

Harvard



Yá'át'ééh 🖐️

EASI-22

Edge AI Summer
Institute 2022

with Navajo Tech



Our website!

tinyMLedu.org/EASI-22

home base for **all information!**

Workshop **Agenda**

Day 1

Introduction to AI and (Tiny)ML

Cloud ML

Day 2

Keyword Spotting for the Navajo Language

Mobile ML

Day 3

Bringing AI/ML from the Cloud to the Edge

Embedded ML

Workshop **Agenda**

Day 1

Introduction to AI and (Tiny)ML

Cloud ML

Day 2

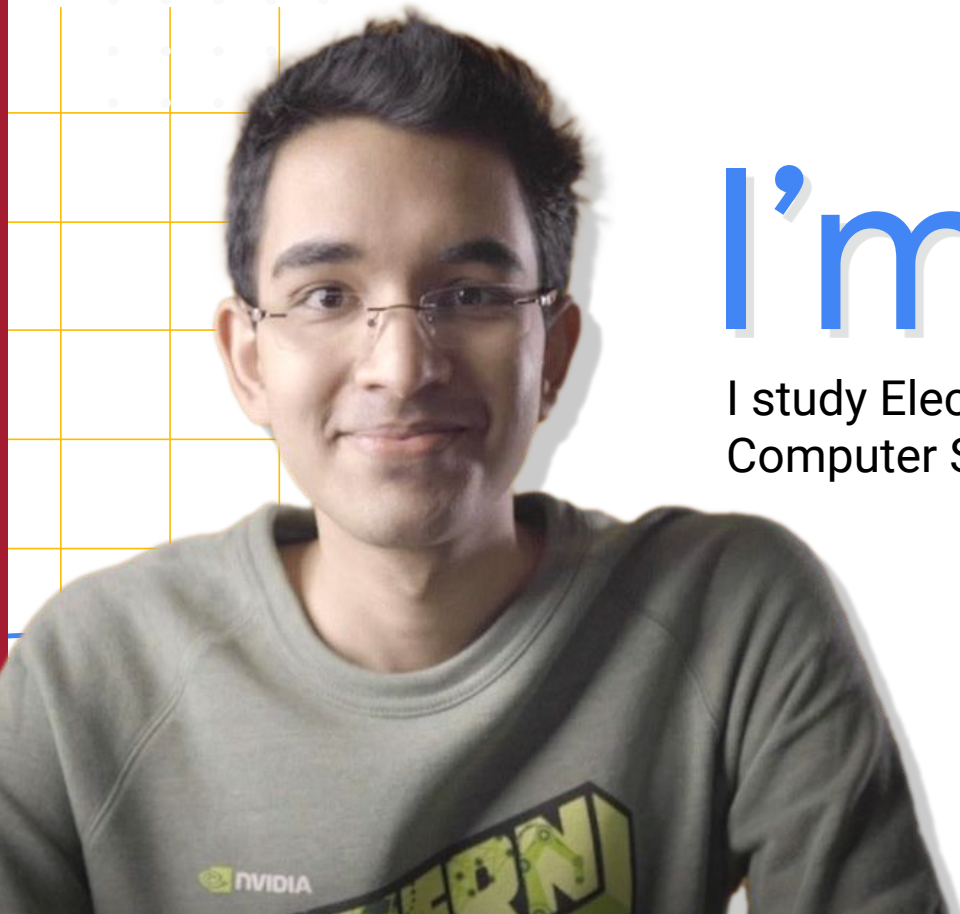
Keyword Spotting for the Navajo Language

Mobile ML

Day 3

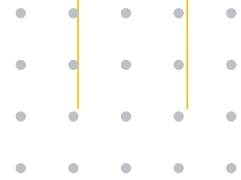
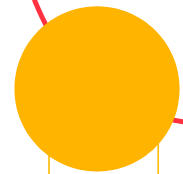
Bringing AI/ML from the Cloud to the Edge

Embedded ML



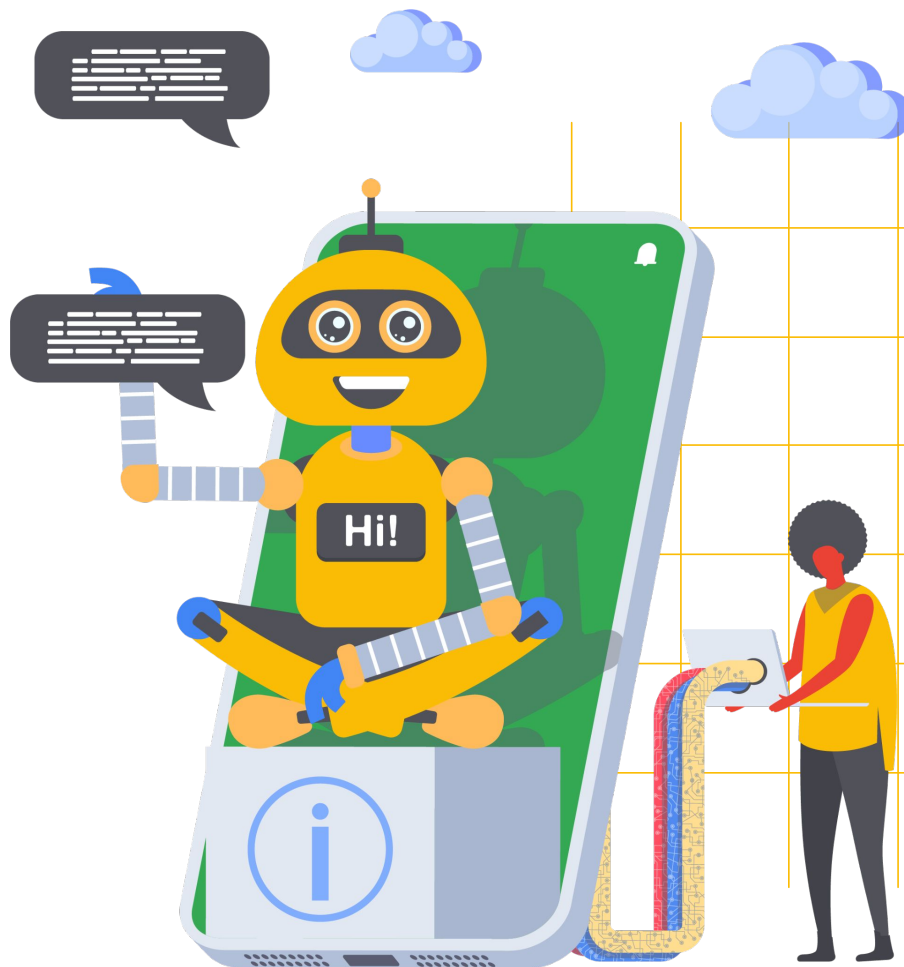
I'm Dhilan!

I study Electrical Engineering and
Computer Science at **Harvard**.



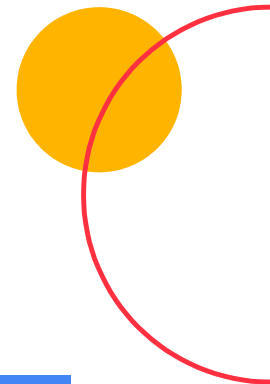
Harvard

BARNARD



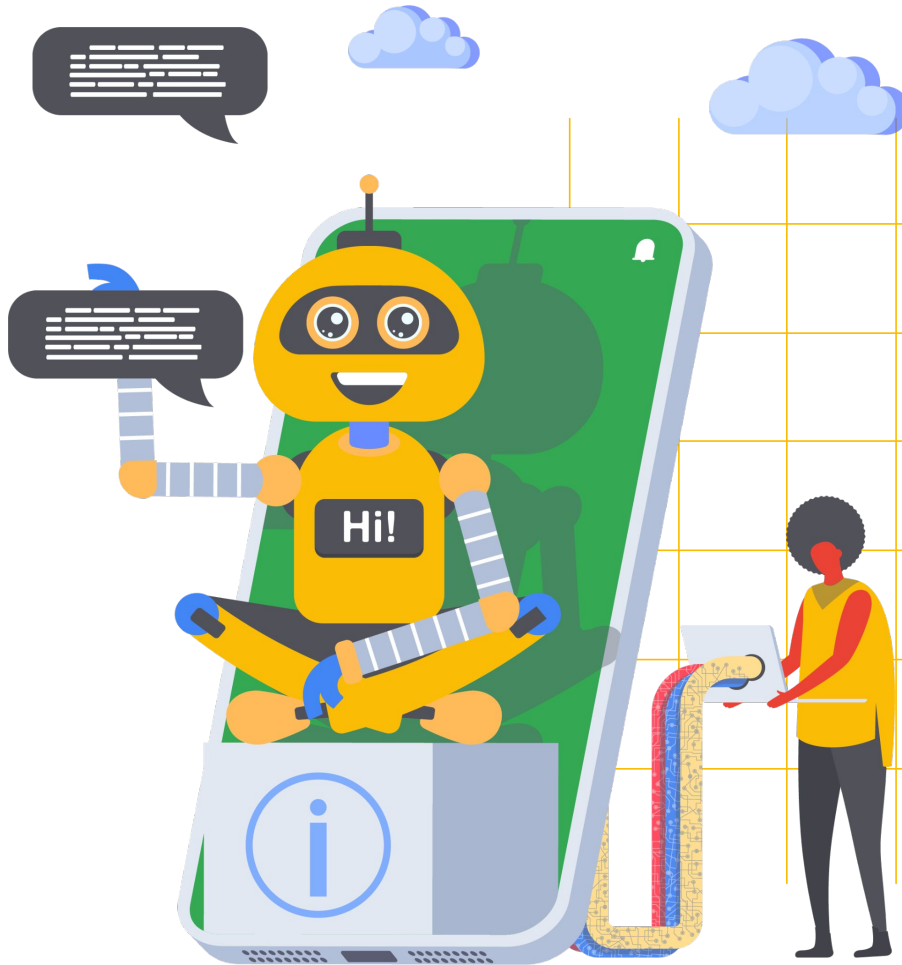
How ML works?

