Dr. Vishwanath Karad MIT WORLD PEACE UNIVERSITY | PUNE
TECHNOLOGY, RESEARCH, SOCIAL INNOVATION & PARTNERSHIPS

Second Year B. Tech (EL&CE)

Semester: IV **Subject:** Basic IoT Lab

Name: Shreerang Mhatre **Class: Sy**

Roll No: 29 **Batch: A2**

Experiment No: 08

Name of the Experiment: To introduce new hardware platforms TinyML

Performed on: 19/04/23

Submitted on: 28/04/23

Marks	Teacher's Signature with date

Aim: To introduce hardware platforms for IoT based design TinyML

Pre-requisite: Knowledge of Electronics and circuits, Knowledge of Assembly and C Programming language.

Objective:

1. To introduce new hardware platforms for IoT based design.
2. To understand different features of new hardware platforms.

Components and equipment required:

Arduino Uno Board, TinyML

Theory:

IoT hardware platforms are considered as the most significant component of the IoT environment. One IoT device can be connected to other IoT device to enable the information exchange using suitable Internet protocols. IoT platforms works as a bridge between the device sensors and data networks. It gives data connectivity to the sensor system and helps to understand using back-end applications and helps to analyse the huge data generated by sensors.

What is TinyML

TinyML is a field of study in Machine Learning and Embedded Systems that explores the types of models you can run on small, low-powered devices like microcontrollers. It enables low-latency, low power and low bandwidth model inference at edge devices. While a standard consumer CPUs consume between 65 watts and 85 watts and standard consumer GPU consumes anywhere between 200 watts to 500 watts, a typical microcontroller consumes power in the order of milliwatts or microwatts. That is around a thousand times less power consumption. This low power consumption enables the TinyML devices to run unplugged on batteries for weeks, months, and in some cases, even years, while running ML applications on edge.

Advantages of TinyML

1. **Low Latency:** Since the model runs on the edge, the data doesn't have to be sent to a server to run inference. This reduces the latency of the output.
2. **Low Power Consumption:** As we discussed before, microcontrollers consume very little power. This enables them to run without being charged for a really long time.
3. **Low Bandwidth:** As the data doesn't have to be sent to the server constantly, less internet bandwidth is used.
4. **Privacy:** Since the model is running on the edge, your data is not stored in any servers.

.

Applications of TinyML

By summarizing and analyzing data at the edge on low power devices, TinyML offers many unique solutions. Even though TinyML is an emerging field, it has been used in production for years. The “OK Google”, “Alexa”, “Hey Siri” wake words are an example of TinyML. Here, the devices are always on and are analyzing your voice to detect the wake word. I’ll add some more applications of TinyML here.

1. **Industrial Predictive Maintenance:** Machines are prone to fault. Using TinyML on low powered devices, it is possible to monitor the machine and predict faults ahead of time constantly. This predictive maintenance can lead to significant cost savings. Ping Services, an Australian startup, has introduced an IoT device that autonomously monitors wind turbines by magnetically attaching to the outside of the turbine and analyzing detailed data at the edge. This device can alert the authorities regarding potential issues even before it occurs.

2. **Healthcare:** The Solar Scare Mosquito project uses TinyML to curb the spread of mosquito-borne diseases like Dengue, Malaria, Zika Virus, Chikungunya, etc. It works by detecting the mosquito breeding conditions and agitates the water to prevent mosquito breeding. It runs on solar power and can thus run indefinitely.

3. **Agriculture:** The Nuru app helps farmers detect diseases in plants just by taking a picture of it by running Machine Learning models on the device using TensorFlow Lite. Since it works on the device, there is no need for an internet connection. This is a crucial requirement for remote farmers since they might not have proper internet connection in their place.

4. **Ocean Life Conservation:** Smart ML-powered devices are used to monitor whales in real-time in waterways around Seattle and Vancouver to avoid whale strikes in busy shipping lanes.

