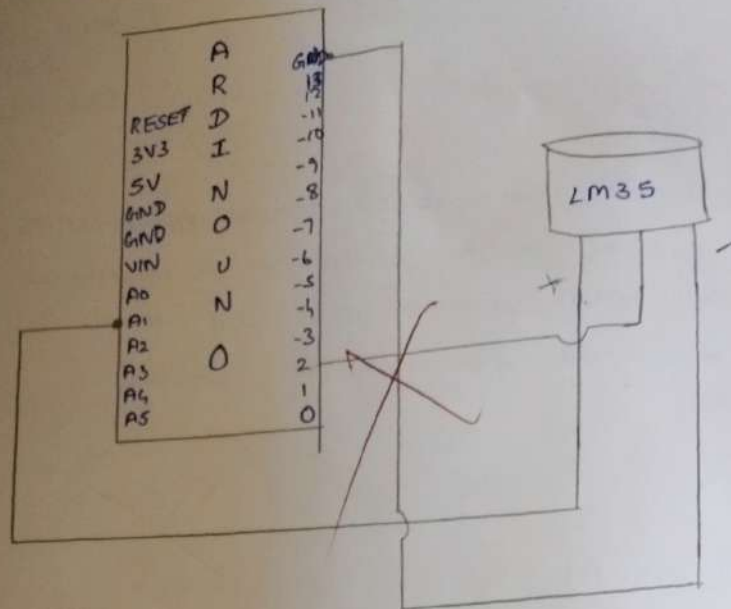
```
  serial.print("Temperature=");
```

```
  serial.print(tmp_val);
```

```
  serial.print("Degree Celsius\n");
```

```
  delay(1000);
```

```
}
```



# 1. PIR

```
const int PIR_SENSOR_OUTPUT_PIN = 4;
```

```
int warm-up;
```

```
void setup() {
```

```
  pinMode(PIR_SENSOR_OUTPUT_PIN, Input);
```

```
  serial.begin(9600); //define baud rate for serial communication
```

~~Serial~~

```
  delay(200000);
```

// power on warm up delay

```
}
```

```
void loop() {
```

~~// power on warm up~~

```
{
```

```
  int sensor-output;
```

```
  sensor-output = digitalRead(PIR_SENSOR_OUTPUT_PIN);
```

```
  if (sensor-output == LOW) {
```

```
    if (warm-up == 1) {
```

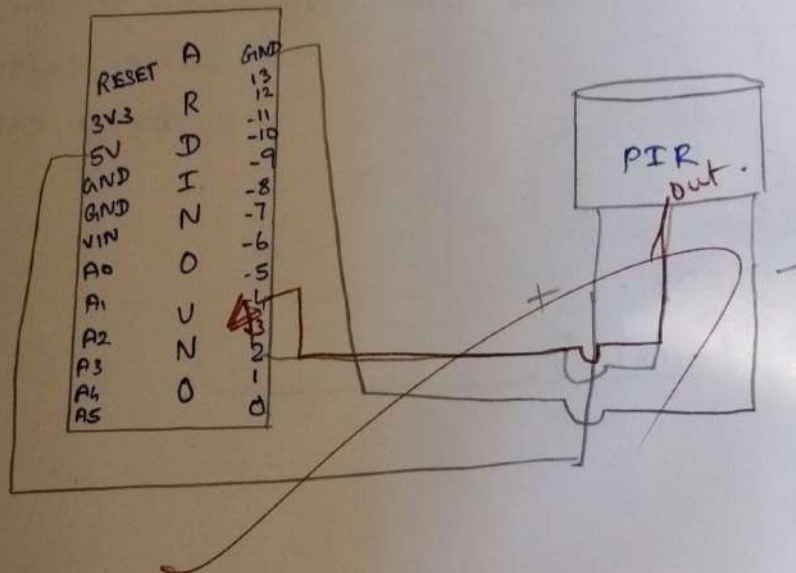
```
      serial.print("Warming up\n");
```

```
      warm-up = 0;
```

```

delay(2000);
}
serial.print("No object in signal");
delay(1000);
}
else
{
serial.print("Object detected");
warm-up=1;
delay(1000);
}
}
}

```





7. Neo 6M GPS

```
#include <TinyGPS++.h>
#include <SoftwareSerial.h>
static const int RXPIN=4, TXPIN=3;
static const uint32_t GPSPBaud=9600;
```

```
TinyGPSPlus gps;
```

```
// TinyGPS++ obj
// The serial connection to the GPS
// device software
SS(RXPIN, TXPIN);
```

```
void setup{
```

```
  Serial.begin(9600);
```

```
  SS.begin(GPSPBaud); }
```

```
void loop{
```

```
// This sketch display info every time a new sentence
correctly encoded
```

```
while (SS.available() > 0) {
  gps.encode(SS.read());
```

```
  if (gps.location.isUpdated())
```

```
{
```

```
  Serial.print("Latitude:");
```

```
  Serial.print(gps.location.lat(), 6);
```