Experiment :10 -Real-Time Weather Monitoring System Using DHT11 Sensor and SPI TFT LCD Display with Arduino

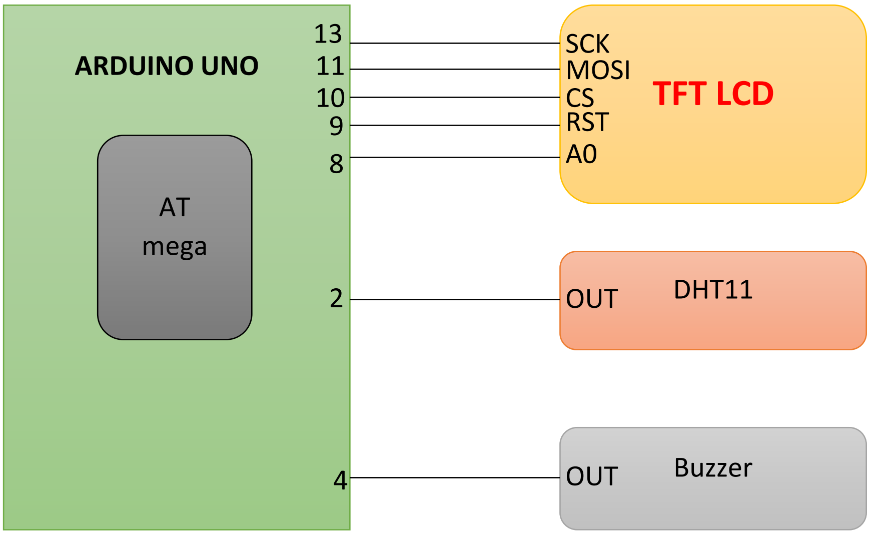
AIM:

To design and implement a real-time weather monitoring system using an Arduino Uno, DHT11 temperature & humidity sensor, SPI TFT LCD display, and a buzzer. The system will display temperature and humidity values on the TFT screen and trigger a buzzer if the temperature exceeds 29°C.

COMPONENTS REQUIRED:

S.No	Component	Quantity
1	Arduino Uno	1
2	DHT11 Sensor	1
3	SPI TFT Display (ST7735)	1
4	Buzzer	1
5	Jumper Wires	As required
6	Breadboard	1
7	Power Supply (5V)	1

CIRCUIT DIAGRAM:



Connections:

Component	Arduino Pin
TFT CS	10
TFT RST	9
TFT DC	8
TFT MOSI	11

TFT SCK	13
DHT11 OUT	2
Buzzer OUT	4
GND (All)	GND
VCC (DHT11, TFT, Buzzer)	5V

Arduino IDE coding:

```
/*
CS → Pin 10
RST → Pin 9
DC → Pin 8
MOSI → Pin 11
SCK → Pin 13
DHT11 OUT → 2
Buzzer OUT → 4
 */
#include <Adafruit_GFX.h>    // Core graphics library
#include <Adafruit_ST7735.h> // Hardware-specific library for ST7735
#include <SPI.h>
#include <DHT.h>             // DHT library for sensor

/* Pin definitions */
#define TFT_CS        10
#define TFT_RST       9
#define TFT_DC        8
#define DHTPIN        2   // DHT11 sensor connected to pin 2
#define DHTTYPE       DHT11 // Define sensor type
#define Buzzer        4
DHT dht(DHTPIN, DHTTYPE);
Adafruit_ST7735 tft = Adafruit_ST7735(TFT_CS, TFT_DC, TFT_RST);

void setup() {
  Serial.begin(9600);
  Serial.println(F("thingZkit_IoT_TFT Test"));
  pinMode(Buzzer, OUTPUT);
  dht.begin(); // Start DHT sensor
  tft.initR(INITR_BLACKTAB);
  Serial.println(F("TFT Initialized"));

  tft.setRotation(3); // Set screen rotation (0, 1, 2, or 3)
  tft.fillScreen(ST77XX_BLACK);
  tft.setTextColor(ST77XX_WHITE);
  tft.setTextSize(2);
}

void loop() {
  float temp = dht.readTemperature();   // Read temperature in
Celsius
  float hum = dht.readHumidity();       // Read humidity

  if (isnan(temp) || isnan(hum)) {
    Serial.println(F("Failed to read from DHT sensor!"));
    return;
  }

  Serial.print(F("Temperature: "));
  Serial.print(temp);
  Serial.print(F("°C  Humidity: "));
  Serial.print(hum);
  Serial.println(F("%"));

  tft.fillScreen(ST77XX_BLACK); // Clear screen
  tft.setCursor(20, 20);
  tft.setTextColor(ST77XX_YELLOW);
  tft.println("DHT11 Sensor");

  tft.setCursor(20, 50);
  tft.setTextColor(ST77XX_WHITE);
  tft.print("Temp: ");
  tft.setTextSize(1);
  tft.print(temp);
  tft.println(" C");

  tft.setCursor(20, 80);
  tft.print("Humidity: ");
  tft.setTextSize(1);
  tft.print(hum);
  tft.println(" %");

  if(temp > 29){
    digitalWrite(Buzzer, HIGH);
  }
```

WORKING:

The real-time weather monitoring system operates through a series of well-defined steps. First, the DHT11 sensor continuously reads the temperature and humidity from the environment. This data is then processed by the Arduino Uno, which receives the sensor readings and prepares them for display. The SPI TFT LCD screen updates in real time, showing the current temperature and humidity values. Additionally, a buzzer alert mechanism is incorporated into the system and if the temperature exceeds 29°C, the buzzer turns on as a warning; otherwise, it remains off. The system executes this process in a continuous loop, refreshing the values every 2 seconds to ensure real-time monitoring of environmental conditions.

OBSERVATION TABLE:

S.No	Temperature (°C)	Humidity (%)	Display Output	Buzzer Status
1	24	45	Temp: 24°C, Humidity: 45%	OFF
2	26	50	Temp: 26°C, Humidity: 50%	OFF
3	29	55	Temp: 29°C, Humidity: 55%	OFF
4	30	58	Temp: 30°C, Humidity: 58%	ON
5	32	60	Temp: 32°C, Humidity: 60%	ON
6	27	52	Temp: 27°C, Humidity: 52%	OFF
7	31	65	Temp: 31°C, Humidity: 65%	ON
8	25	47	Temp: 25°C, Humidity: 47%	OFF

RESULT:

The real-time weather monitoring system was successfully implemented using an Arduino board. The temperature and humidity readings were displayed on the SPI TFT LCD screen. The buzzer was triggered when the temperature exceeded 29°C, indicating a high-temperature alert.