Description :

1. **Hardware:** The **Arduino Nano 33 BLE Sense** is the suggested hardware for deploying Machine Learning models on edge. It contains a 32-bit ARM Cortex-M4F microcontroller running at 64MHz with 1MB of program memory and 256KB RAM. This microcontroller provides enough horsepower to run TinyML models. The Arduino Nano 33 BLE Sense also contains colour, brightness, proximity, gesture, motion, vibration, orientation, temperature, humidity, and pressure sensors. It also contains a digital microphone and a Bluetooth low energy(BLE) module. This sensor suite will be more than enough for most applications.

2. **Machine Learning Framework:** There are only a handful of frameworks that cater to TinyML needs. Of that, **TensorFlow Lite** is the most popular and has the most community support. Using TensorFlow Lite Micro, we can deploy models on microcontrollers.

3. **Learning Resources:** Since TinyML is an emerging field, there aren't many learning materials as of today. But there are a few excellent materials like Pete Warden and Daniel Situnayake's book, "TinyML: Machine Learning with TensorFlow Lite on Arduino and Ultra-Low-Power", Harvard University's Course on TinyML by Vijay Janapa Reddi, and Digikey's blogs and videos on TinyML.

Diagram of TinyML:


SYNTIANT
TinyML
T126A
P1_210805

Key Features

- Neural Decision Processor: NDP101
- Host processor: SAMD21 Cortex-M0+ 32bit low power ARM MCU, including:
- 256KB flash memory
- 32KB host processor SRAM
- Board power supply: 5V micro-USB or 3.7V LiPo battery
- 5 Digital I/Os compatible with Arduino MKR series boards
- 1 UART interface (included in the digital I/O Pins
- 1 I2C interface (included in the digital I/O Pins)
- 2MB on-board serial flash
- 48MHz system clock
- One user defined RGB LED
- uSD card slot (uSD card not included)
- BMI160 6 axis motion sensor
- SPH0641LM4H microphone

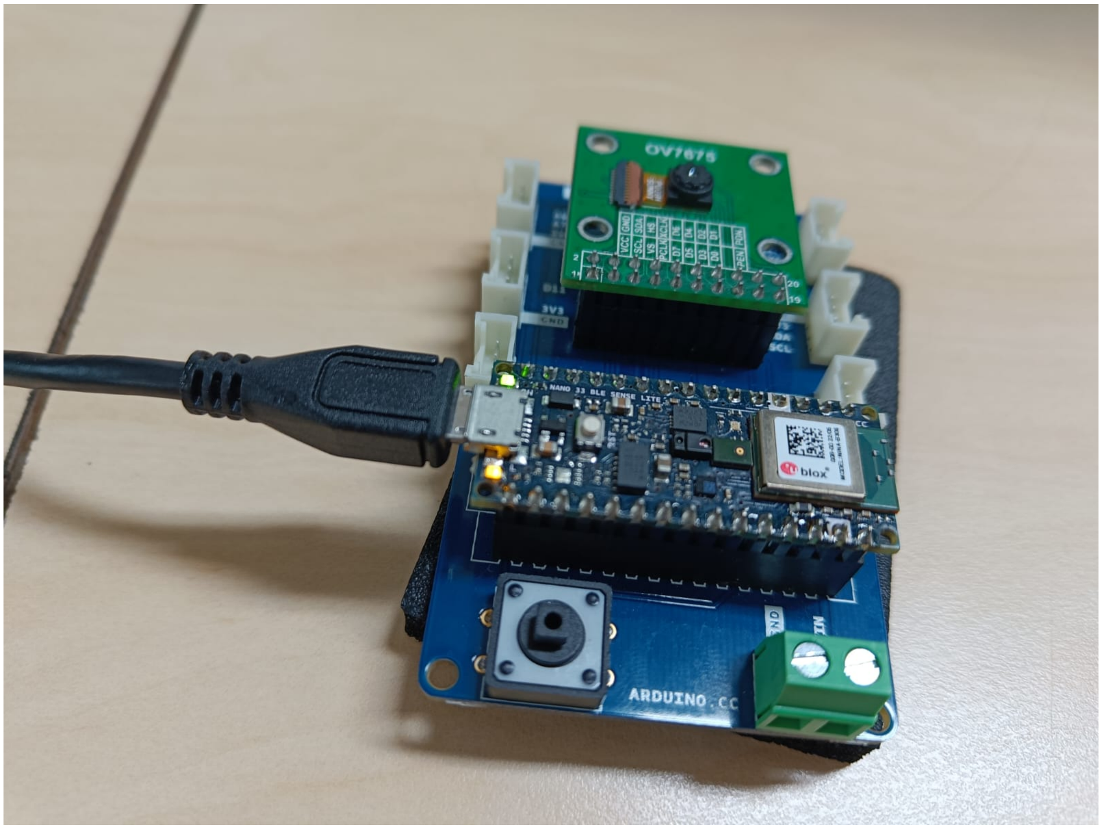
Program CODE :

