

Kammavari Sangham ® 1952
K.S.Group of Institution

K S INSTITUTE OF TECHNOLOGY

(Affiliated to VTU, Approved by AICTE & Accredited by NAAC)
14, Kanakapura Road, Raghuvanahalli, Bangalore, Karnataka-560109



Department of Computer Science and Engineering
INTERNET OF THINGS LABORATORY
(18CS81)

Prepared By:

Mr. Krishna Gudi
Assistant Professor
Department of CSE, KSIT

Program 1: Transmit a string using UART

```
void setup() {  
  // put your setup code here, to run once:  
  Serial.begin(9600);  
  
}  
  
void loop() {  
  // put your main code here, to run repeatedly:  
  Serial.println("WELCOME TO CSE DEPARTMENT");  
  delay(1000);  
  
}
```

Connections: - Connect Arduino Mega 2560 to the laptop/PC via cable.

Step 1. Save the following code as a sketch in Arduino IDE

Step 2. Compile the sketch.

Step 3. After successful compilation, check for correct board (in our case it is – Arduino/Genuino Mega or Mega 2560 and select proper COM port.

Step 4. Upload the code onto the board.

Step 5. After successful upload, open serial monitor in Arduino IDE for the output (Baud rates should match).

Program 2: Point-to-Point communication of two Motes over the radio frequency.

```
#define ledPin 12
int data = 0;
void setup()
{
  pinMode(ledPin, OUTPUT);
  digitalWrite(ledPin, LOW);
  Serial.begin(9600); //default baud rate for bt 38400
}

void loop()
{
  if(Serial.available() > 0)
  {
    data = Serial.read(); // Reads the data from the serial port

    if (data == '0')
    {
      digitalWrite(ledPin, LOW); // Turn LED OFF
      Serial.println("LED: OFF");
    }
    else if (data == '1')
    {
      digitalWrite(ledPin, HIGH);
      Serial.println("LED: ON");
    }
  }
}
```

Connections: - Make the connections of HC-05 with Arduino

Step 1. Save the following code as a sketch in Arduino IDE

Step 2. Compile the sketch.

Step 3. After successful compilation, check for correct board (in our case it is – Arduino/Genuino Mega or Mega 2560 and select proper COM port.

Step 4. Upload the code onto the board.

Step 5. After successful upload, check whether the Bluetooth module is working or not. (The easiest way is to see if the RED indicator is blinking).

Install Arduino Bluetooth Controller App in the android-based phone. Now, go to Bluetooth Settings on the phone and pair the phone with the HC-05 Device. The key is 1234. After successfully pairing, open the Bluetooth Controller App and connect with the HC-05 module.

Program 3: Multi-point to single point communication of Motes over the radio frequency-LAN (Subnetting).

```
//FirebaseESP8266.h must be included before ESP8266WiFi.h
#include "FirebaseESP8266.h" // Install Firebase ESP8266 library
#include <ESP8266WiFi.h>
#include <DHT.h> // Install DHT11 Library and Adafruit Unified
Sensor Library

#define FIREBASE_HOST "esp8266-98959-default-
rtadb.firebaseio.com" // the project name
address from firebase id
#define FIREBASE_AUTH
"FL2hGoHImW8a00R403fInHfwhtFbBk5iaZRYNIiX" // the secret
key generated from firebase
#define WIFI_SSID "Gudi_2G"
#define WIFI_PASSWORD "aarush18"

#define DHTPIN D4 // Connect Data pin of DHT to D4
//int led =5; // Connect LED to D5

#define DHTTYPE DHT11
DHT dht(DHTPIN, DHTTYPE);

//Define FirebaseESP8266 data object
FirebaseData firebaseData;
//FirebaseData ledData;

FirebaseJson json;

void setup()
{
    Serial.begin(115200);

    dht.begin();
    // pinMode(led,OUTPUT);

    WiFi.begin(WIFI_SSID, WIFI_PASSWORD);
    Serial.print("Connecting to Wi-Fi");
    while (WiFi.status() != WL_CONNECTED)
    {
        Serial.print(".");
        delay(300);
    }
}
```

```
}

Serial.println();
Serial.print("Connected with IP: ");
Serial.println(WiFi.localIP());
Serial.println();

Firebase.begin(FIREBASE_HOST, FIREBASE_AUTH);
Firebase.reconnectWiFi(true);

}

void sensorUpdate(){

float h = dht.readHumidity();
// Read temperature as Celsius (the default)
float t = dht.readTemperature();
// Read temperature as Fahrenheit (isFahrenheit = true)
float f = dht.readTemperature(true);

// Check if any reads failed
if (isnan(h) || isnan(t) || isnan(f)) {
  Serial.println(F("Failed to read from DHT sensor!"));
  return;
}

Serial.print(F("Humidity: "));
Serial.print(h);
Serial.print(F("%  Temperature: "));
Serial.print(t);
Serial.print(F("C  ,"));
Serial.print(f);
Serial.println(F("F  "));

if (Firebase.setFloat(firebaseData, "/FirebaseIOT/temperature",
t))
{
  Serial.println("PASSED");
}
else
{
  Serial.println("FAILED");
}

if (Firebase.setFloat(firebaseData, "/FirebaseIOT/humidity", h))
{
```

```
    Serial.println("PASSED");
  }

  else
  {
    Serial.println("FAILED");
  }
}
void loop() {
  sensorUpdate();
}
```

