EXPERIMENTNUMBER-1.2

NAME	UID	SECTION/GROUP
SUJOYRAY	22BCS12022	115/B
PARVEENKUMAR	22BCS15865	115/B
DEEPAKKUMARSINGH	22BCS12849	115/B
ALOKKUMARYADAV	22BCS13070	115/B
DEEPANSHU	22BCS15133	115/B

AIM: Develop a smart phone application for smart home voice-assistant.

TOOLS/ COMPONENTS USED:

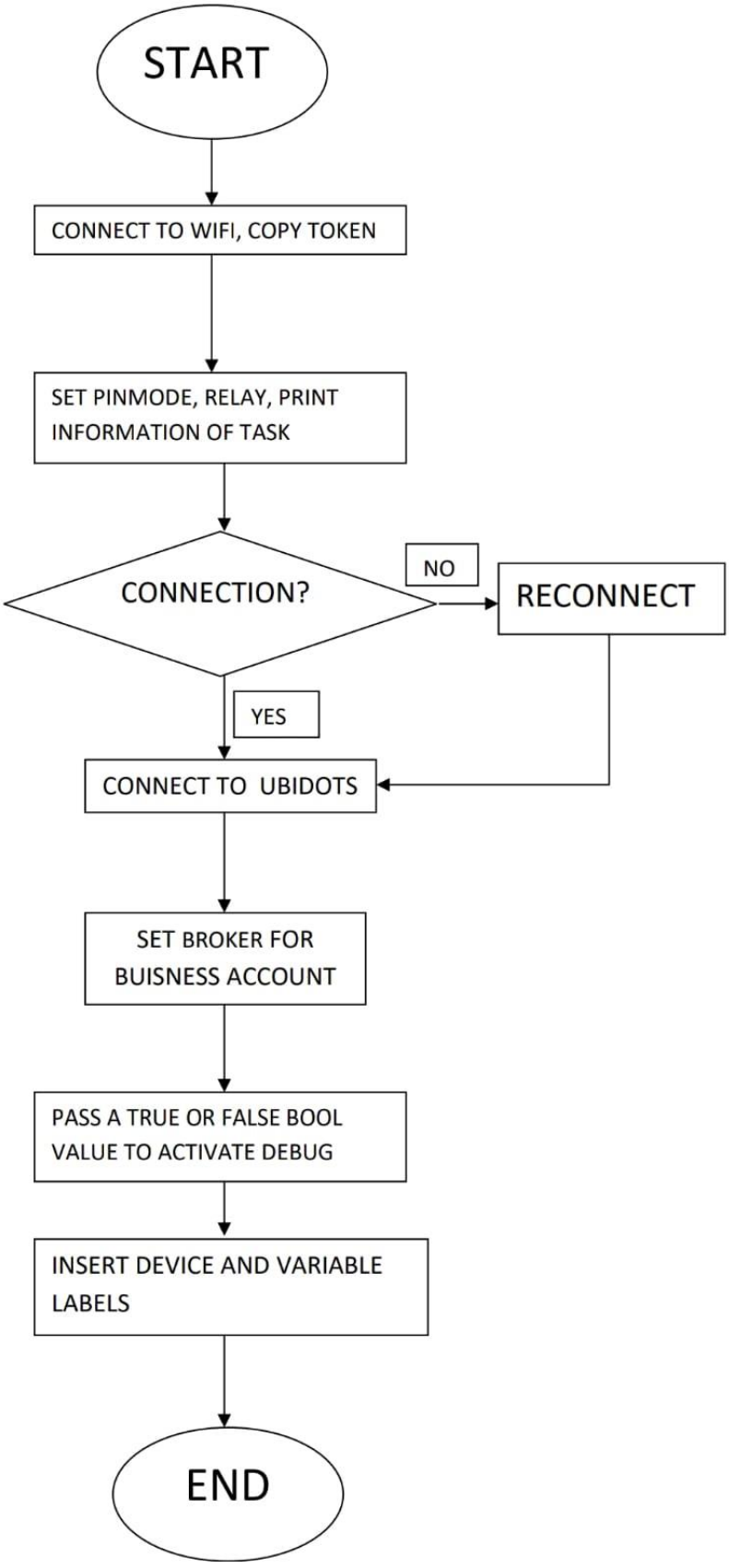
1. MIT App Inventor
2. ESP32
3. Stepper Motor