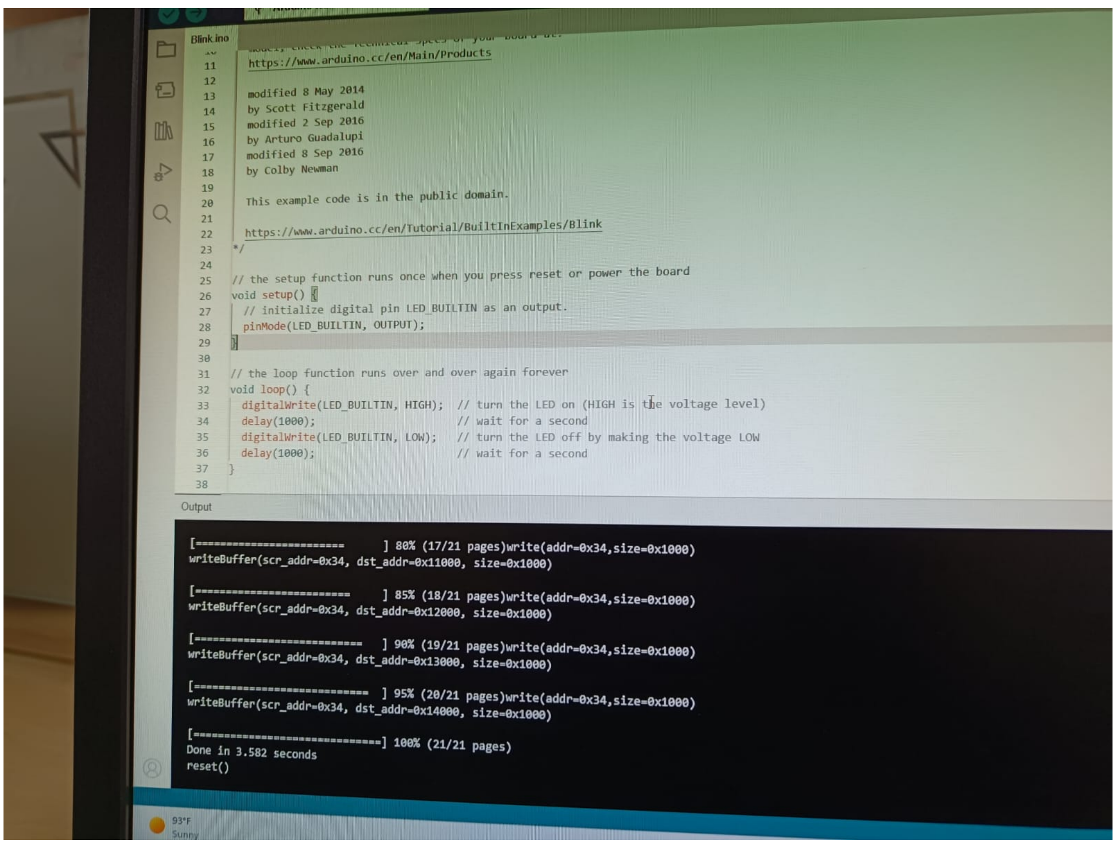
Output :

