



MALINENI LAKSHMAIAH WOMEN'S ENGINEERING COLLEGE (AUTONOMOUS)

Accredited by "NBA" & " NAAC A+ Grade | Approved by AICTE, New Delhi & Affiliated to JNTUK, Kakinada
Pulladigunta(V), Vatticherukuru(M), Guntur(Dt), A.P.



IOT WORKSHOP

Day-2

ELECTRONIC COMMUNICATION & ENGINEERING[ECE]

Team members:-

E. Hima Bindu(24KE5A0406)

Ch. Alekhyaa Gayathri(24KE5A0404)

D. Thrisha(24KE5A0405)

K. Priya Darshini(23KE1A0450)

G. Siva Parvathi(23KE1A0431)

FROM:ECE-A.....

Team Name:

SMART CREATORS

ACTIVITY - 6

Q. .Reading gas sensors values displaying the serial monitor.

➤ Components:-

*ESP32



*GAS SENSOR



*FEMALE TO FEMALE CONNECTORS



*USB CABLE



➤ Connections:-

- Connect AO pin of the gas sensor to the ESP32 34th pin.
- Connect Ground pin of the gas sensor to the ESP32 ground pin.
- Connect VCC of the gas sensor to the ESP32 to vin(v5).

➤ PROGRAM:-

```
Int gas = 34;  
Void setup() {  
    pinMode (gas , INPUT) ;  
    Serial . begin (9600);  
}  
Void loop () {  
    Int g = analogRead (gas) ;  
    Serial . println (g) ;  
}
```

Output:

- ✓ After writing the code we have to compile code.
- ✓ After that connect the usb cable laptop to ESP32. We have to dump the code on the ESP32.
- ✓ Next connect all connections.
- ✓ After that we have to connect again usb cable to the laptop to ESP32.

- ✓ Then observe the output on serial monitor the initial values of the gas sensor will be high when the gas sensor not detect any harmfull gases output is shown below

```

Activity:6 | Arduino 1.8.19
File Edit Sketch Tools Help
Activity:6
Activity:6
int gas = 34;
void setup() {
pinMode(gas,INPUT);
Serial.begin(9600);
// put your setup code here, to run once
}
void loop() {
int g = analogRead(gas);
Serial.println(g);
// put your main code here, to run repeatedly
}

1
2057
2033
2025
2033
2032
2028
2027
2023
2027
2033
2026
2024
2021
2030
2029

Autoscroll Show timestamp
Newline 9600 baud Clear output

Read of data verified.
Compressed 8192 bytes to 47...
Writing at 0x00000000... (100 %)
Write 8192 bytes (47 compressed) at 0x00000000 in 0.1 seconds (effective 1045.6 kbit/s)...
Read of data verified.
Compressed 8192 bytes to 161285...
Writing at 0x00010000... (10 %)
Writing at 0x00010050... (20 %)
Writing at 0x00010100... (30 %)
Writing at 0x00020050... (40 %)
Writing at 0x00020100... (50 %)
Writing at 0x00030050... (60 %)
Writing at 0x00030100... (70 %)
Writing at 0x00040050... (80 %)
Writing at 0x00040100... (90 %)
Writing at 0x00050050... (100 %)
Write 287782 bytes (161285 compressed) at 0x00010000 in 3.0 seconds (effective 773.9 kbit/s)...
Read of data verified.

Low voltage...
Hard resetting via RST pin...

```

- ✓ Then observe the output on serial monitor.the final values are low, when the gas sensor detects some harmfull gasses the output is shown below.

```

Activity:6 | Arduino 1.8.19
File Edit Sketch Tools Help
Activity:6
Activity:6
int gas = 34;
void setup() {
pinMode(gas,INPUT);
Serial.begin(9600);
// put your setup code here, to run once
}
void loop() {
int g = analogRead(gas);
Serial.println(g);
// put your main code here, to run repeatedly
}

1
1093
1071
1098
1097
1102
1114
1110
110
1098
1104
1106
1103
1122
1102
1093
110

Autoscroll Show timestamp
Newline 9600 baud Clear output

Read of data verified.
Compressed 8192 bytes to 47...
Writing at 0x00000000... (100 %)
Write 8192 bytes (47 compressed) at 0x00000000 in 0.1 seconds (effective 1045.6 kbit/s)...
Read of data verified.
Compressed 8192 bytes to 161285...
Writing at 0x00010000... (10 %)
Writing at 0x00010050... (20 %)
Writing at 0x00010100... (30 %)
Writing at 0x00020050... (40 %)
Writing at 0x00020100... (50 %)
Writing at 0x00030050... (60 %)
Writing at 0x00030100... (70 %)
Writing at 0x00040050... (80 %)
Writing at 0x00040100... (90 %)
Writing at 0x00050050... (100 %)
Write 287782 bytes (161285 compressed) at 0x00010000 in 3.0 seconds (effective 773.9 kbit/s)...
Read of data verified.

Low voltage...
Hard resetting via RST pin...

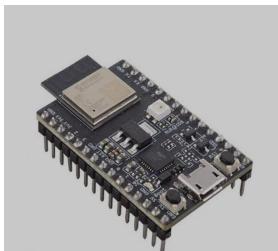
```

