Internship Project
Report

Brief on PCB assembly process and Temperature sensor using STMicrocontrollers

Submitted by

Chetan Moturi
Student Intern
Iljin Electronics

R&D Department
Iljin Electronics
Greater Noida

Winter Internship 2022

Abstract

In this internship report, I am trying to highlight what I have done from doing an internship. In my term as a student intern at Iljin Electronics my main focus or objective was to learn about new technologies and improve my skills in interdisciplinary fields.
I have learnt android development and using STM32CubeIDE for programming ST micro controller for making a temp sensor.

Contents

Chapter 1

Objective

The main objectives of this internship were to:

- To learn about the manufacturing process of Printed Circuit Boards (PCBs) and the various machines used to make them.
- To learn Android development and make android apps using android studio.
- To make a temperature sensor using STM32F103cb micro controller on STM32CubeIDE.

Chapter 2

Introduction

This chapter contains a brief intro of the company, Iljin Electronics, PCB manufacturing, android studio and STM32CubeIDE

2.1 ILJIN ELECTRONICS

ILJIN ELECTRONICS (I) PVT. LTD is a South Korean Professional Electronics Design and Manufacturing Company with uncompromising integrity based at GREATER NOIDA, INDIA which has been actively functioning in India since 2002.
They are mainly involved in the manufacture of Printed Circuit Boards (PCBs) , but also manufacture Emergency lights, LED lighting for Home and Decor, DC and AC power supply and other appliances.

The PCB manufacturing process is explained below,

2.1.1 PCB Manufacturing

Here, at Iljin electronics once the PCBs are designed,etched and drilled they are assembled mainly by three processes, they are elucidated below,

Automatic Insertion

It is based on Through-hole Technology in which leads of the components are inserted through holes on the board and then soldered or glued (depending on requirement) to pads on the opposite side.

Figure 2.2: An Automatic Insertion machine

The insertion mount machine (figure 2.2) used has a rotary table and X-axis and Y-axis positioning system. The insertion mount machine can work in axial or radial mode depending on the requirement.

The components are then soldered; once cooled and solidified the solder holds the components in place and electrically connects them to the board.

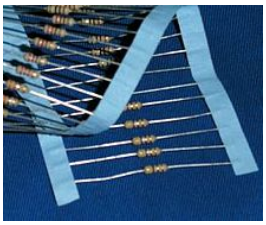


Figure 2.1: resistor strip fed to the machine.

Surface Mount Technology

In surface-mount technology (SMT; figure 2.3), the component is placed on the PCB so that the pins line up with the conductive pads or lands on the surfaces of the PCB; solder paste, which was previously applied to the pads, holds the components in place temporarily; if surface-mount components are applied to both sides of the board, the bottom-side components are glued to the board.

Again the components are soldered to the board. Once soldered the board is then washed to remove any stray solder balls and flux residues, which can short closely placed leads of the components.

Figure 2.3: A SMT machine

The main advantages of SMT are,

- Smaller components.
- Much higher component density (components per unit area) and many more connections per component.
- Components can be placed on both sides of the circuit board.
- Higher density of connections because holes do not block routing space on inner layers, nor on back-side layers if components are mounted on only one side of the PCB.
- Lower initial cost and time of setting up for mass production, using automated equipment.
- Many SMT parts cost less than equivalent through-hole parts.

The disadvantages of SMT are,

- SMDs' solder connections may be damaged by potting compounds going through thermal cycling.
- Many types of SMT component packages cannot be installed in sockets, which provide for easy installation or exchange of components to modify a circuit and easy replacement of failed components. (Virtually all through-hole components can be socketed.)

Manual Insertion

This is basically same as Automatic insertion, the main difference being that components are now manually inserted by workers. The reason why manual insertion is still in practice is because,

Figure 2.4: resistor strip fed to the machine.

- Some components are not able to be adequately inspected by the automated optical inspection equipment and require a technician to verify their placement and touch-up any soldering problems.
- in the case of low-volume production runs, it can sometimes be faster and less expensive to do the assembly by hand.