with Brian



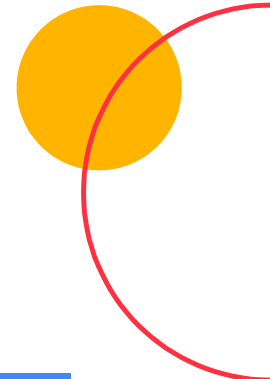
Harvard

BARNARD

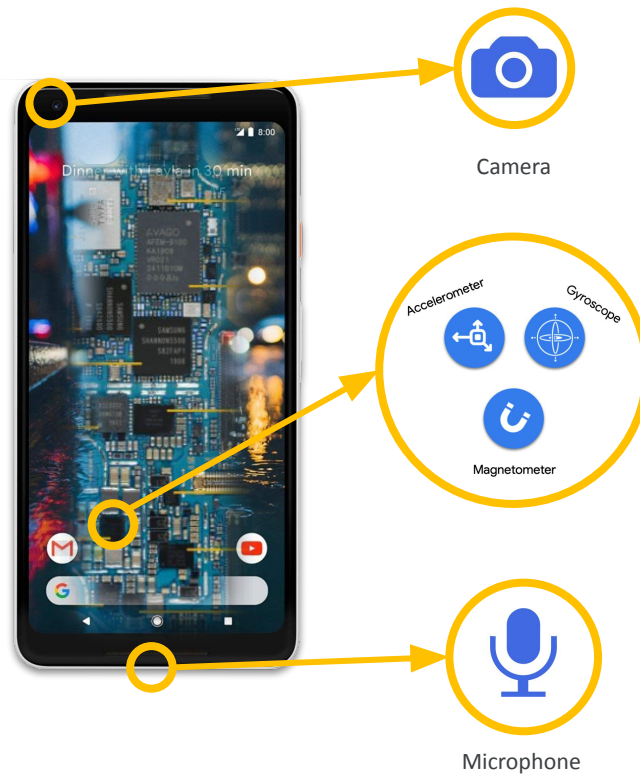
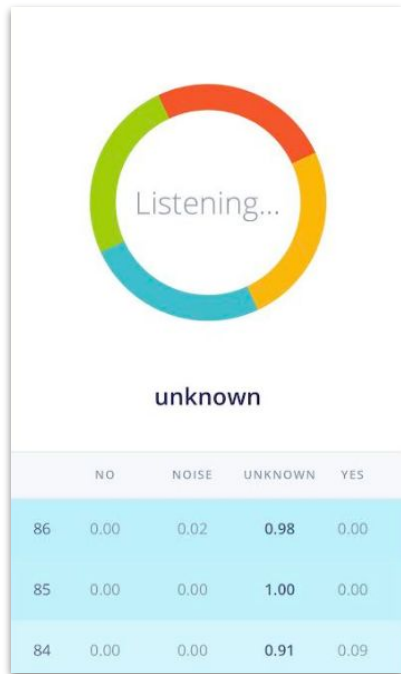


Where ML works?

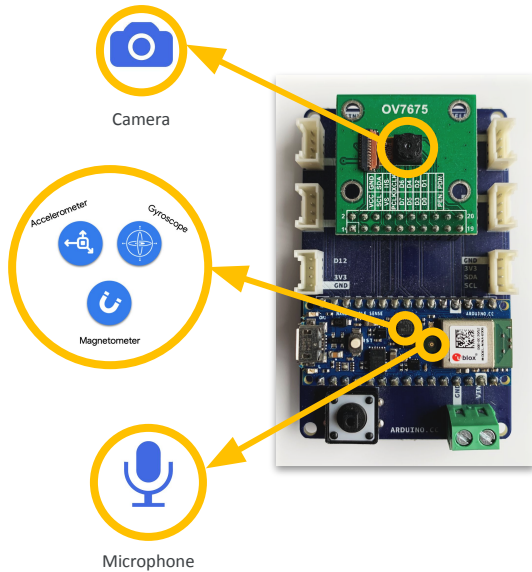
with Dhilan



Last time: Phone



Today: TinyML Kit



```
File Edit Sketch Tools Help
/dev/ttyACM0
Recording done
Predictions (DSP: 205 ms., Classification: 5 ms., Anomaly: 0 ms.):
no: 0.00000
Predictions (DSP: 205 ms., Classification: 5 ms., Anomaly: 0 ms.)
no: 0.00000
noise: 0.10156
unknown: 0.89062
yes: 0.00781
Starting inferring in 2 seconds...
Recording...
Predictions (DSP: 205 ms., Classification: 5 ms., Anomaly: 0 ms.):
no: 0.00000
noise: 0.10156
unknown: 0.89062
yes: 0.00781
Starting inferring in 2 seconds...
Recording...
Recording done
Autoscroll Show timestamp Both NL & CR 9600 baud Clear output
Done in 6.694 seconds
```

Today's Agenda

- Review + Why Tiny?
- Hardware Basics
- Installing and Starting the Arduino IDE
- Testing Your TinyML Kit
- Deploying KWS model onto Arduino
- Summary & Next Steps

Today's Agenda

Review + Why Tiny?

- Hardware Basics
- Installing and Starting the Arduino IDE
- Testing Your TinyML Kit
- Deploying KWS model onto Arduino
- Summary & Next Steps



Machine Learning **Workflow**



Machine Learning Workflow

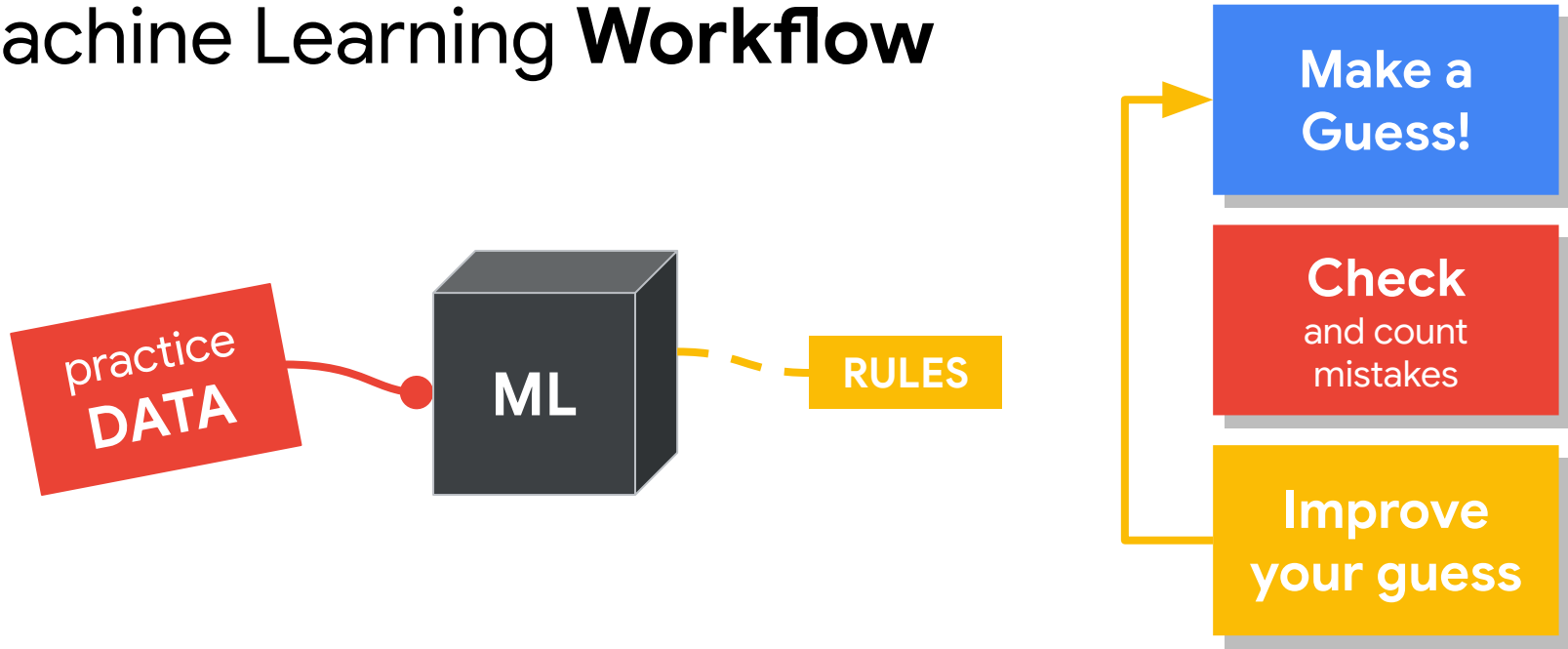


0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4
5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5
6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6
7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7
8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8
9	9	9	9	9	9	9	9	9	9	9	9	9	9	9	9

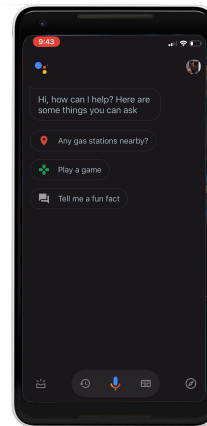
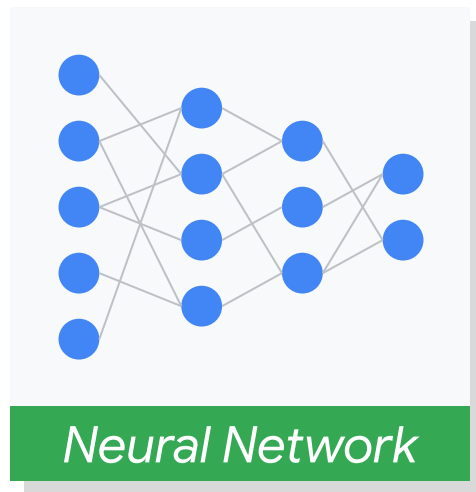
INPUTS **LABELS**