ACTIVITY – 7

Q.Checking the conditions for gas sensors.

Components:-

*ESP32



*USB CABLE



*GAS SENSOR



*FEMALE TO FEMALE WIRES



Connections:-

- Connect pin AO of the gas sensor to the ESP32 34th pin.
- Connect Ground pin of the gas sensor to the ESP32 ground pin.
- Connect VCC of the gas sensor to the ESP32 to vin(v5).

PROGRAM:-

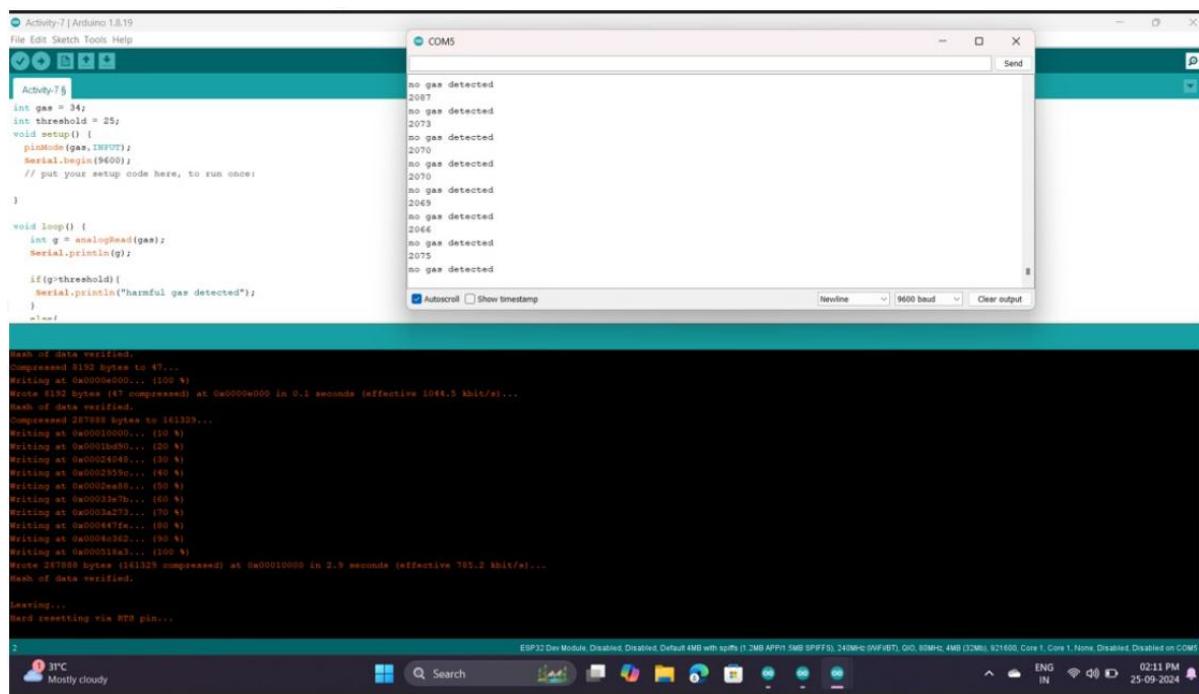
```

int gas = 34;
int threshold = 25;
void setup() {
    pinMode(gas,INPUT);
    Serial.begin(9600);
}
void loop() {
    int g = analogRead(gas);
    Serial.println(g);
    if(g>threshold){
        Serial.println("harmful gas detected");
    }
    else{
        Serial.println("no gas detected");
    }
}

```

Output:

- ✓ After writing the code we have to complie code.
 - ✓ After that connect the usb cable laptop to ESP32. We have to dump the code on the ESP32.
 - ✓ Next connect all connections.
 - ✓ After that we have to connect again usb cable to the laptop to ESP32.
 - ✓ Then observe the output on serial monitor when the gas sensor detects no harmfull gas then prints output on serial monitor with condition output is shown below



- ✓ Then observe the output on serial monitor when the gas sensor detects harmful gases then prints the output on serial monitor with condition output is shown below

Activity-7 | Arduino 1.8.19

File Edit Sketch Tools Help

Activity-7

```
int threshold = 2033;
void setup() {
  pinMode(gas, INPUT);
  Serial.begin(9600);
  // put your setup code here, to run once:
}

void loop() {
  int g = analogRead(gas);
  Serial.println(g);

  if(g>threshold){
    Serial.println("harmful gas detected");
  }
  else{
    Serial.println("no gas detected");
  }
}
```

1361
harmful gas detected
1363
harmful gas detected
1360
harmful gas detected
1366
harmful gas detected
1354
harmful gas detected
1354
harmful gas detected
1351
harmful gas detected
1345
h

Autoscroll Show timestamp Newline 9600 baud Clear

Last of data verified.
Compressed 8192 bytes to 47...
Writing at 0x0000e000... (100 %)
Update 8192 bytes (47 compressed) at 0x0000e000 in 0.1 seconds (effective 1023.6 kbit/s)...
Last of data verified.
Compressed 287888 bytes to 161330...
Writing at 0x00010000... (10 %)
Writing at 0x0001bd02... (20 %)
Writing at 0x00024044... (30 %)
Writing at 0x00029559... (40 %)
Writing at 0x0002ea86... (50 %)
Writing at 0x00033e76... (60 %)
Writing at 0x0003a271... (70 %)
Writing at 0x000447fe... (80 %)
Writing at 0x0004c361... (90 %)
Writing at 0x000518a1... (100 %)
Update 287888 bytes (161330 compressed) at 0x00010000 in 2.0 seconds (effective 775.4 kbit/s)...
Last of data verified.
Leaving...
Hard resetting via RST pin...

14 31°C
Search

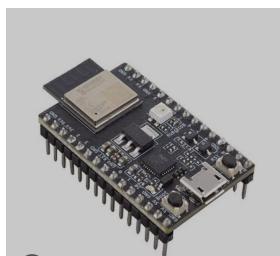
ESP32 Dev Module, Disabled, Disabled, Default 4MB with splits (1.2MB APP/1.5MB SPIFFS), 240MHz (WiFi/BT), QIO, 80MHz, 4MB (32MHz), SD

ACTIVITY – 8

Q. Reading IR sensor values and displaying in serial monitor

Components:-

*ESP32



*USB CABLE



*FEMALE TO FEMALE WIRES



*IR SENSOR



Connections:-

- Connect the Do pins of the ir sensor to the ESP32 12th pin.
- Connect the Ground pin of the ir sensor to the ESP32 ground pin.
- Connect VCC pin of the ir sensor to the ESP32 to vin(v5)

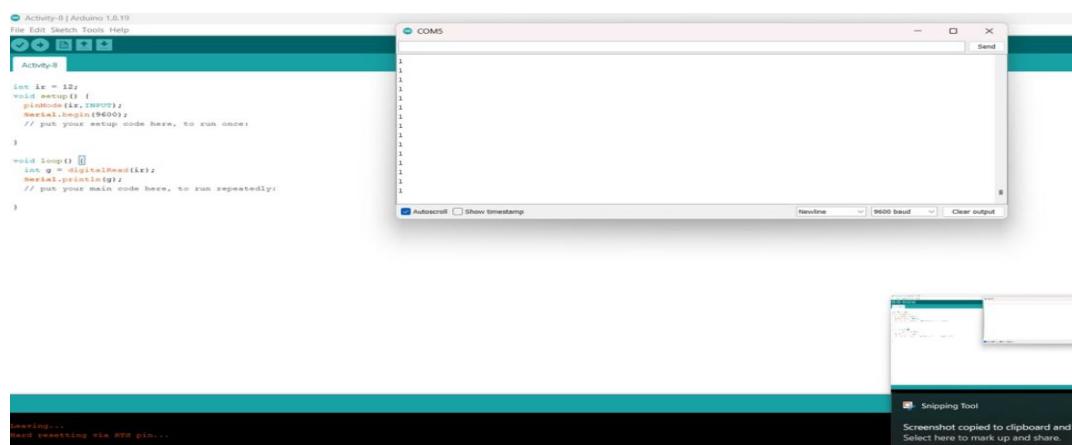
PROGRAM:-

```
Int ir =12 ;  
  
Void setup () {  
    pinMode(ir,INPUT);  
    Serial . begin(9600);  
}  
  
Void loop (){  
    Int g = digital Read (ir) ;  
    Serial .print ln(g);  
}
```

