**Internet of Things**

**Lab Report 11**

**Hafiz Ahmad**

**19l-1316**

**Section-7A2**

**MIT App Inventor and NODEMCU**

**INTRODUCTION:**

You can create fully functional apps for Android phones, iPhones, and Android/iOS tablets using the online MIT app inventor platform.Utilizing its block-based tools, it encourages its users to create innovative and well-designed apps.Since the 1950s, artificial intelligence has played a significant role in computing.By automating previously manual tasks, artificial intelligence (AI) continuously boosts business productivity.By utilizing its user-friendly block-based facilities and designing highly impactful apps in significantly less time than traditional programming environments, MIT provides its students with new opportunities to explore and create original AI applications.

**OBJECTIVES:**

To learn about IoT (Internet of Things)MIT APP INVENTOR and NODEMCU.

**Lab code:**

#include <ThingSpeak.h> // add librery

#include <ESP8266WiFi.h>

WiFiClient client;

unsigned long counterChannelNumber = 1027878; // Channel ID

const char \* myCounterReadAPIKey = "CXADD6SZ5W4RKRHD"; // Read API Key

const int FieldNumber1 = 1; // The field you wish to read

uint8\_t LEDpin = D6;

unsigned int presentValue = 0;

unsigned int changeValue = 0;

void setup()

{

Serial.begin(115200);

Serial.println();

WiFi.begin("POCO PHONE", "9004652173"); // write wifi name & password

Serial.print("Connecting");

while (WiFi.status() != WL\_CONNECTED)

{

delay(500);

Serial.print(".");

}

Serial.println();

Serial.print("Connected, IP address: ");

Serial.println(WiFi.localIP());

ThingSpeak.begin(client);

}

void loop()

{

int A = ThingSpeak.readLongField(counterChannelNumber, FieldNumber1, myCounterReadAPIKey);

presentValue = A;

if (presentValue != changeValue)

{

Serial.println(A);

analogWrite(LEDpin , A);

changeValue = presentValue;

}

}

#include <ThingSpeak.h> // add librery

#include <ESP8266WiFi.h>

WiFiClient client;

unsigned long counterChannelNumber = 980157; // Channel ID

const char \* myCounterReadAPIKey = "CL8DHB4NZ43291ZI"; // Read API Key

const int FieldNumber1 = 1; // The field you wish to read

const int FieldNumber2 = 2; // The field you wish to read

void setup()

{

pinMode(13,OUTPUT);

Serial.begin(115200);

Serial.println();

WiFi.begin("POCO PHONE", "9004652173"); // write wifi name & password

Serial.print("Connecting");

while (WiFi.status() != WL\_CONNECTED)

{

delay(500);

Serial.print(".");

}

Serial.println();

Serial.print("Connected, IP address: ");

Serial.println(WiFi.localIP());

ThingSpeak.begin(client);

}

void loop()