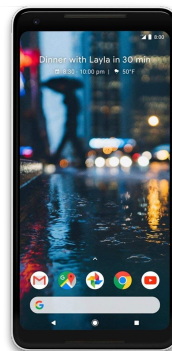
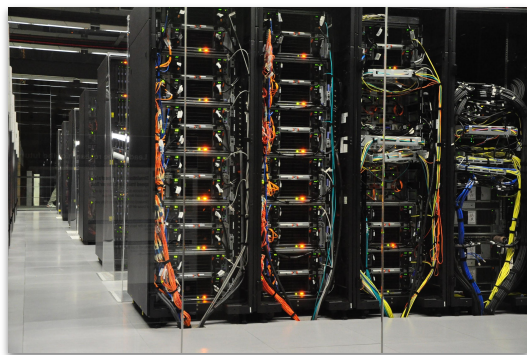
Machine Learning Workflow



Machine Learning **Workflow**



Machine Learning Workflow



Google
Assistant



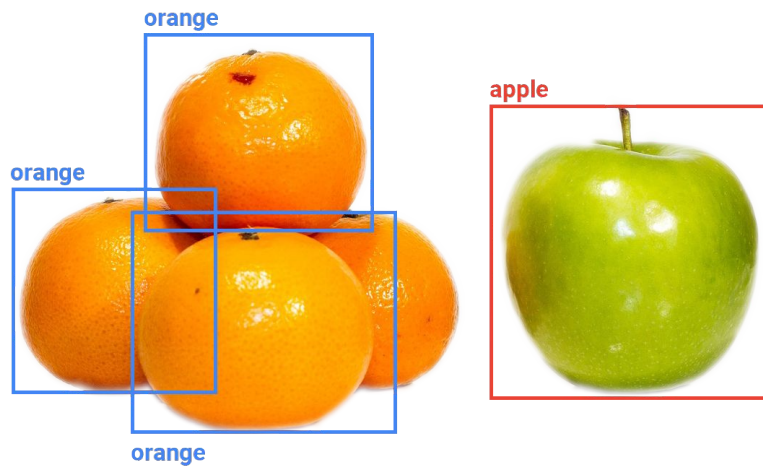
Collect &
Transform Data

Design & Train
a Model

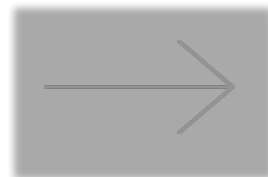
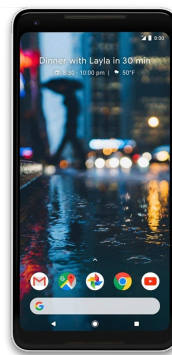
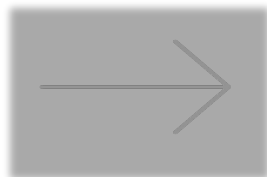
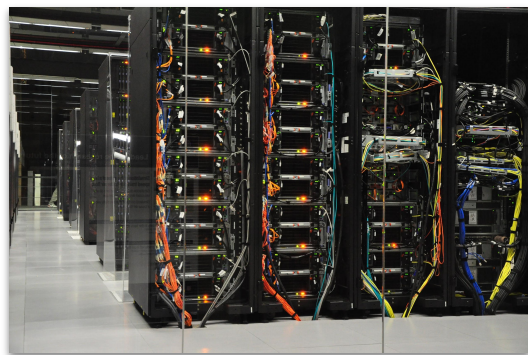
Deploy
Model

Make
Inferences

Machine Learning **Workflow**



Machine Learning Workflow



Google
Assistant



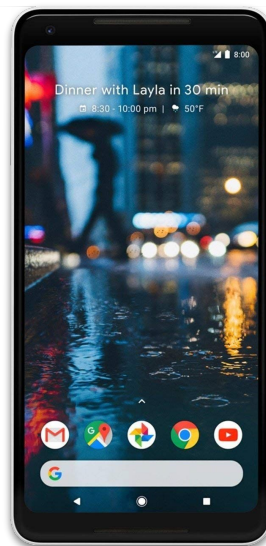
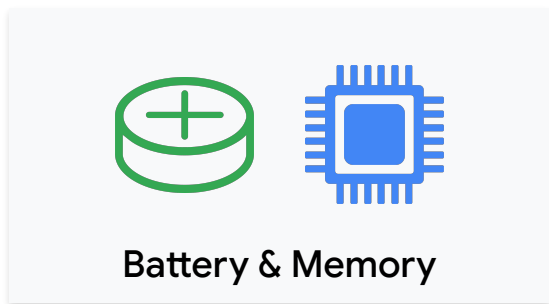
Collect &
Transform Data

Design & Train
a Model

Deploy
Model

Make
Inferences

Machine Learning **Workflow**



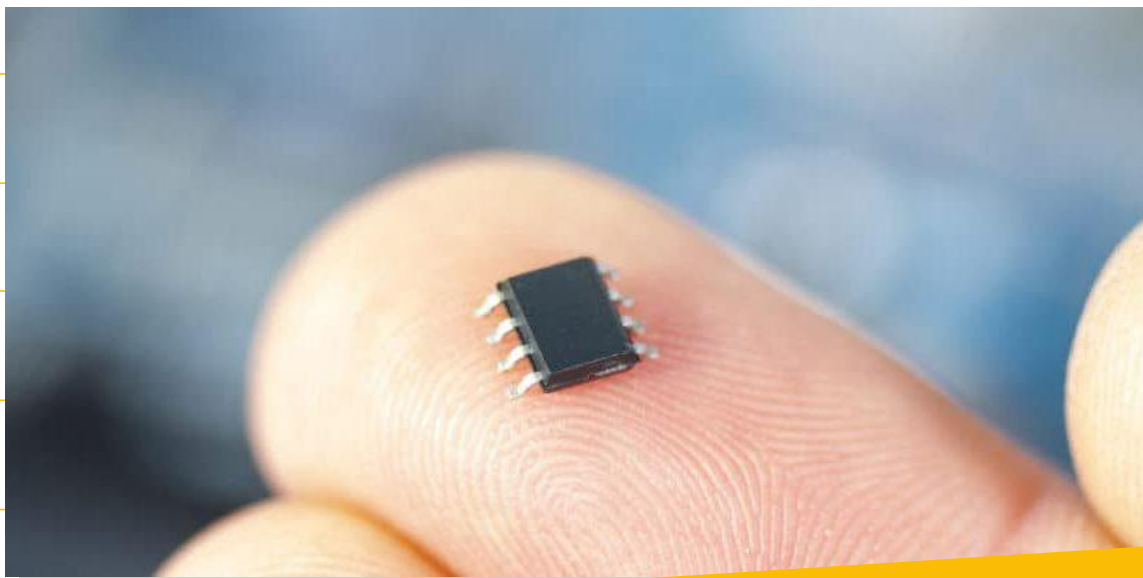
- Less memory
- Less compute power
- Only focused on *inference*