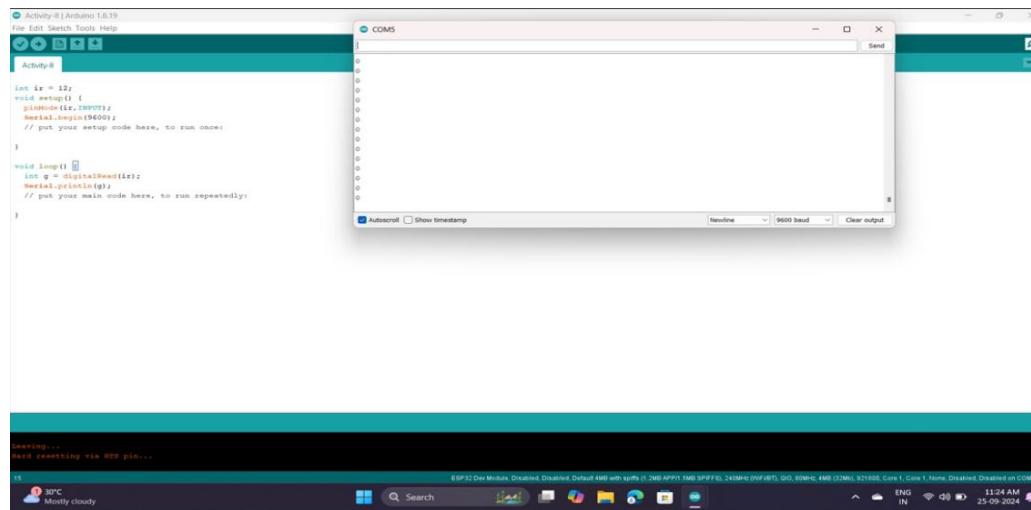
Output:

- ✓ After writing the code we have to compile code.

- ✓ After that connect the usb cable laptop to ESP32. We have to dump the code on the ESP32.
 - ✓ Next connect all connections.
 - ✓ After that we have to connect again usb cable to the laptop to ESP32.
 - ✓ Then observe the output on serial monitor when the ir sensor detects no object then it shows the output is 1 output is shown below



- ✓ Then observe the output on serial monitor when the ir sensor detects object then it shows the output is 0 output is shown below



ACTIVITY – 9

Q.Check the conditions for IR sensor.

Components:-

*ESP32



*USB CABLE



*FEMALE TO FEMALE WIRES



*IR SENSOR



Connections:-

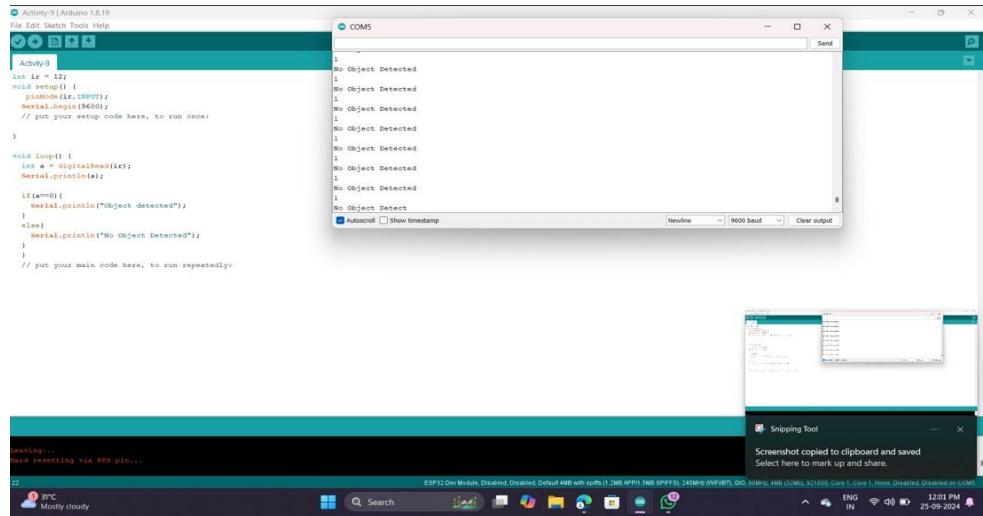
- Connect the Do pins of the ir sensor to the ESP32 12th pin.
- Connect the Ground pin of the ir sensor to the ESP32 ground pin.
- Connect VCC pin of the ir sensor to the ESP32 to vin(v5)