Basic code and command description

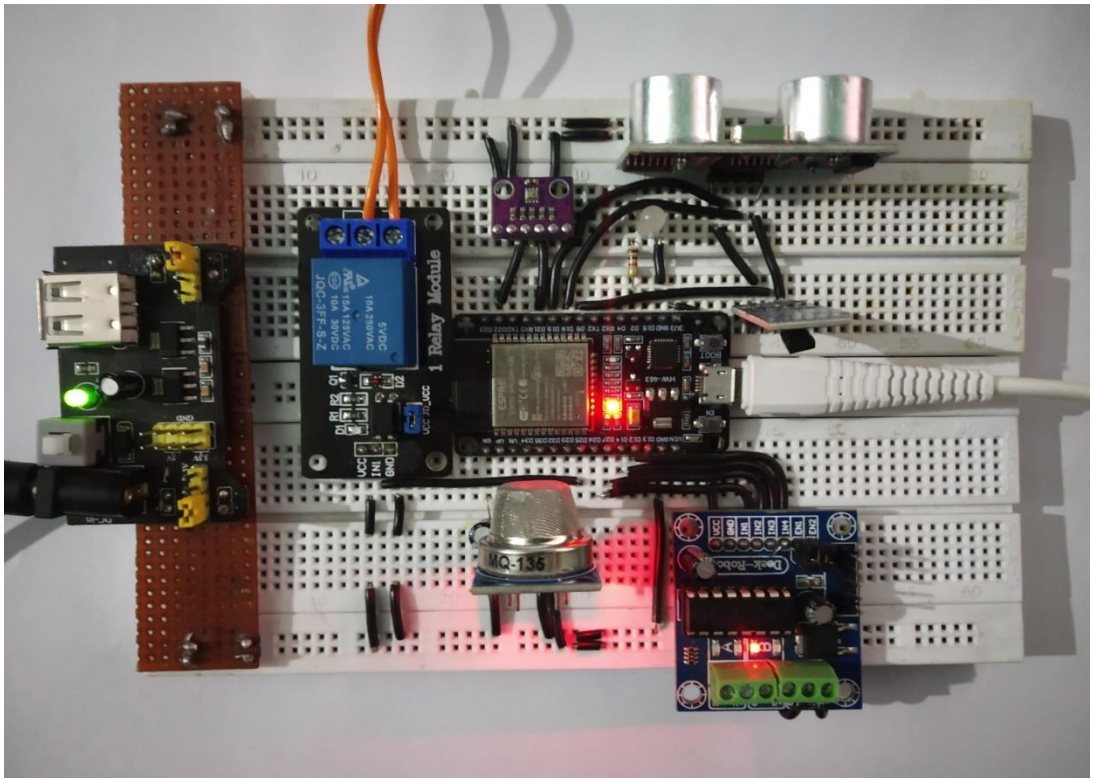
1. MIT App Inventor is an intuitive, visual programming environment that allows everyone even children to build fully functional apps for smartphones and tablets.