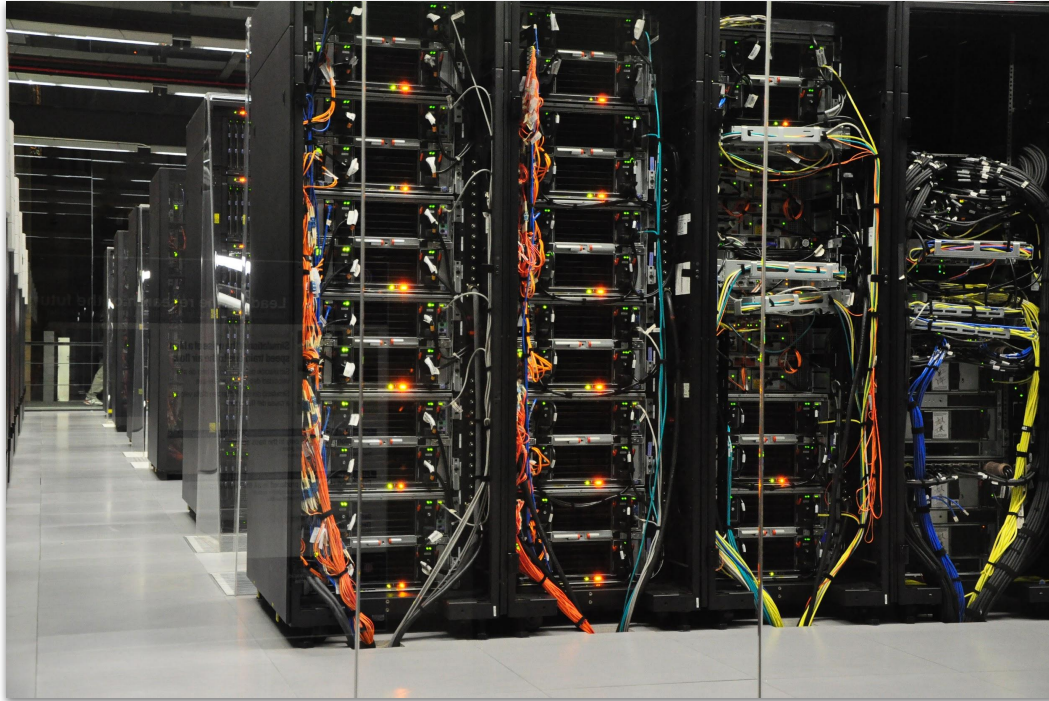
deploy

to your tiny **devices!**

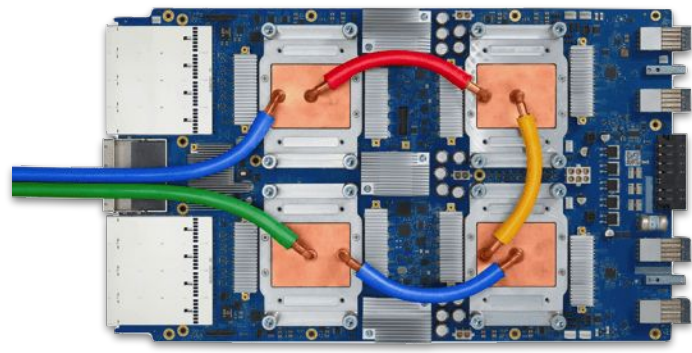


TinyML

Datacenter

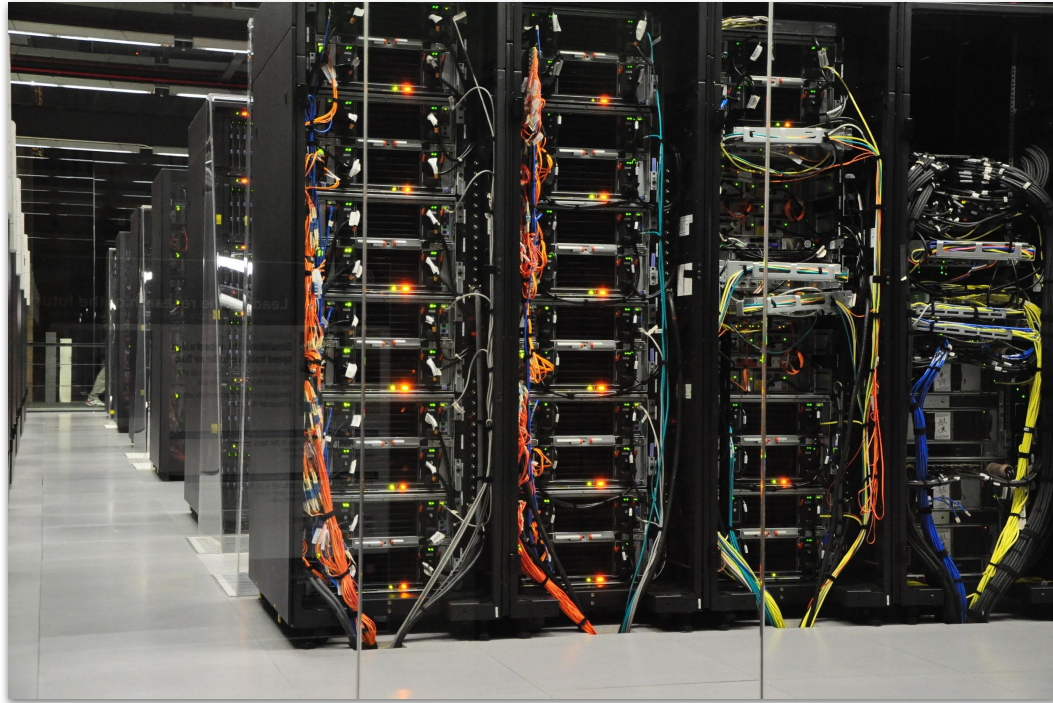


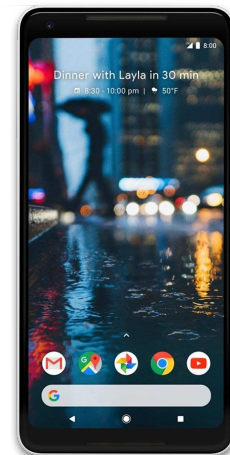
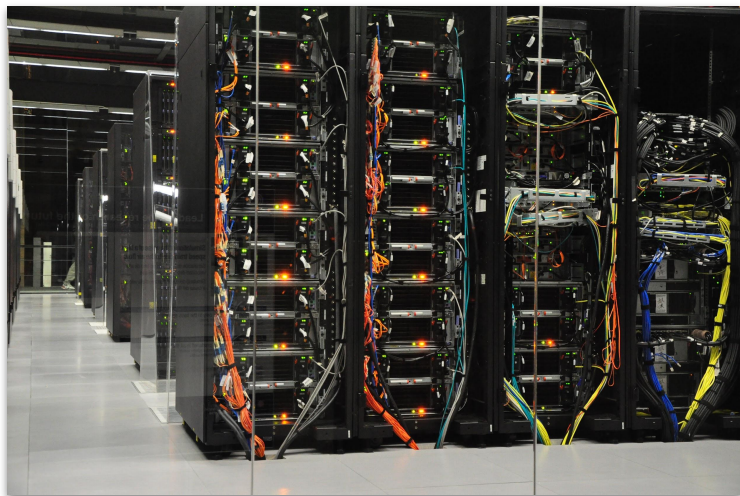
TPUs/GPUs



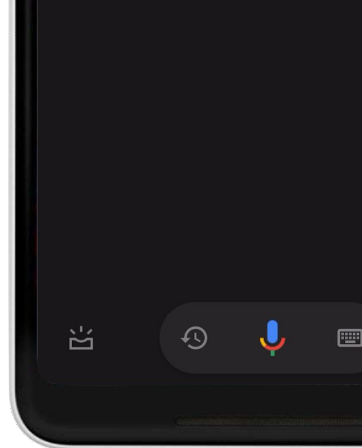
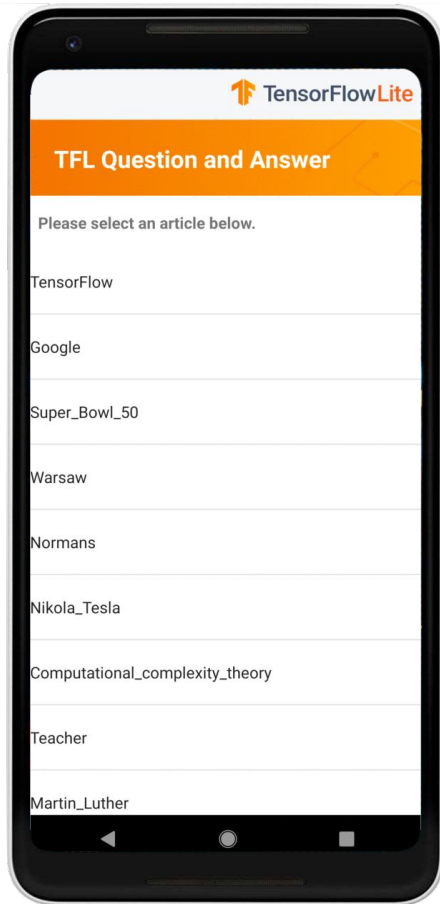
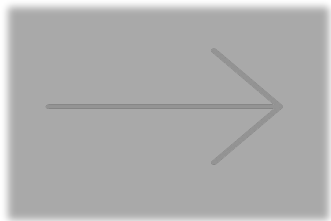


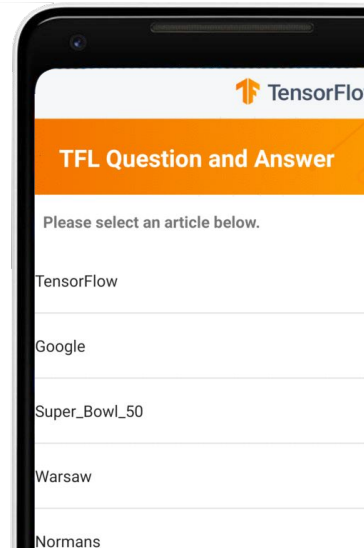
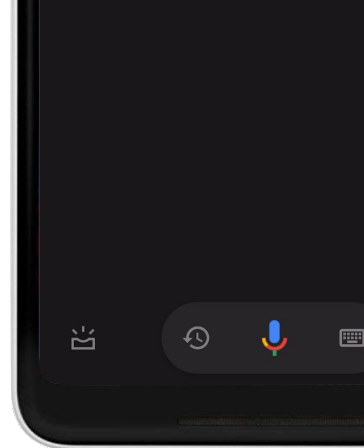
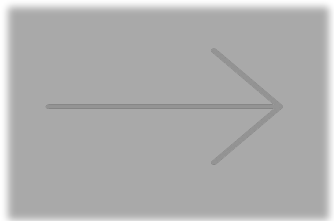
Bigger Is Not
Always Better.

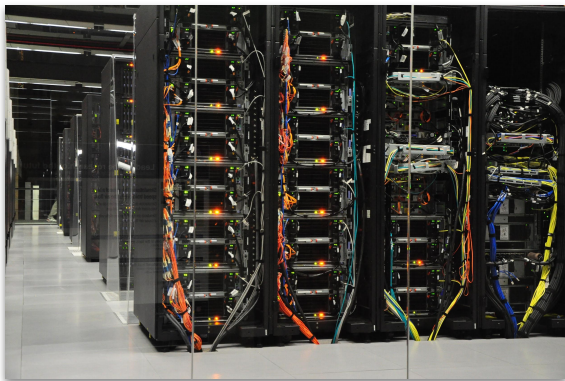










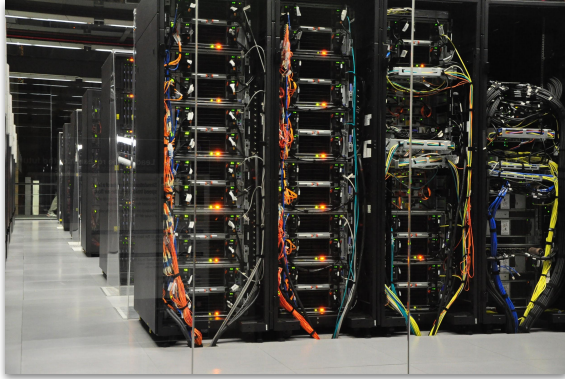


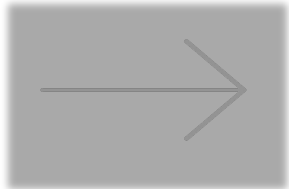
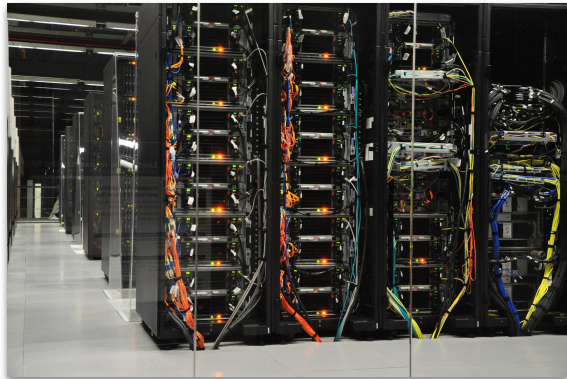
High power
High bandwidth
High latency