PROGRAM:-

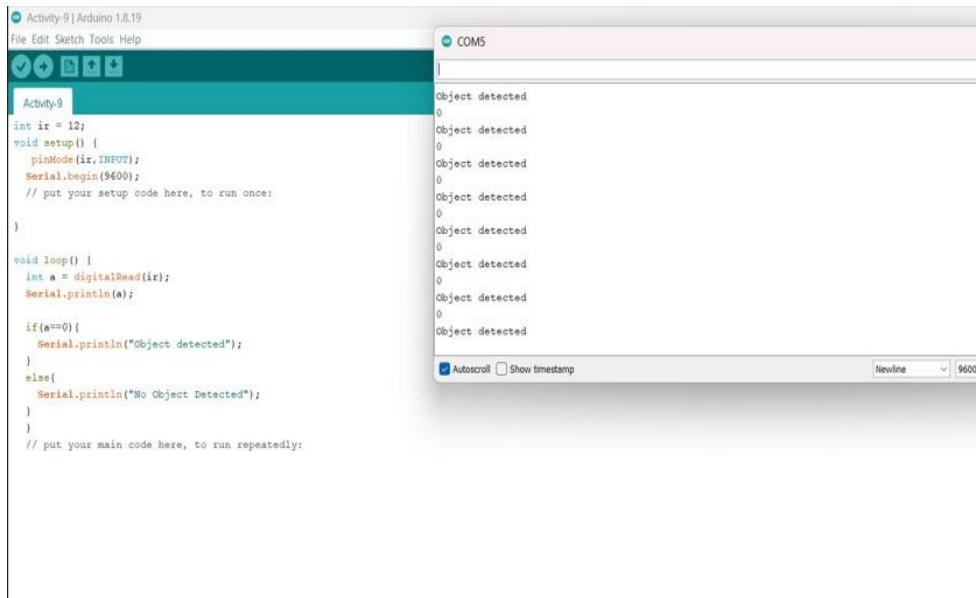
```
int ir= 12;  
  
void setup() {  
    // put your setup code here, to run once:  
    pinMode(ir,INPUT);  
    Serial.begin(9600);  
}
```

Output:

- ✓ After writing the code we have to compile code.
- ✓ After that connect the usb cable laptop to ESP32. We have to dump the code on the ESP32.
- ✓ Next connect all connections.
- ✓ After that we have to connect again usb cable to the laptop to ESP32.
- ✓ Then observe the output on serial monitor when the ir sensor detects no object then it shows the output is 1 and prints the statement no object detected output is shown below



- ✓ Then observe the output on serial monitor when the ir sensor detects object then it shows the output is 0 and prints the statement object detected output is shown below



ACTIVITY – 10

Q.Reading soil moisture sensor values and displayed in serial monitor.

Components:-

*ESP32

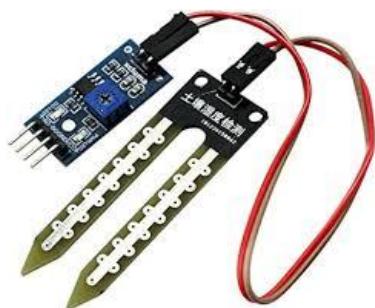


*USB CABLE



*FISCHER SENSOR

*SOIL MOISTURE SENSOR



*FEMALE TO FEMALE WIRES



Connections:-

- Connect the Ao pins of the fish sensor to the ESP32 34th pin.
- Connect the both soil moisture sensor and fish sensor.
- Connect the Ground pin of the soil moisture sensor to the ESP32 ground pin.
- Connect the VCC pin of the soil moisture sensor to the ESP32 to vin(v5).