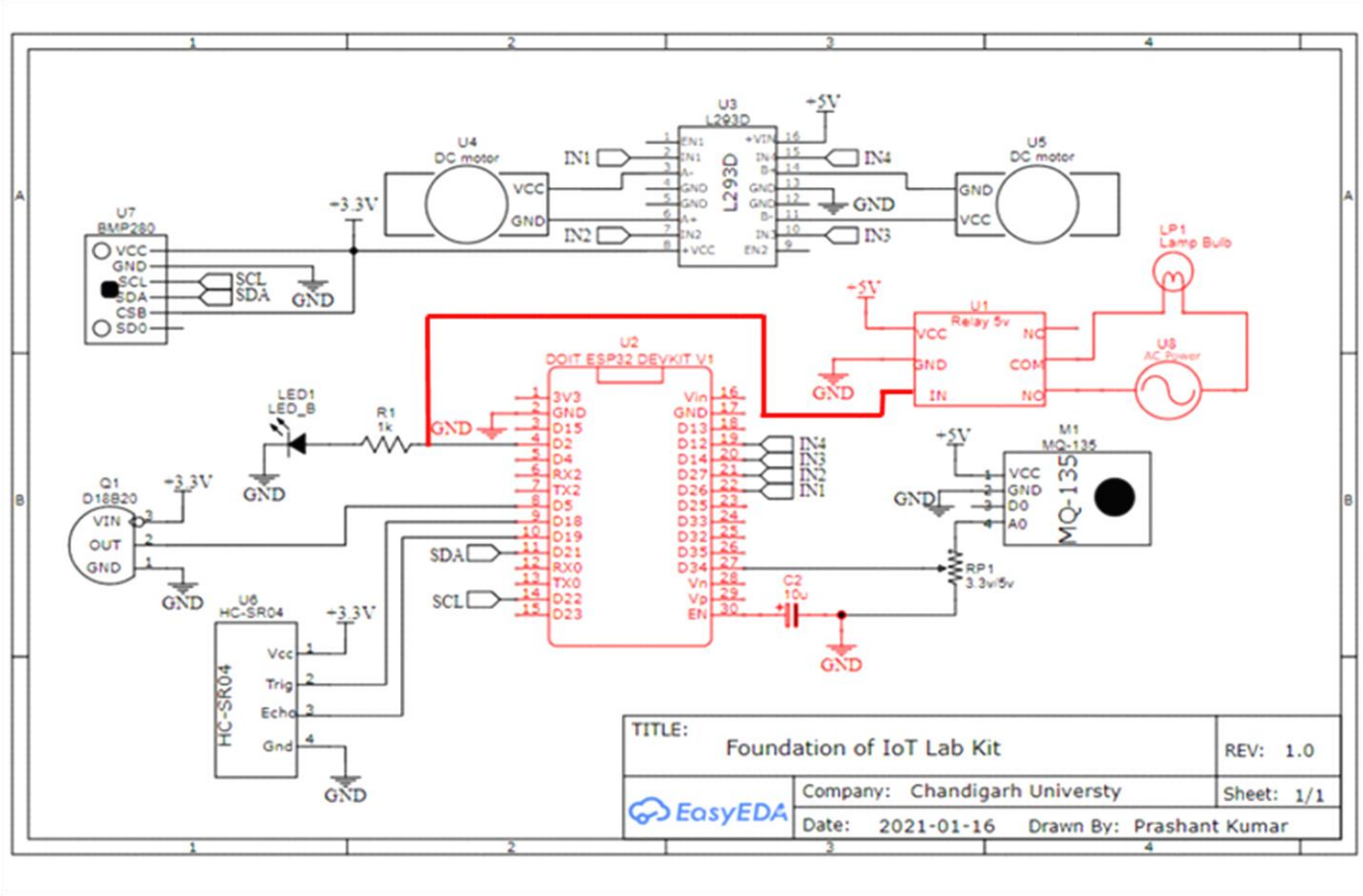
2. ESP32 can perform as a complete standalone system or as a slave device to a host MCU, reducing communication stack overhead on the main application processor. ESP32 can interface with other systems to provide Wi-Fi and Bluetooth functionality through its SPI / SDIO or I2C / UART interfaces.

FLOWCHART-


START
CONNECT TO WIFI, COPY TOKEN
SET PINMODE, RELAY, PRINT INFORMATION OF TASK
CONNECTION?
NO
RECONNECT
YES
CONNECT TO UBIDOTS
SET BROKER FOR BUISNESS ACCOUNT
PASS A TRUE OR FALSE BOOL VALUE TO ACTIVATE DEBUG
INSERT DEVICE AND VARIABLE LABELS
END

FOUNDATION TO LAB KIT





SUBJECTCODE-22ECH-103

PROGRAM CODE-

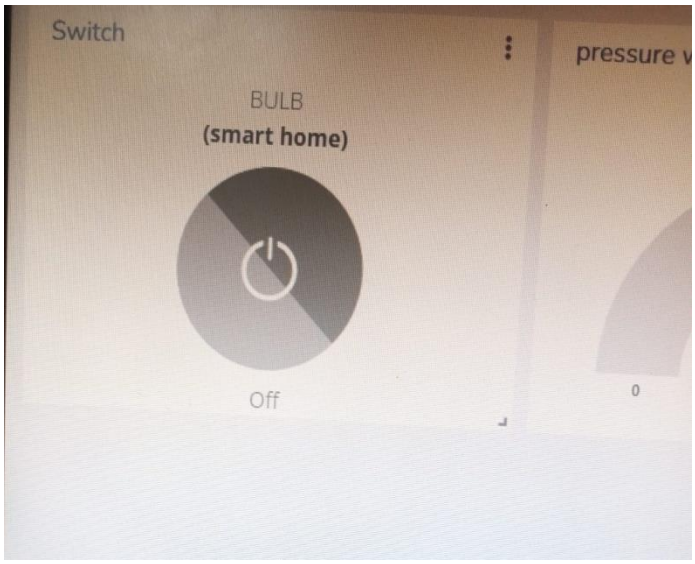
```
#include <UbidotsESPMQTT.h>

#define RELAY 25

#define TOKEN "BBFF-MYmH7HuZvdmMZhBjd8EUYFfOBNcez2"   // Your Ubidots TOKEN

#define WIFISSID "pc34" // Your SSID

#define WIFIPASS "@123456789" // Your Wifi Pass

Ubidots client(TOKEN);

void callback(char* topic, byte* payload, unsigned int length) {
 Serial.print("Message arrived [");
 Serial.print(topic);
 Serial.print("] ");
 for (int i = 0; i < length; i++) {
  Serial.print((char)payload[i]);
 }
 Serial.println();
 Serial.print("Command: ");
 bool command = *payload - 48;
```

```
 Serial.println(command);
 digitalWrite(RELAY, !command);
}
void setup() {
 Serial.begin(9600);
 Serial.println("Init... T4_Smart_Home");
 pinMode(RELAY, OUTPUT);
 Serial.print("Connecting to SSID: ");
 Serial.print(WIFISSID);
 Serial.print(", Password: ");
 Serial.println(WIFIPASS);
 client.wifiConnection(WIFISSID, WIFIPASS);
 Serial.println("Done");
 Serial.println(" Initializing Ubidots Connection...");
 client.ubidotsSetBroker("industrial.api.ubidots.com"); // Sets the broker
properly for the business account
 client.setDebug(true);          // Pass a true or false bool value to activate
debug messages
 client.begin(callback);
```