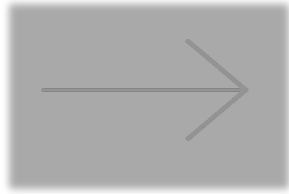
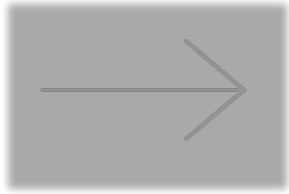
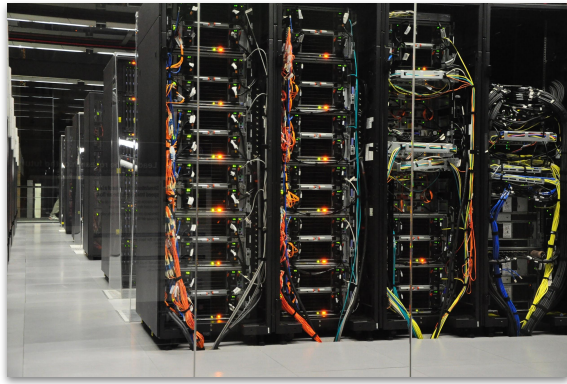


Low power
Low bandwidth
Low latency









Google Assistant



Endpoint Devices



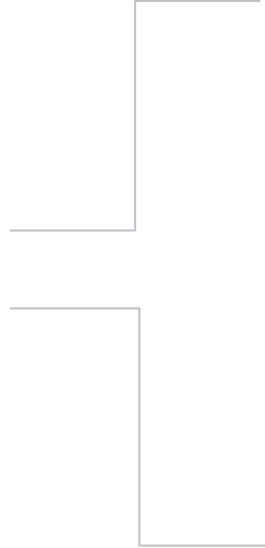
Google Assistant



Endpoint Devices



Google Assistant



Endpoints Have **Sensors**, Tons of Sensors

Motion Sensors

Gyroscope, radar,
magnetometer, accelerator

Acoustic Sensors

Ultrasonic, Microphones,
Geophones, Vibrometers

Environmental Sensors

Temperature, Humidity,
Pressure, IR, etc.

Touchscreen Sensors

Capacitive, IR

Image Sensors

Thermal, Image

Biometric Sensors

Fingerprint, Heart rate, etc.

Force Sensors

Pressure, Strain

Rotation Sensors

Encoders

Endpoints Have **Sensors**, Tons of Sensors

Motion Sensors

Gyroscope, radar,
magnetometer, accelerator

Acoustic Sensors

Ultrasonic, Microphones,
Geophones, Vibrometers

Environmental Sensors

Temperature, Humidity,
Pressure, IR, etc.

Touchscreen Sensors

Capacitive, IR

Image Sensors

Thermal, Image

Biometric Sensors

Fingerprint, Heart rate, etc.

Force Sensors

Pressure, Strain

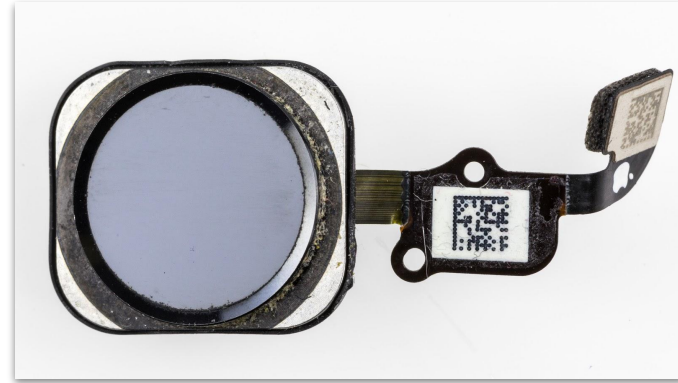
Rotation Sensors

Encoders

Biometric Sensors



Non-invasive Glucose Monitoring



Fingerprint + Photoplethysmography (PPG)

Endpoints Have **Sensors**, Tons of Sensors

Motion Sensors

Gyroscope, radar,
magnetometer, accelerator

Acoustic Sensors

Ultrasonic, Microphones,
Geophones, Vibrometers

Environmental Sensors

Temperature, Humidity,
Pressure, IR, etc.

Touchscreen Sensors

Capacitive, IR

Image Sensors

Thermal, Image

Biometric Sensors

Fingerprint, Heart rate, etc.

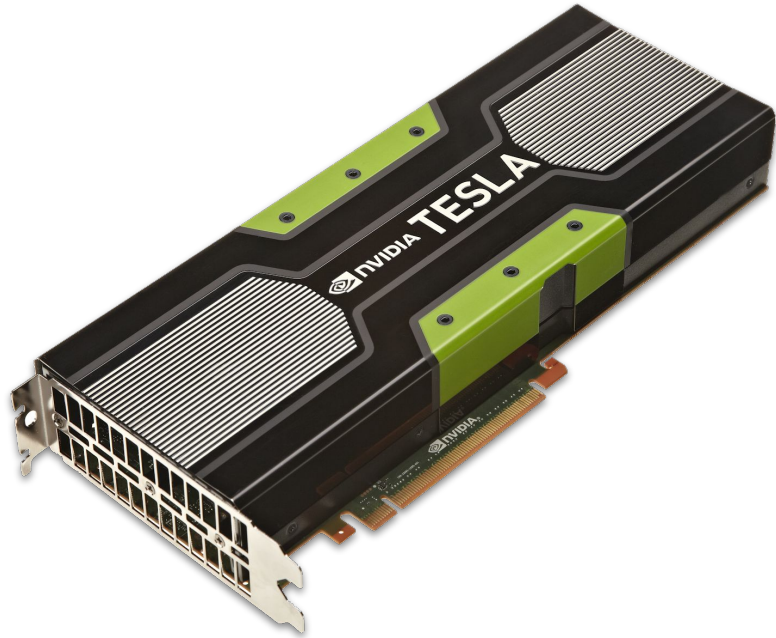
Force Sensors

Pressure, Strain

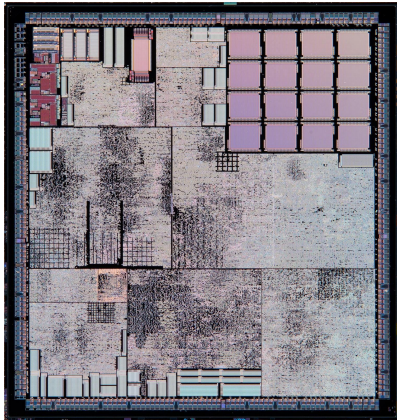
Rotation Sensors

Encoders

Thinking **Big**



Thinking Big



Thinking Big



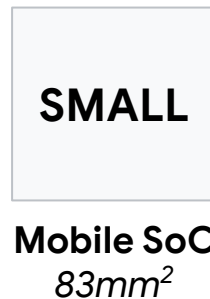
Thinking Small



Thinking Small



Thinking Small



Thinking Tiny



Mobile SoC
 83mm^2



Thinking Tiny



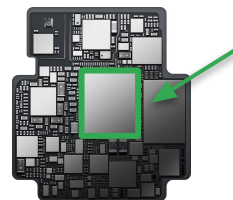
Mobile SoC
 83mm^2



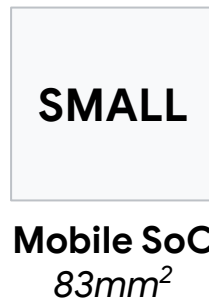
Thinking Tiny



Mobile SoC
 83mm^2

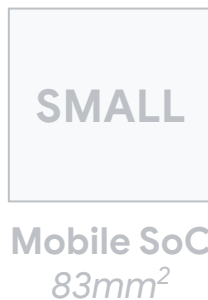


Thinking Tiny

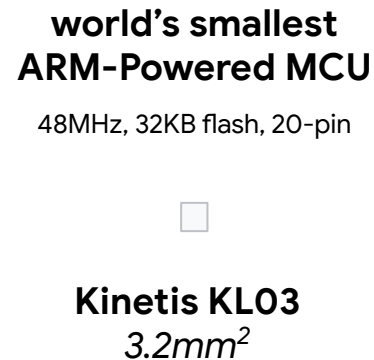
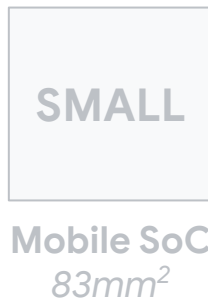


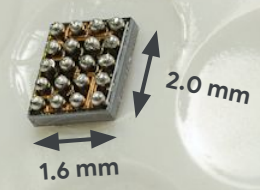
We're just getting started.