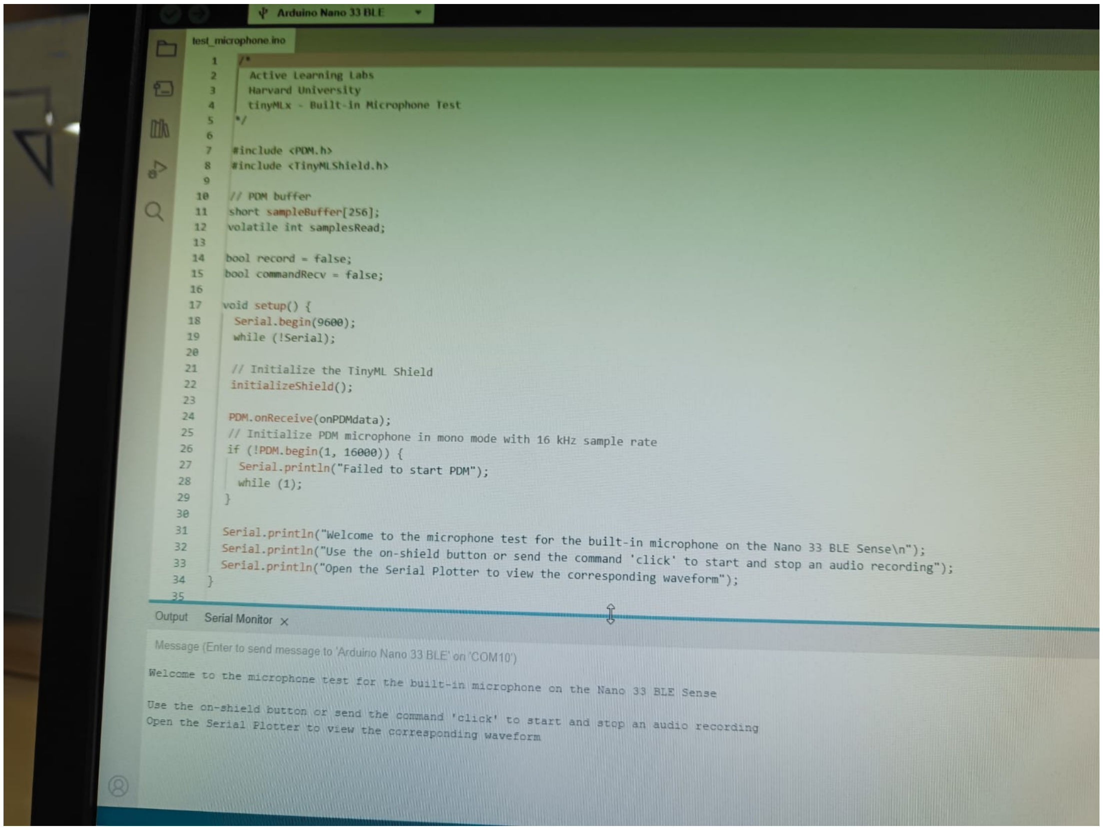
Conclusion :