Chapter 3

Work Done

I have worked on mainly two things in this internship, Android studio and Temperature sensor using STM32F103cb and STM32CubeIDE.

3.1 Android Studio and Development

Android Studio is the new official IDE for Android. Version 1 was developed in Eclipse.

3.1.1 Installation

Android Studio needs to be downloaded from here . Once the .exe file for your windows system (32 bit or 64 bit) has finished downloading, click on it to run it.
In case you don‘t have Java Development Kit (JDK) on your computer, you will be asked to download it from here.
Download and run it to install the latest version of JDK. Then proceed to install Android Studio.
Once installation of Android Studio is complete, you need to download and install the Software Development Kits (SDKs), which can be done by starting the SDK Manager in Android Studio (an icon present in the top right hand corner of the screen).

3.1.2 Project

I made a Tic Tac Toe game as my first android studio project. It is a two player game in which a player chooses between X and O.The goal here is to get three consecutive X or O in a horizontal, vertical, or diagonal direction.

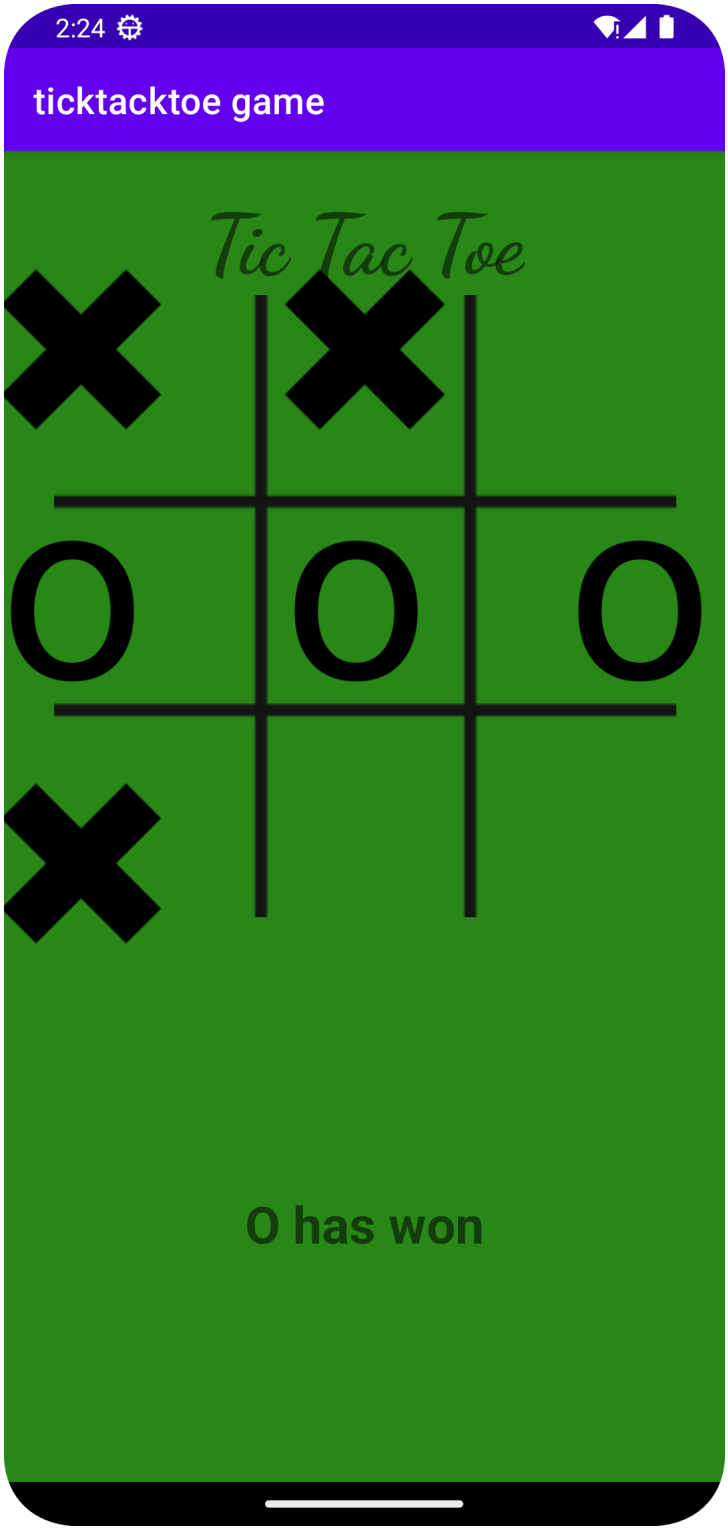


Figure 3.1: Screenshot of game

It is a single activity project. The wining positions are stored in a 2D matrix and compared regularly after every move. The status is shown below.

3.2 Temperature Sensor using STmicrocontroller

The temperature sensor is made using DHT11 sensor and STM32F103C8.

3.2.1 Components Used

- STM32F103C8
- DHT11 Temperature & Humidity Sensor
- 16x2 LCD Display
- IIC/I2C Serial Interface Adapter Module
- Breadboard
- Connecting wires.

3.2.2 DHT11 Sensor

DHT11 is a Temperature and humidity sensor which as the name implies is used to measure the atmospheric temperature and humidity in a particular environment or in a confined closed space.