Thinking Record-breaking



Thinking Record-breaking





1.6 mm

2.0 mm

250 Billion
today

Challenges



Latency & Bandwidth



Accuracy & Personalization



Security & Privacy



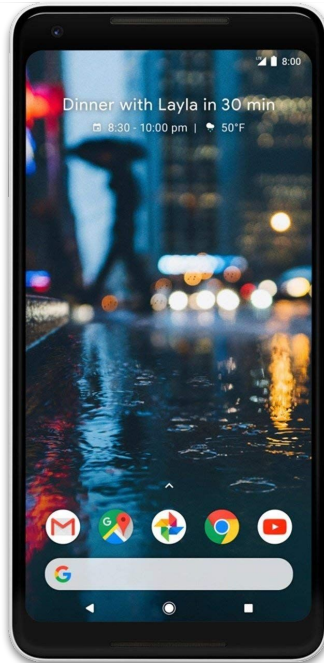
Battery & Memory



Source: Google



Source: Google



Less memory

Less compute power

Only focused on *inference*

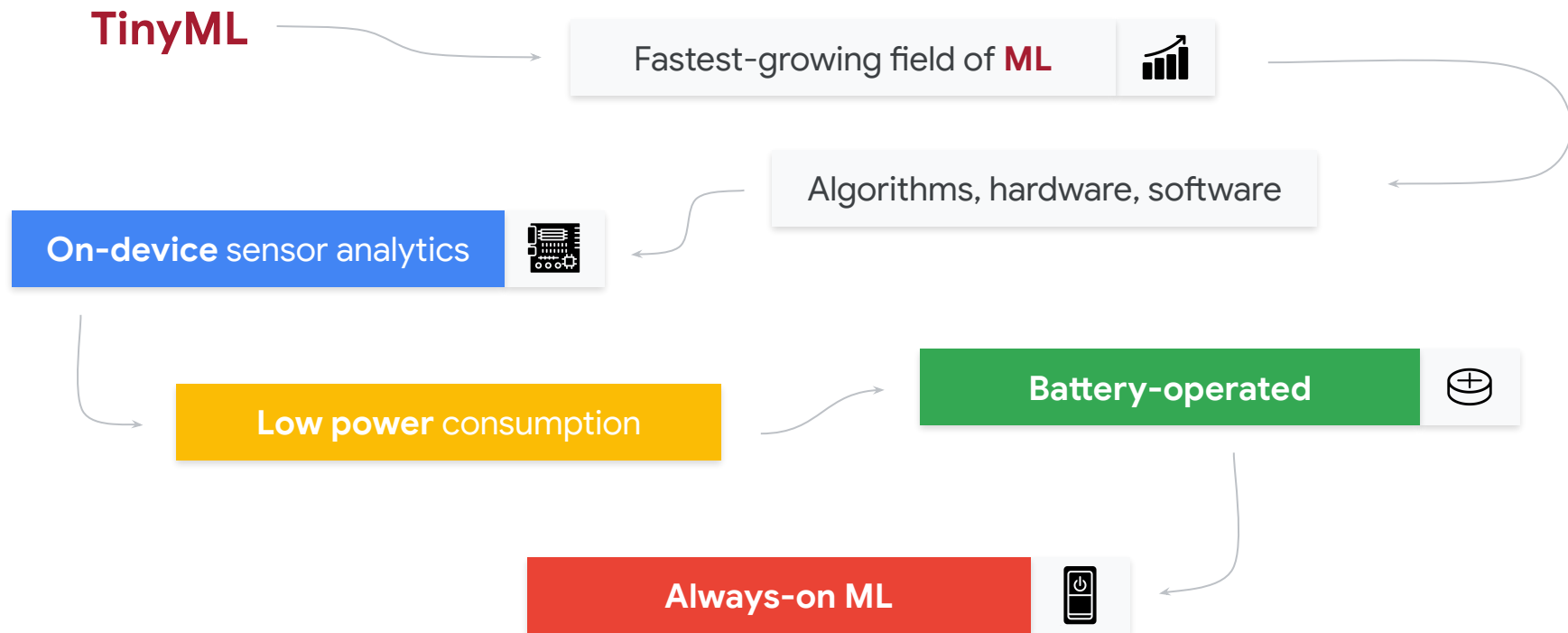


Even less memory

Even less compute power

Also, only focused on *inference*

What is Tiny Machine Learning (**TinyML**)?





Measuring capacity by watt-hours lets us compare any type of battery



1

12 V car battery

1 battery
x (12 V x 50 Ah)

600 Wh

≈



133

9 V batteries

133 batteries
x (9 V x .5 Ah)

600 Wh

≈



67

3.6 V smartphones

67 smartphones
x (3.6 V x 2.5 Ah)

600 Wh

TYPICAL BATTERY LIFE



50KB EVERY
2 HOURS



200KB EVERY
2 HOURS



50KB EVERY
24 HOURS



200KB EVERY
24 HOURS

Data Usage

TYPICAL BATTERY LIFE



50KB EVERY
2 HOURS



200KB EVERY
2 HOURS



50KB EVERY
24 HOURS



200KB EVERY
24 HOURS

Data Usage

**transmit less
data!**

Today's Agenda

- Review + Why Tiny?
- Hardware Basics
- Installing and Starting the Arduino IDE
- Testing Your TinyML Kit
- Deploying KWS model onto Arduino
- Summary & Next Steps

Today's Agenda

- Review + Why Tiny?

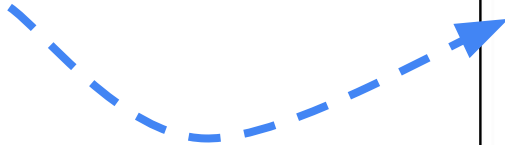
Hardware Basics

- Installing and Starting the Arduino IDE
- Testing Your TinyML Kit
- Deploying KWS model onto Arduino
- Summary & Next Steps

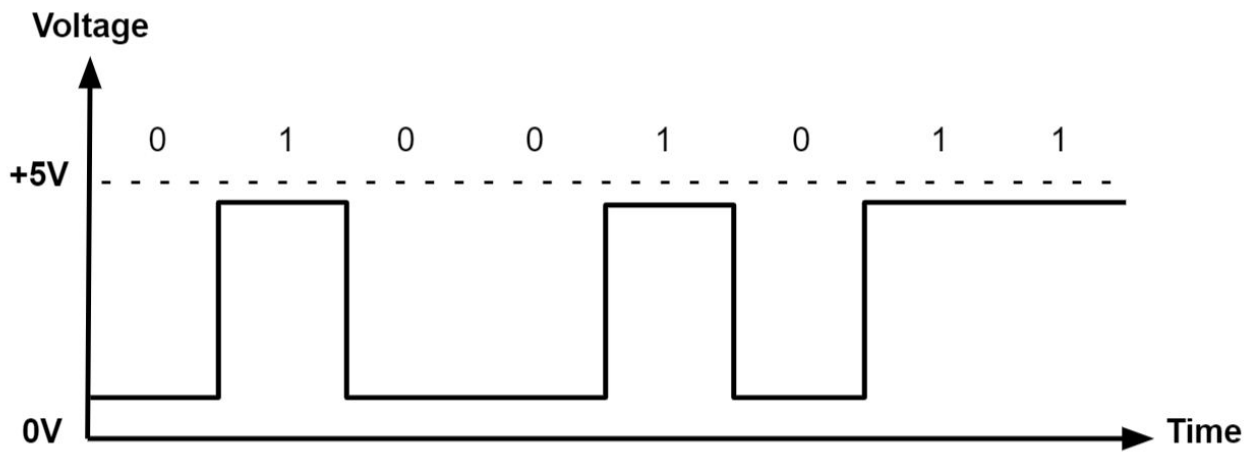


language of
computers

Binary	Hex	Decimal
0000	0	0
0001	1	1
0010	2	2
0011	3	3
0100	4	4
0101	5	5
0110	6	6
0111	7	7
1000	8	8
1001	9	9
1010	A	10
1011	B	11
1100	C	12
1101	D	13
1110	E	14
1111	F	15