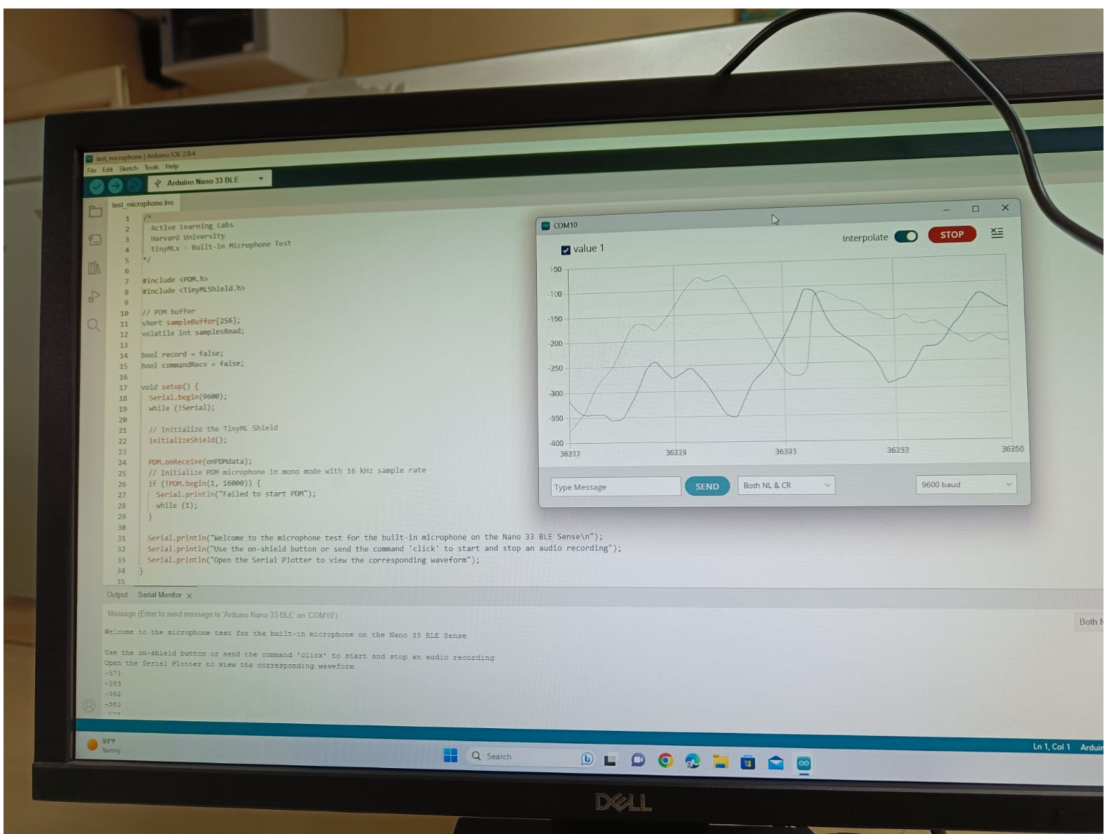
Post lab questions :

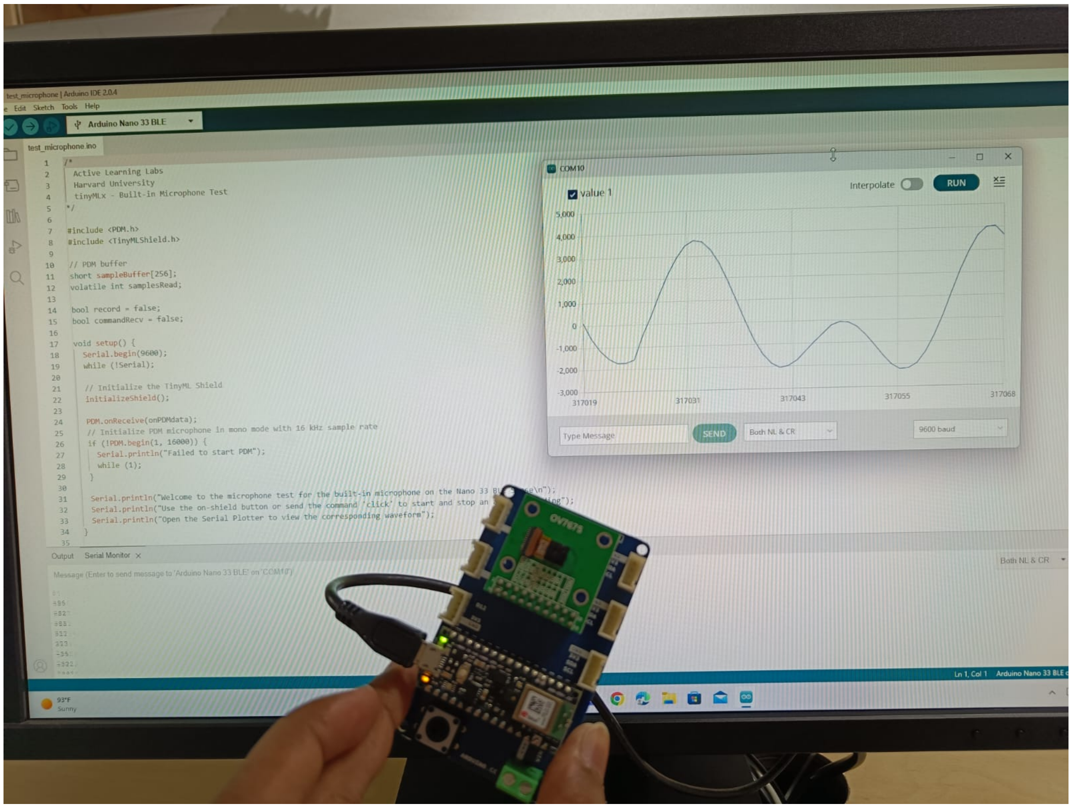
1. Describe different features present in the TinyML kit .