Step 1. Save the following code as a sketch in Arduino IDE.

Step 2.

- After adding necessary libraries.
- Go to : <https://firebase.google.com> and goto sign in for signing up with gmail login.
- Save the sketch and then compile.

Step 3. After successful compilation, check for correct board (in our case it is NodeMCU 1.0 ESP 12E Module) and select proper COM port.

Step 4. Upload the code onto the board.

Step 5. After successful upload, open project in Firebase link and check the real time values of temperature and humidity pushed onto the cloud server.

Program 4: I2C protocol study

```
#include <Wire.h>
#include <LiquidCrystal_I2C.h>

// Set the LCD address to 0x27 for a 16 chars and 2 line display
LiquidCrystal_I2C lcd(0x27, 16, 2);

void setup()
{
    // initialize the LCD
    lcd.begin();

    // Turn on the backlight and print a message.
    lcd.backlight();
    //lcd.print("Hello, world!");
}

void loop()
{
    // Do nothing here...
    lcd.clear();
    lcd.print("Hello World!");
    delay(1000);
    lcd.clear();

    lcd.print("I'm powered by");
    lcd.setCursor(3,1);
    lcd.print("Arduino!");
    delay(3000);
    lcd.clear();

    lcd.print("UNO R3");
    delay(3000);
    lcd.clear();
    lcd.print("Good day!");
    delay(3000);
    lcd.clear();
}
```


Connections: - Connect Arduino Mega 2560 to the laptop/PC via cable and do the connection from 16x2 LCD to arduino.

Step 1. Save the following code as a sketch in Arduino IDE

Step 2. Compile the sketch after adding required library (eg: to locate LiquidCrystal.h, header file)

Step 3. After successful compilation, check for correct board (Arduino/Genuino Mega or Mega 2560 and select proper COM port.)

Step 4. Upload the code onto the board.

Step 5. After successful upload, view the output on the LCD screen.

Program 5: Reading Temperature and Relative Humidity value from the sensor.

```
#include <DHT.h>
#define Type DHT11
int sensePin=2;
DHT HT(sensePin,Type);
float humidity;
float tempC;
int setTime=500;
int dt=1000;
void setup()
{
    // setup code to run once:
    Serial.begin(9600);
    HT.begin();
    delay(setTime);
}

void loop()
{
    // main code to run repeatedly:
    humidity=HT.readHumidity();
    tempC=HT.readTemperature();

    Serial.print("Humidity: ");
    Serial.print(humidity);
    Serial.print(" Temperature: ");
    Serial.print(tempC);
    Serial.print(" C ");
    Serial.println();
    delay(dt);
}
```

Step 1. Save the following code as a sketch in Arduino IDE.

Step 2. Compile the sketch after adding necessary libraries.

Step 3: After successful compilation, check for correct board (Arduino/Genuino Mega or Mega 2560 and select proper COM port).

Step 4. Upload the code onto the board.

Step 5. After successful upload, open serial monitor in Arduino IDE for the output (Baud rates should match).