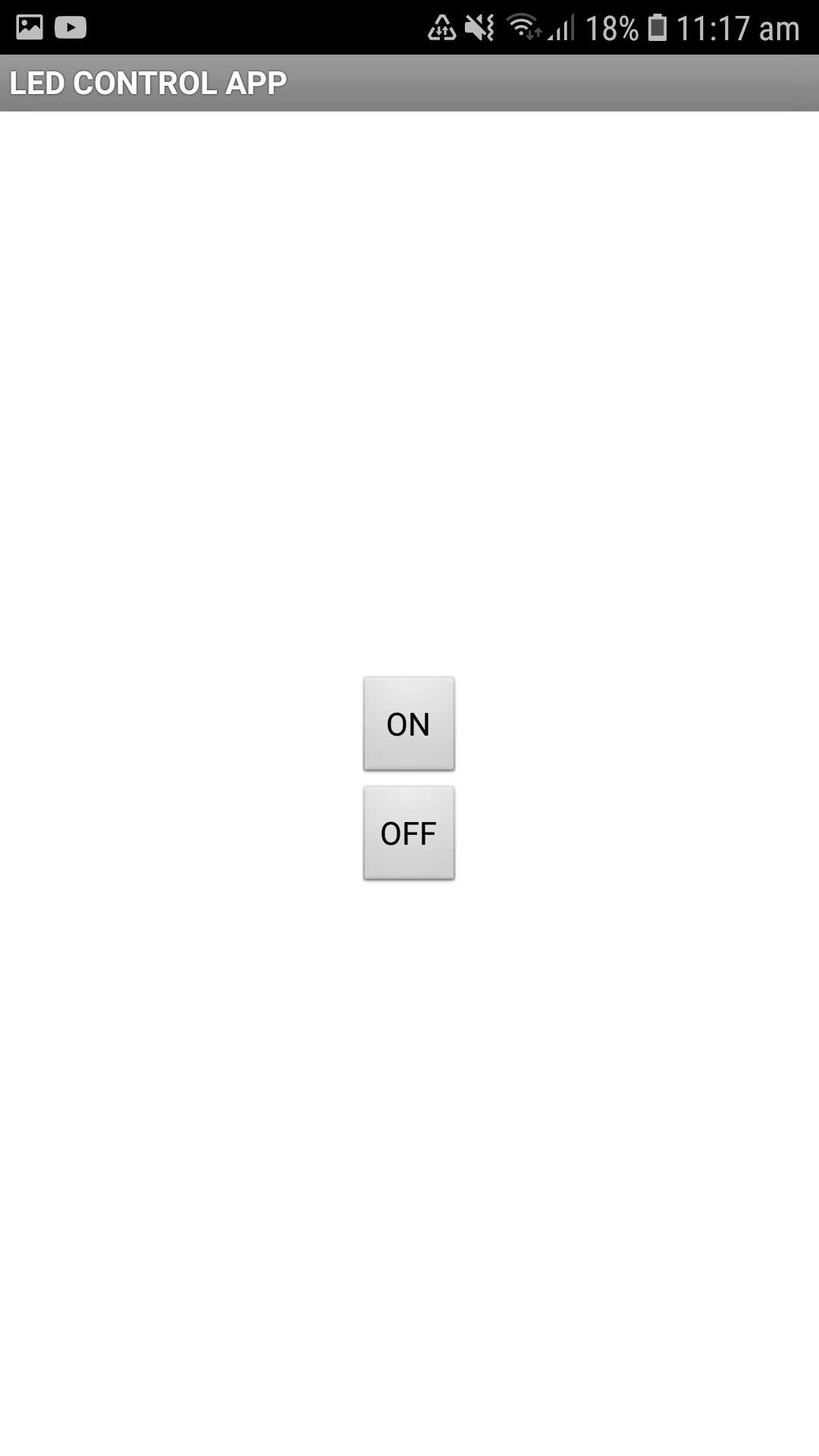
{

int A = ThingSpeak.readLongField(counterChannelNumber, FieldNumber1, myCounterReadAPIKey);

Serial.println(A);

digitalWrite(13,A);

}



**Application:**

MIT App Inventor is a visual programming environment that is easy to use and lets anyone, even kids, create fully functional apps for smartphones and tablets.In less than 30 minutes, novice users of MIT App Inventor can launch a basic first app.Using a programming language based on blocks, you can create your own mobile apps with the free cloud-based service App Inventor.There are two main categories of parts:visible as well as invisible.components that are visible, such as a Button, TextBox, Label, etc.are incorporated into the User Interface.

**Issues:**

we never find any issue regarding this lab.

**Conclusion:**

In this lab we perform IoT (Internet of Things) based application on android using MIT app inventor. The block-based programming environment known as MIT App Inventor makes it easy for novice programmers to create useful apps for smartphones and tablets.IoT extensions from the MIT App In- ventor team have made it possible for people to design and create apps that can interact with physical devices.