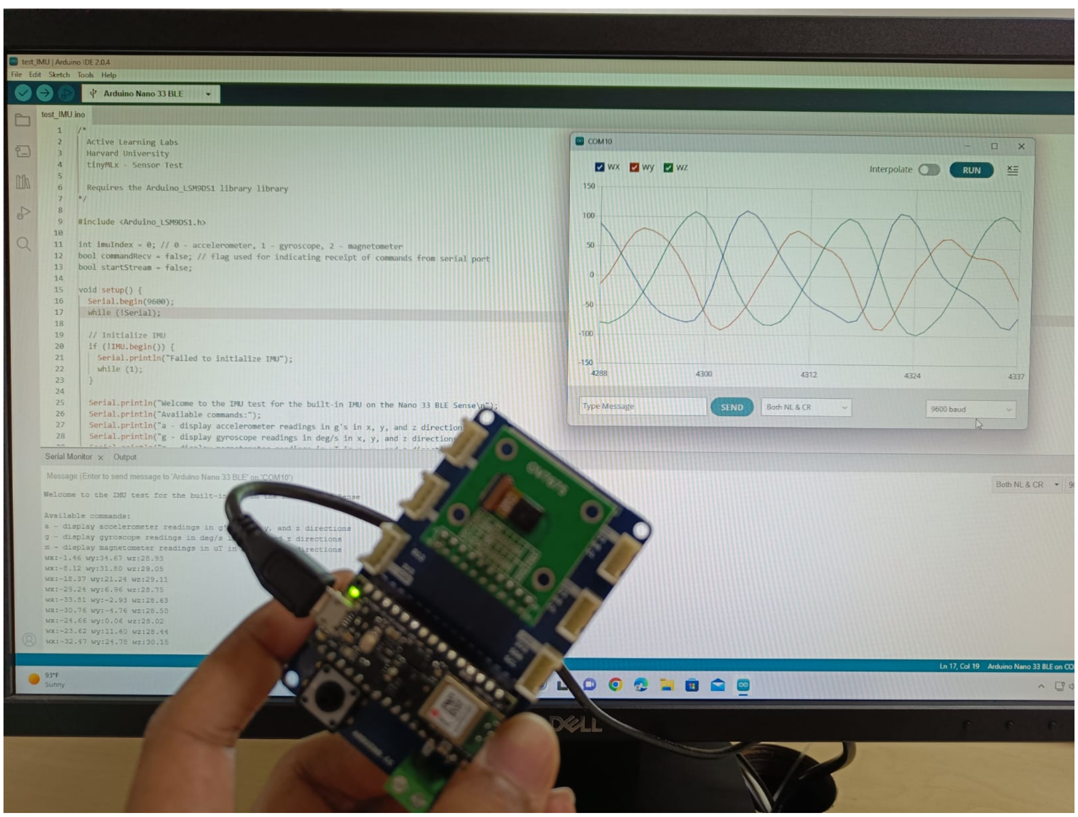

OV7675
ARDUINO.CC
NANO 33 BLE SENSE LITE
blox


Blink.ino
https://www.arduino.cc/en/Main/Products
modified 8 May 2014
by Scott Fitzgerald
modified 2 Sep 2016
by Arturo Guadalupi
modified 8 Sep 2016
by Colby Newman
This example code is in the public domain.
https://www.arduino.cc/en/Tutorial/BuiltInExamples/Blink
*/
// the setup function runs once when you press reset or power the board
void setup() {
// initialize digital pin LED_BUILTIN as an output.
pinMode(LED_BUILTIN, OUTPUT);
}
// the loop function runs over and over again forever
void loop() {
digitalWrite(LED_BUILTIN, HIGH); // turn the LED on (HIGH is the voltage level)
delay(1000); // wait for a second
digitalWrite(LED_BUILTIN, LOW); // turn the LED off by making the voltage LOW
delay(1000); // wait for a second
}
Output
[======================] 80% (17/21 pages)write(addr=0x34,size=0x1000)
writeBuffer(scr_addr=0x34, dst_addr=0x11000, size=0x1000)
[=======================] 85% (18/21 pages)write(addr=0x34,size=0x1000)
writeBuffer(scr_addr=0x34, dst_addr=0x12000, size=0x1000)
[========================] 90% (19/21 pages)write(addr=0x34,size=0x1000)
writeBuffer(scr_addr=0x34, dst_addr=0x13000, size=0x1000)
[=========================] 95% (20/21 pages)write(addr=0x34,size=0x1000)
writeBuffer(scr_addr=0x34, dst_addr=0x14000, size=0x1000)
[==========================] 100% (21/21 pages)
Done in 3.582 seconds
reset()
93°F
Sunny


Arduino Nano 33 BLE
test_microphone.ino
/*
Active Learning Labs
Harvard University
tinyMLx - Built-in Microphone Test
*/
#include <PDM.h>
#include <TinyMLShield.h>
// PDM buffer
short sampleBuffer[256];
volatile int samplesRead;
bool record = false;
bool commandRecv = false;
void setup() {
Serial.begin(9600);
while (!Serial);
// Initialize the TinyML Shield
initializeShield();
PDM.onReceive(onPDMdata);
// Initialize PDM microphone in mono mode with 16 kHz sample rate
if (!PDM.begin(1, 16000)) {
Serial.println("Failed to start PDM");
while (1);
}
Serial.println("Welcome to the microphone test for the built-in microphone on the Nano 33 BLE Sense\n");
Serial.println("Use the on-shield button or send the command 'click' to start and stop an audio recording");
Serial.println("Open the Serial Plotter to view the corresponding waveform");
}
Output
Serial Monitor