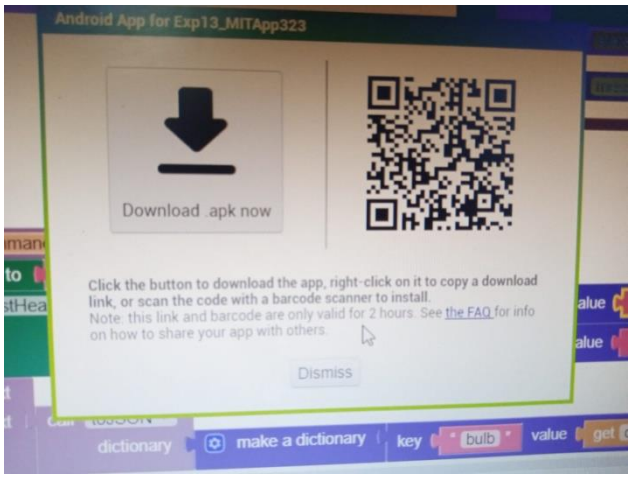
```
 client.ubidotsSubscribe("smart-home-voice-assistant","bulb"); //Insert the Device and Variable's Labels

 Serial.println("Done");

 Serial.println("DONE");

}

void loop() {

   // Establising connection with Ubidots

 if (!client.connected()) {

  client.reconnect();

  client.ubidotsSubscribe("smart-home-voice-assistant","bulb"); //Insert the Device and Variable's Labels }

 client.loop();

 delay(1000);

}
```

Observations, stimulation, screenshots and discussions-

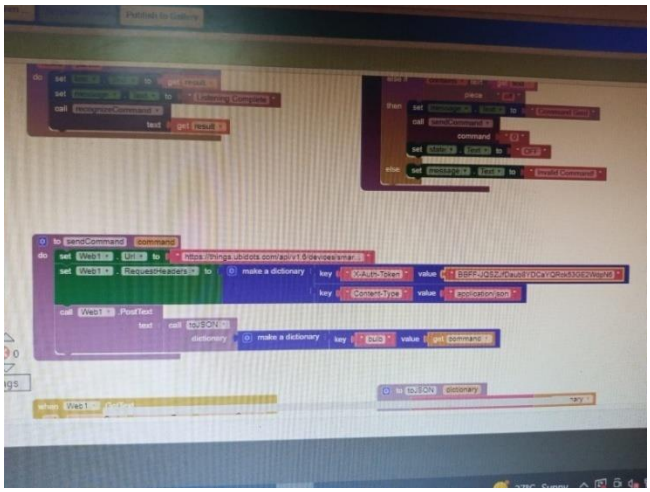
App interface



Ubidots button



App download link



Block structure of application

Result and summary-

1. **MIT App Inventor:** MIT App Inventor is an intuitive, visual programming environment that allows everyone even children to build fully functional apps for smartphones and tablets.

2. **Ubidots**: Ubidots is a cloud based Internet of Things(IoT) data analytics and visualization platform.

EVALUATION COLUMN
(To be filled by Concerned faculty only)

S.No.	Parameters	Maximum Marks	Marks Obtained
1	WorksheetCompletion	10	
2.	Viva	8	
3.	Conduct	12	
	TotalMarks	30	