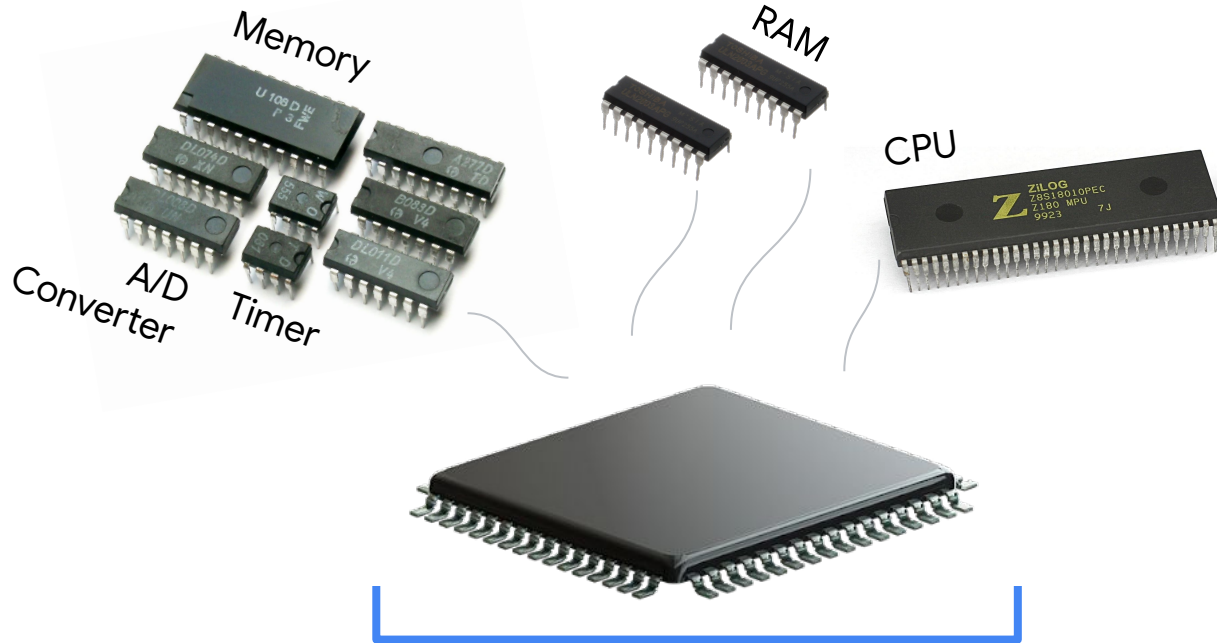








Microcontroller

CPU	Read-Only Memory (ROM)	Read-Write Memory
Timer	I/O Port	Serial Interface





Microcontroller: a **complete package**

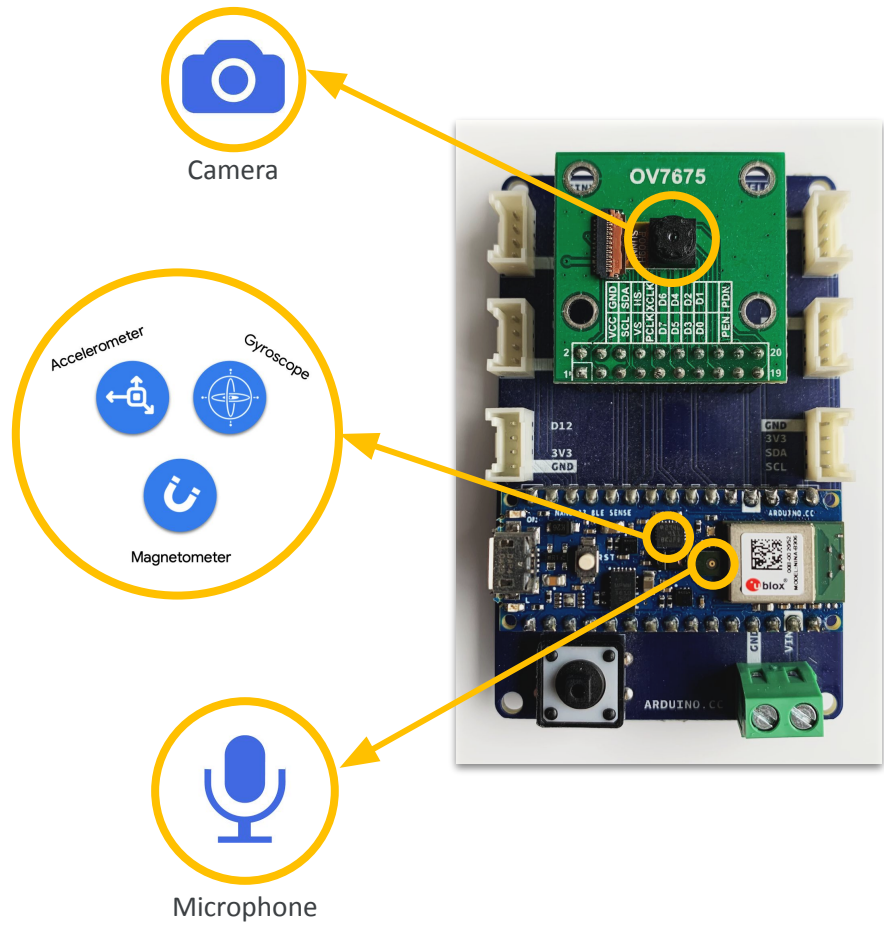


Embedded Systems

	Board	MCU / ASIC	Clock	Memory	Sensors	Radio
	Himax WE-I Plus EVB	HX6537-A 32-bit EM9D DSP	400 MHz	2MB flash 2MB RAM	Accelerometer, Mic, Camera	None
	Arduino Nano 33 BLE Sense	32-bit nRF52840	64 MHz	1MB flash 256kB RAM	Mic, IMU, Temp, Humidity, Gesture, Pressure, Proximity, Brightness, Color	BLE
	SparkFun Edge 2	32-bit ArtemisV1	48 MHz	1MB flash 384kB RAM	Accelerometer, Mic, Camera	BLE
	Espressif EYE	32-bit ESP32-D0WD	240 MHz	4MB flash 520kB RAM	Mic, Camera	WiFi, BLE

Embedded Systems

Board	MCU / ASIC	Clock	Memory	Sensors	Radio
 Himax WE-I Plus EVB	HX6537-A 32-bit EM9D DSP	400 MHz	2MB flash 2MB RAM	Accelerometer, Mic, Camera	None
 Arduino Nano 33 BLE Sense	32-bit nRF52840	64 MHz	1MB flash 256kB RAM	Mic, IMU, Temp, Humidity, Gesture, Pressure, Proximity, Brightness, Color	BLE
 SparkFun Edge 2	32-bit ArtemisV1	48 MHz	1MB flash 384kB RAM	Accelerometer, Mic, Camera	BLE
 Espressif EYE	32-bit ESP32-D0WD	240 MHz	4MB flash 520kB RAM	Mic, Camera	WiFi, BLE





Nano 33 BLE Sense (+ USB cable)



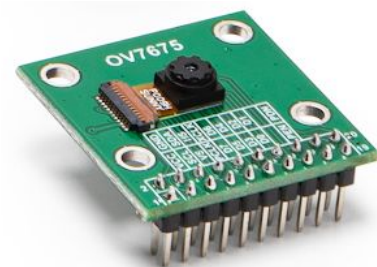
Purpose

AI-enabled developmental **microcontroller board** with USB-A to microB cable

Specifications

- nRF52840 MCU (ARM Cortex-M4): 3.3V, 64MHz, 1MB flash, 256 kB RAM
- Sensors on board: microphone, IMU, color, light, proximity, temperature, humidity, and more!
- BLE module with application-adjacent protocol layers (GAP, GATT) covered by **ArduinoBLE** library

OV 7675 Camera Module



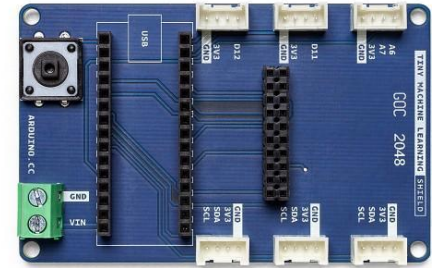
Purpose

Breakout PCB for *tiny* camera, for use in person-detection exercises

Specifications

- Low-voltage, 0.3 MP CMOS VGA (can step down to QVGA, QQVGA) image sensor
- Serial Camera Control Bus (SCCB) + Camera Parallel Interface (CPI) / Digital Video Port (DVP) interface
- Breaks ribbon cable out to 2x10 pin array