The sensor could measure the temperature from 0°C to 50°C with an accuracy of 1°C. It is commonly used in controlled environments such as Heat ventilation systems, temperature chambers etc to monitor temperature and take corrective measures.
The sensor is commonly used in monitoring environmental parameters in many applications like Agriculture, Food Industries, Hospitals, Automobile, Weather Stations etc. The measuring range of humidity is from 20% to 90% with an accuracy of 1%.
The specifications of the sensor are,

- Operating Voltage: 3.5V to 5.5V
- Operating current: 0.3mA (measuring) 60uA (standby)
- Temperature Range: 0°C to 50°C
- Humidity Range: 20% to 90%

Figure 3.2: dht11 sensor

- Resolution: Temperature and Humidity both are 16-bit
- Accuracy: ±1°C and ±1%

3.2.3 Preparing the IDE for STM32F103CB

We first need to find the I2C address of the adapter to communicate with the LCD screen.
Follow the below steps to find the I2C address of your LCD I2C interface module,

1. First check the STM32 package for Arduino IDE is installed. If not follow the link Programming your STM32 in ARDUINO IDE.
2. While installing packages for programming STM32 using Arduino IDE by above link the wire library is installed in default.
3. Program for scanning the I2C device connected is present in the examples (In Arduino IDE: Files->Examples->Wire->I2C scanner wire). Before that select the board in Tools->Board->Generic STM32F103C8 Series as shown below.

Now that we know the I2C address we need to download a library for communicating to the LCD display through I2C. After downloading the zip file install I2C LCD library in the Arduino IDE by sketch->import library.This library can also be used with Arduino boards for communicating with I2C LCD display modules.