PROGRAM:-

```
int soil = 34;

void setup() {
    pinMode(soil,INPUT);
    Serial.begin(9600);

    // put your setup code here, to run once:

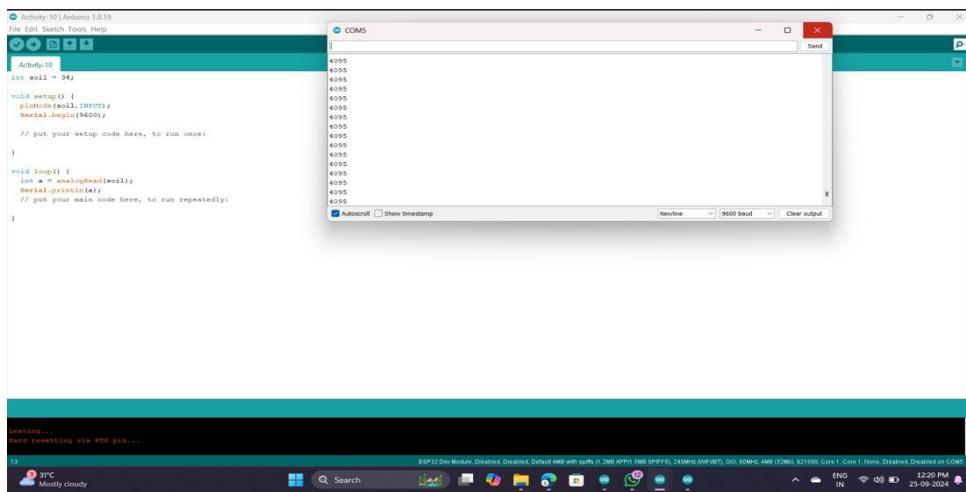
}

void loop() {
    int a = analogRead(soil);
    Serial.println(a);
    // put your main code here, to run repeatedly:
```

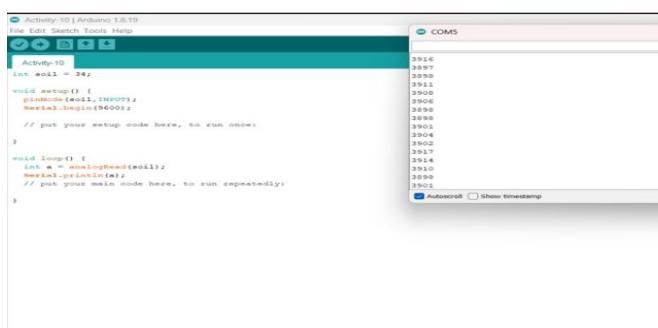
}

Output:

- ✓ After writing the code we have to complie code.
- ✓ After that connect the usb cable laptop to ESP32. We have to dump the code on the ESP32.
- ✓ Next connect all connections.
- ✓ After that we have to connect again usb cable to the laptop to ESP32.
- ✓ Then observe the output on serial monitor when the soil moisture sensor detects when the soil content is not have moisture then it shows the output values on serial monitor is shown below



- ✓ Then observe the output on serial monitor when the soil moisture sensor detects when the soil content have moisture then it shows the output values on serial monitor is shown below

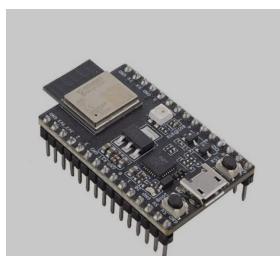


ACTIVITY – 11

Q.Checking the conditions for gas sensors.

Components:-

*ESP32

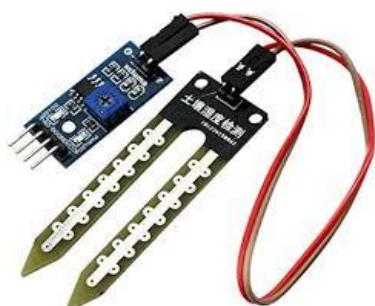


*USB CABLE



*FISH SENSOR

*SOIL MOISTUR SENSOR



*FEMALE TO FEMALE WIRES



Connections:-

- Connect the Ao pins of the fish sensor to the ESP32 34th pin.
- Connect the both soil moisture sensor and fish sensor.
- Connect the Ground pin of the soil moisture sensor to the ESP32 ground pin.
- Connect the VCC pin of the soil moisture sensor to the ESP32 to vin(v5).