Message (Enter to send message to 'Arduino Nano 33 BLE' on 'COM10')
Welcome to the microphone test for the built-in microphone on the Nano 33 BLE Sense
Use the on-shield button or send the command 'click' to start and stop an audio recording
Open the Serial Plotter to view the corresponding waveform


COM10
value 1
Interpolate
STOP
Type Message
SEND
Both NL & CR
9600 baud
Serial Monitor
Welcome to the microphone test for the built-in microphone on the Nano 33 BLE Sense
Use the on-shield button or send the command 'click' to start and stop an audio recording
Open the Serial Plotter to view the corresponding waveform
DELL


test_microphone | Arduino IDE 2.0.4
Arduino Nano 33 BLE
test_microphone.ino
COM10
value 1
Interpolate
RUN
Type Message
SEND
Both NL & CR
9600 baud


test_IMU | Arduino IDE 2.0.4
Arduino Nano 33 BLE
test_IMU.ino
COM10
wx
wy
wz
Interpolate
RUN
Type Message
SEND
Both NL & CR
9600 baud

Name: Shreerang Mhatre
Rollno: 29
Batch: A2

19/04/2023

Exp No. 8

* Post lab Questions:

Q1) Describe different features present in the Tiny ML kit.

-> Different features present in the Tiny ML kit are -

① Micro controller -
A microcontroller is a small computer that can be embedded in a device to control its functions. A Tiny ML kit usually includes a microcontroller that is optimized for running machine learning algorithms.

② Sensors -
Sensors are devices that can detect and measure physical phenomena such as temperature motion, and light. Tiny ML kits often come with sensors that can be used to collect data for training and testing machine learning models

->

www.mitwpu.edu.in

③ Machine learning algorithms -
Tiny ML kits typically include pre trained machine learning algorithms that can be run on the microcontroller. These algorithms are usually optimized for running on low-power devices with limited resources

④ Development tools -
Tiny ML kits often come with software development tools that make it easier to develop, test, and deploy machine learning models. These tools may include libraries.

⑤ connectivity -
Tiny ML kits may include features that allow them to connect to other devices or the internet. This can be useful for sending data to cloud-based machine learning models for training or for receiving updates to the device's machine learning algorithms

SSP
28/4/23

www.mitwpu.edu.in