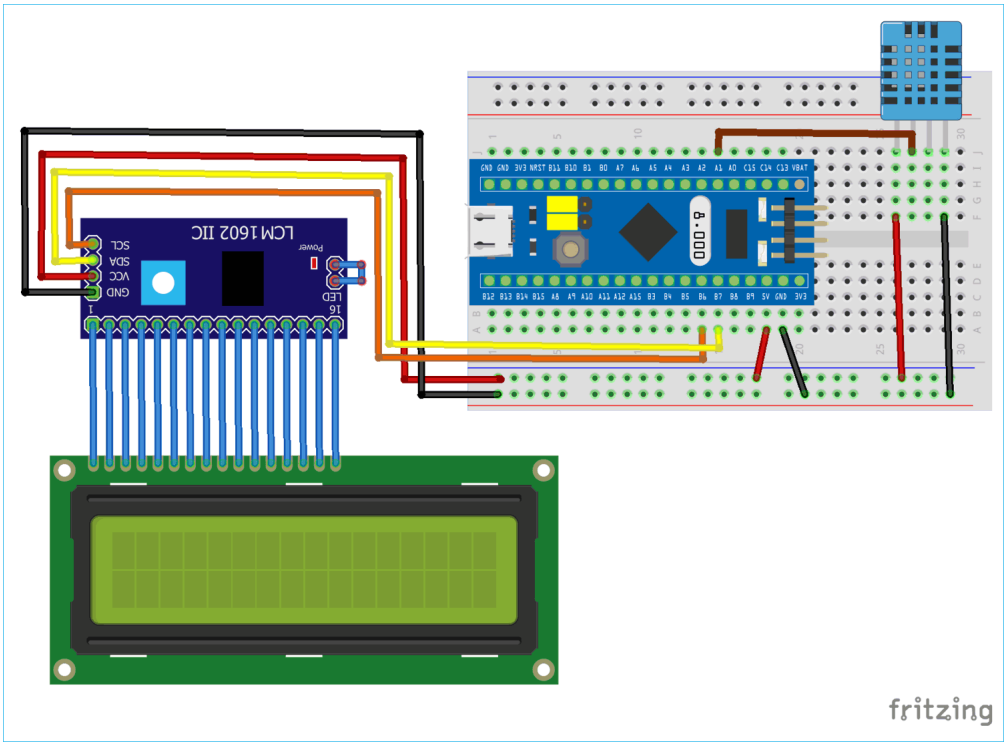


Figure 3.3: Circuit Diagram

Similarly in order to read the serial data from DHT11 sensor we will use the DHT11 library. After downloading, install DHT library in the Arduino IDE by using sketch import library. Again the same library can also be used with Arduino boards.

3.2.4 Code

```
#include <Wire.h>                //Library for using I2C
#include <LiquidCrystal_I2C.h>  //Library for using I2C type LCD display
#include <DHT.h>                //Library for using DHT sensor

#define DHTPIN PA1
#define DHTTYPE DHT11

LiquidCrystal_I2C lcd(0x27, 16, 2);

*/initilize object lcd for  class liquidCrystal_I2C
with I2C address of 0x27 and 16x2 type LCD display.*/

DHT dht(DHTPIN, DHTTYPE);
*/initilize object dht for class DHT with DHT pin
with STM32 and DHT type as DHT11.*/

void setup()
{

  // initialize the LCD
  lcd.begin();
  dht.begin();      //Begins to receive Temperature and humidity values.
  lcd.backlight(); // Turn on the blacklight and print a welcome message.

  lcd.setCursor(0,0);

  lcd.print("CIRCUIT DIGEST");

  lcd.setCursor(0,1);

  lcd.print("DHT11 with STM32");

  delay(3000);
```

```
  lcd.clear();

}

void loop()
{

  float h = dht.readHumidity();       //Gets Humidity value

  float t = dht.readTemperature();    //Gets Temperature value

  lcd.setCursor(0,0);

  lcd.print("Temp: ");

  lcd.print(t);

  lcd.print(" C");

  lcd.setCursor(0,1);

  lcd.print("Humid: ");

  lcd.print(h);

  lcd.print(" %");

}
```

3.2.5 Code Explanation

The Explanation for the code is as follows,

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

Include the required libraries. Include Wire.h library for using I2C in STM32F103C8, LiquidCrystal_I2C.h for using I2C type LCD display and DHT.h for using DHT sensor functions.
Now the pin name of DHT11 (OUT pin) that is connected with the PA1 of STM32F103C8 is defined, also DHTTYPE is defined as DHT11

```
#define DHTPIN PA1
#define DHTTYPE DHT11
```

Next the object lcd for class LiquidCrystal_I2C with I2C address of 0x27 and 16x2 type LCD display is initialized.And also, object dht for class DHT with DHT pin with STM32 and DHT type as DHT11 is initialized.