PROGRAM:-

```
int soil = 34;  
int threshold = 4095;  
  
void setup() {  
    pinMode(soil,INPUT);  
    Serial.begin(9600);  
    // put your setup code here, to run once:  
  
}  
  
  
void loop() {  
    int a = analogRead(soil);  
    Serial.println(a);  
    if(a<threshold){  
        Serial.println("moisture Detected");  
    }  
}
```

```

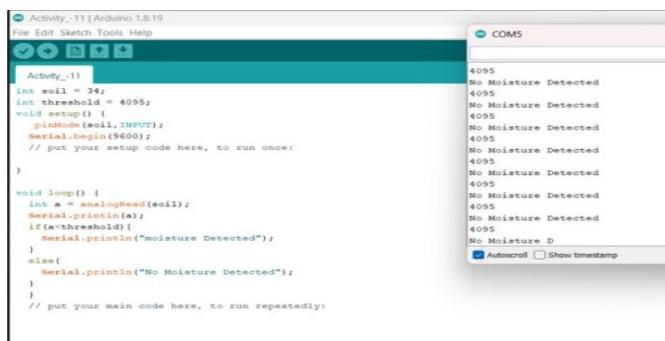
else{
    Serial.println("No Moisture Detected");
}

}

```

Output:

- ✓ After writing the code we have to complie code.
- ✓ After that connect the usb cable laptop to ESP32. We have to dump the code on the ESP32.
- ✓ Next connect all connections.
- ✓ After that we have to connect again usb cable to the laptop to ESP32.
- ✓ Then observe the output on serial monitor when the soil moisture sensor detects when the soil content is not have moisture then it prints with statement it shows the output values on serial monitor is shown below



- ✓ Then observe the output on serial monitor when the soil moisture sensor detects when the soil content have moisture then it prints with statement it shows the output values on serial monitor is shown below

The screenshot shows the Arduino IDE interface. The left pane displays the code for 'Activity_11' (File: Activity_11.ino). The code reads analog input from pin A0, checks it against a threshold, and prints 'moisture Detected' or 'No Moisture Detected' to the Serial Monitor. The right pane shows the Serial Monitor window titled 'COM5'. It displays the serial data being sent from the Arduino. The bottom status bar indicates the build and upload progress.

```
Activity_11
int soil = 34;
int threshold = 4095;
void setup() {
  pinMode(A0,INPUT);
  Serial.begin(9600);
  // put your setup code here, to run once:
}

void loop() {
  int a = analogRead(soil);
  Serial.println(a);
  if(a>threshold) {
    Serial.println("moisture Detected");
  }
  else{
    Serial.println("No Moisture Detected");
  }
  // put your main code here, to run repeatedly:
}

leaving...
and resetting via RST pin...
```

Serial Plotter

COM5

Send

2544
moisture Detected
2545
moisture Detected
2544
moisture Detected
2559
moisture Detected
255
moisture Detected
2544
moisture Detected
2549
moisture Detected
2547
moisture Det

Autoscroll Show timestamp

Newline 9600 baud Clear output

ES2017 Document Generated by Eclipse IDE with Eclipse IDE 2020-09-15 10:38:46.578Z (1605321126578ms) 5142000 Class 1. Class 1. Name: Default_PropertiesForCPE

ACTIVITY – 12

Q.PRINT THE VALUES OF SOIL MOISTURE SENSOR.

Components:-

*ESP32



*USB CABLE



*ULTRASONIC SENSOR



*FEMALE TO FEMALE WIRES



PROGRAM:-

```
int trig = 14;  
int echo = 4;  
  
void setup() {  
    pinMode(trig,OUTPUT);  
    pinMode(echo,INPUT);  
    Serial.begin(9600);  
    // put your setup code here, to run once:  
}  
  
void loop() {  
    digitalWrite(trig,0);  
    delayMicroseconds(2);  
    digitalWrite(trig,1);  
    delayMicroseconds(10);  
    digitalWrite(trig,0);  
  
    float duration =pulseIn(echo,1);  
    duration=duration/2;
```

```

float distance=(0.034)*duration;

Serial.println(distance);

}

```

Connections:-

- *Connect the trig pin to 14th pin of ESP32
- * Connect the ECO pin to the 4 th pin of ESP32
- * Connect ground to ground
- * Connect Vcc to the Power terminal.