Next in void setup ():
First initialize the LCD using following,

```
lcd.begin();
```

And then use below statement to begin to receive Temperature and humidity values from DHT11 sensor.

```
dht.begin();
```

Turn on the blacklight and print a welcome message and clear them after three seconds.

```
lcd.backlight();
lcd.setCursor(0,0);
lcd.print("CIRCUIT DIGEST");
lcd.setCursor(0,1);
lcd.print("DHT11 with STM32");
delay(3000);
lcd.clear();
```

Next in the void loop():
The value is received from the DHT11 sensor continuously. In order to get the separate values of temperature and humidity and store it in a variable following statement is used.
To get only the Humidity value,

```
float h = dht.readHumidity();
```

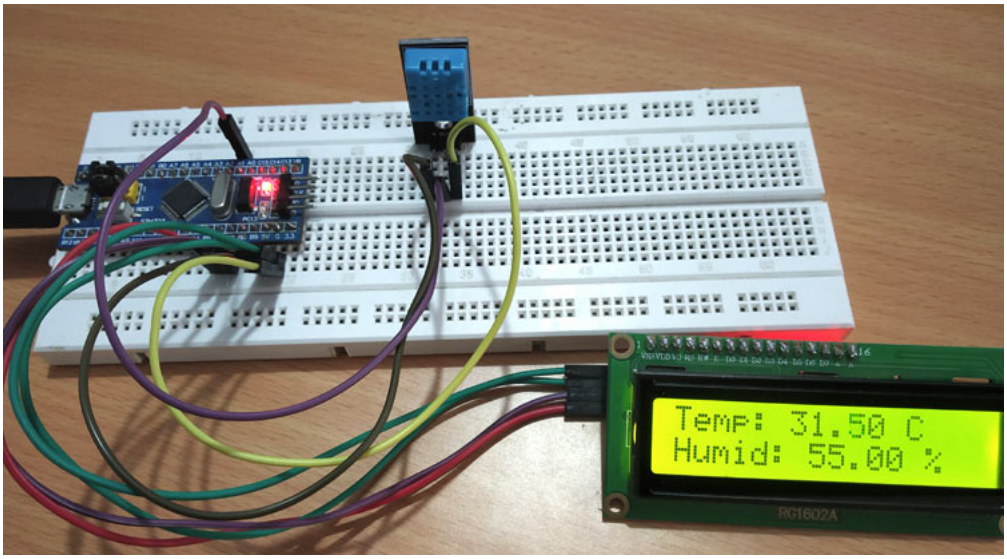
To get only the Temperature value,

```
float t = dht.readTemperature();
```

And finally print those in the 16X2 LCD display,

```
lcd.setCursor(0,0);
lcd.print("Temp: ");
lcd.print(t);
```

```
lcd.print(" C");
lcd.setCursor(0,1);
lcd.print("Humid: ");
lcd.print(h);
lcd.print(" %");
```



Chapter 4

Future Work and Conclusions

These two projects can be further improvised and integrated in the future.

- Use APIs for the app and make a leaderboard of sorts.
- Make a weather app which will take data from APIs and the temperature sensor to give real time temperature and humidity data to the user.
- Using more sensors like, daylight sensor, etc. to make a smart garden which will regulate water given to plants and inform owners about the nutrient level in soil via an app.

Acknowledgment

First of all, I would like to thank Mr. Sahil for helping to organise this internship. I also thank Mr. Aaakash for helping me settle in the office environment and guiding and monitoring me throughout the internship. Overall, it was a wonderful and enriching experience at iljin electronics.

References

[1] Wiki on PCBs - https://en.wikipedia.org/wiki/Printed_circuit_board

[2] Wiki on SMT - https://en.wikipedia.org/wiki/Surface-mount_technology

[3] ST website for stm32f103cb - https://www.st.com/en/microcontrollers-microprocessors/stm32f103cb.html