Output:

- ✓ After writing the code we have to complie code.
- ✓ After that connect the usb cable laptop to ESP32. We have to dump the code on the ESP32.
- ✓ Next connect all connections.
- ✓ After that we have to connect again usb cable to the laptop to ESP32.
- ✓ Then observe the output on serial monitor when the ultra sonic sensor detects when the object is far to the sensor then the output values on serial monitor is shown below



The screenshot shows the Arduino IDE interface with the following details:

- Code (Activity-12.ino):**

```

#include <Arduino.h>
int trig = 14;
int echo = 4;

void setup() {
    // initialize digital pin 14 as an output
    pinMode(trig, OUTPUT);
    // initialize digital pin 4 as an input
    pinMode(echo, INPUT);
    // Set the serial port to 9600 bps
    Serial.begin(9600);
    // put your setup code here, to run once
}

void loop() {
    //digitalWrite(trig, 0);
    //delayMicroseconds(2);
    //digitalWrite(trig, 1);
    //delayMicroseconds(10);
    //digitalWrite(trig, 0);

    float duration = pulseIn(echo, 1);
    float distance = (0.034) * duration;
    Serial.println(distance);
}

// put your main code here, to run repeatedly

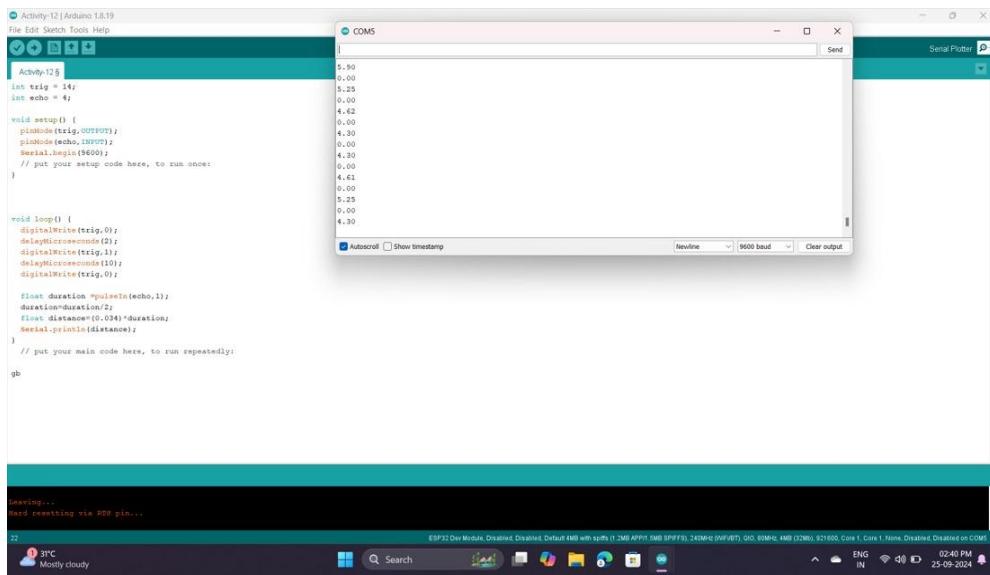
```
- Serial Monitor (COMS):**

```

27.20
0.00
27.20
0.00
27.20
0.00
27.20
0.00
27.20
0.00
26.62
0.00
25.75
0.00
25.49
0.00
25.48

```

- ✓ Then observe the output on serial monitor when the ultra sonic sensor detects when the object is near to the sensor then the output values on serial monitor is shown below



The screenshot shows the Arduino IDE interface with the following details:

- Code Area:** Displays the Arduino sketch named "Activity12.ino". The code initializes pins 12 and 13 for trig and echo respectively, sets up the serial port at 9600 baud, and defines a constant for the duration of the pulse sent by the trig pin.
- Serial Monitor:** A window titled "COM5" shows the output of the sketch. The data is timestamped and includes values such as 5.90, 0.00, 4.25, 0.00, 4.62, 0.00, 4.30, 0.00, 4.20, 0.00, 4.61, 0.00, 4.25, 0.00, 4.30. Below the monitor are controls for "Autoscroll", "Show timestamp", "Newline", "9600 baud", and "Clear output".
- Bottom Status Bar:** Shows the sketch name "Activity12.ino", the board "ESP32 Dev Module", the processor "Draagon", and the frequency "400 MHz with 128 MB APP RAM (8 MB SRAM)". It also displays the current time "02:40 PM" and date "25-09-2024".

Thank You!