Tiny Machine Learning Shield

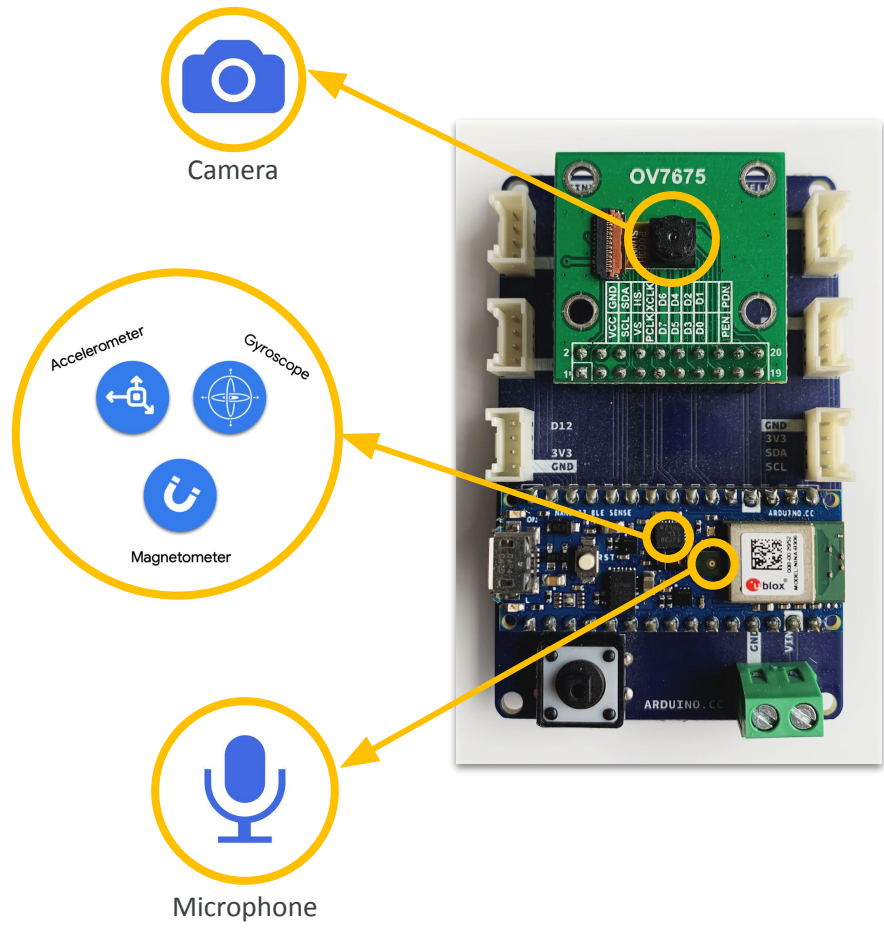


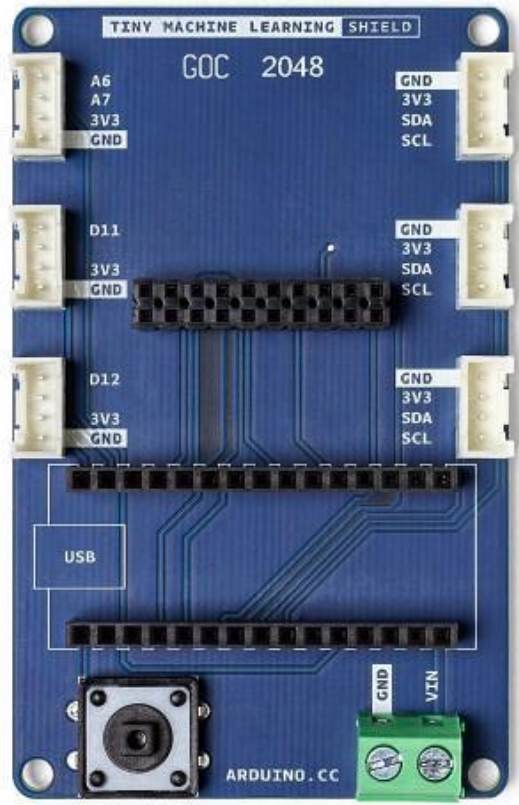
Purpose

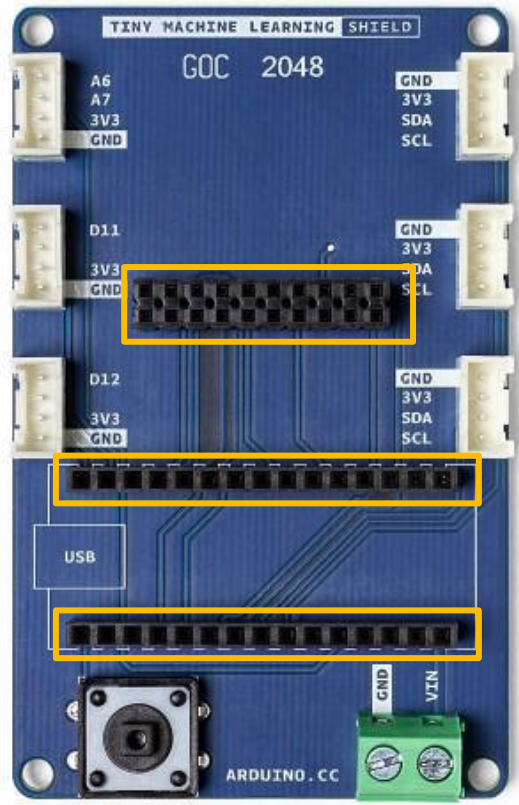
A daughter PCB designed to **breakout the I/O** from the Nano 33 BLE sense to permit easy, reliable **communication with** other local, **off-board elements**

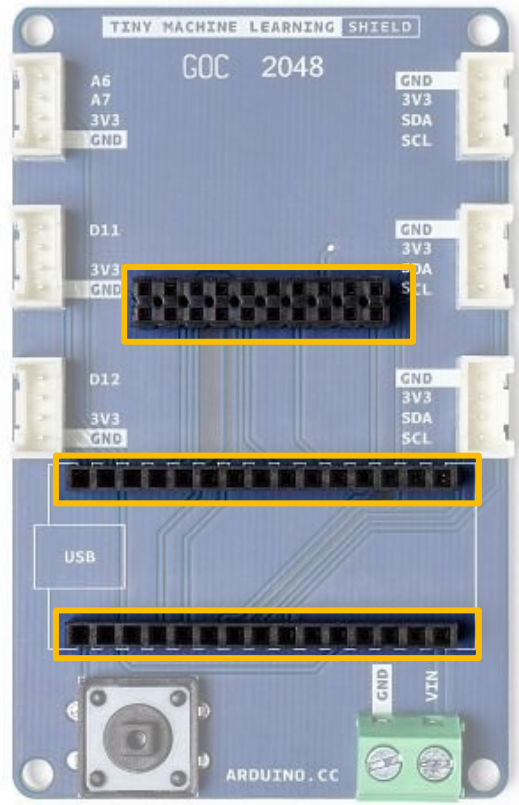
Specifications

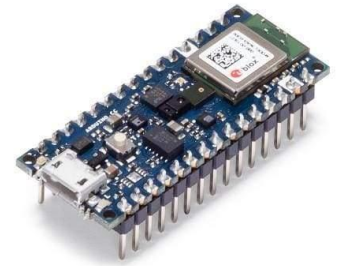
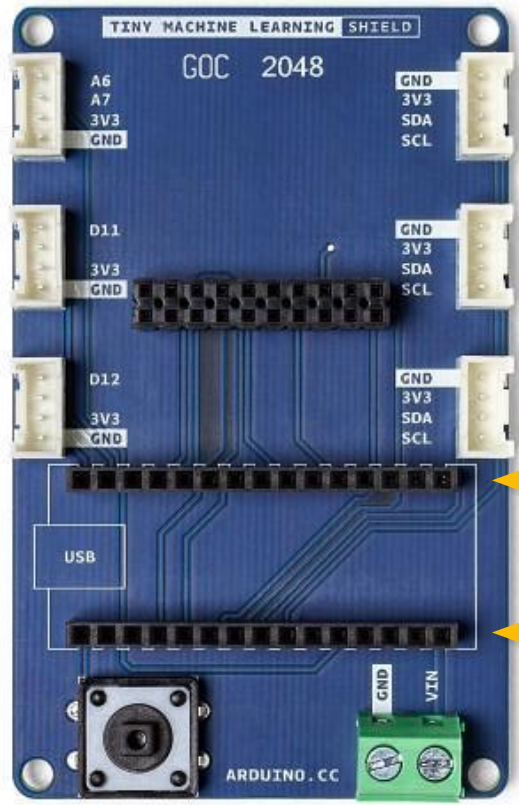
- Grove connectors (3.3V I2C and simple digital / analog - see pinouts)
- 2x10 pin array for OV7675 camera module
- Voltage input terminal block, accepts 4.5 to 21V (down regulated to 3.3V on Nano 33)

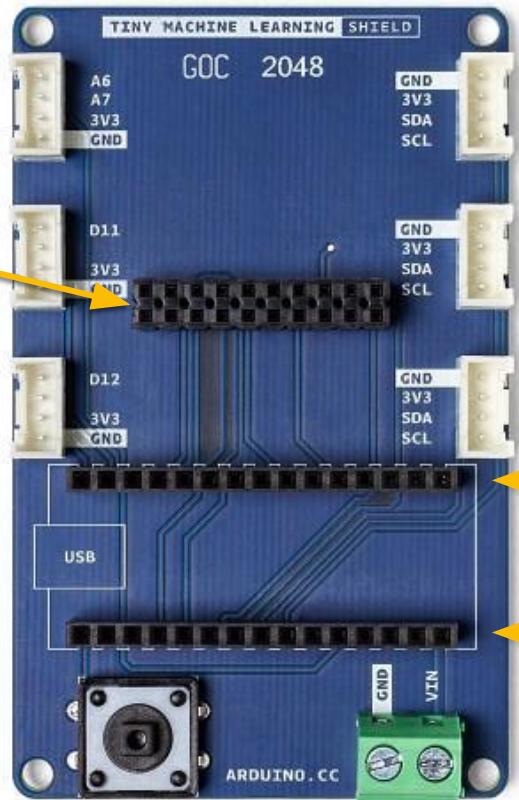
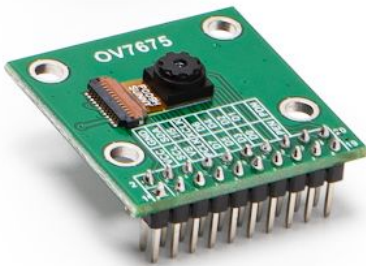


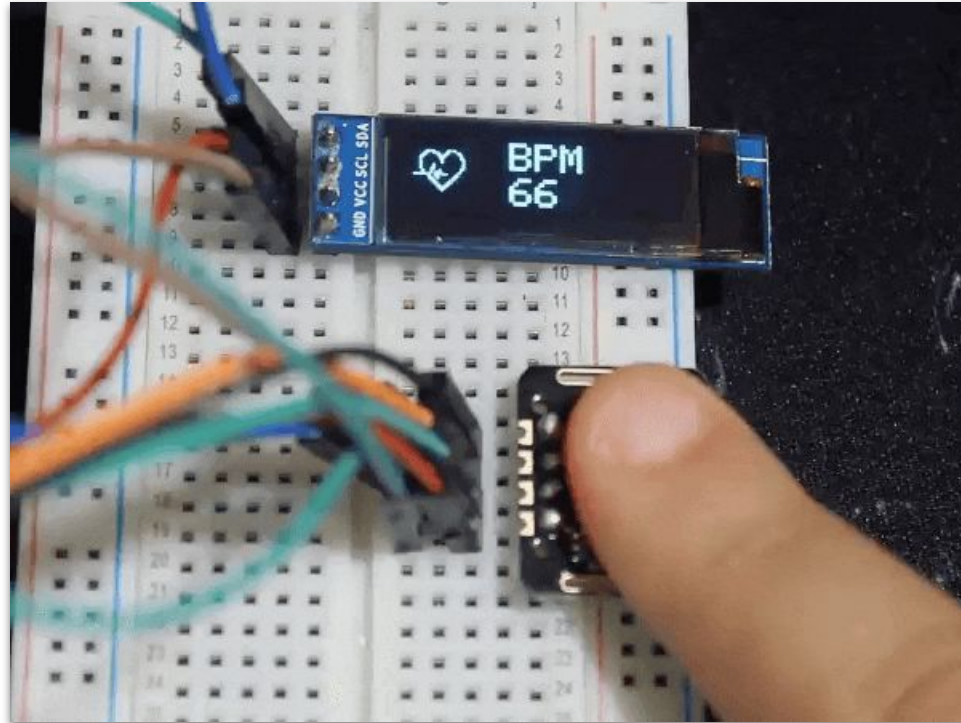














Today's Agenda

- Review + Why Tiny?
- Hardware Basics
- Installing and Starting the Arduino IDE
- Testing Your TinyML Kit
- Deploying KWS model onto Arduino
- Summary & Next Steps

Today's Agenda

- Review + Why Tiny?
- Hardware Basics

Installing and Starting the Arduino IDE

- Testing Your TinyML Kit
- Deploying KWS model onto Arduino
- Summary & Next Steps



Downloads



Arduino IDE 1.8.19

The open-source Arduino Software (IDE) makes it easy to write code and upload it to the board. This software can be used with any Arduino board.

Refer to the [Getting Started](#) page for Installation instructions.

SOURCE CODE

Active development of the Arduino software is [hosted by GitHub](#). See the instructions for [building the code](#). Latest release source code archives are available [here](#). The archives are PGP-signed so they can be verified using [this](#) gpg key.

DOWNLOAD OPTIONS

Windows Win 7 and newer

Windows ZIP file

Windows app Win 8.1 or 10



Linux 32 bits

Linux 64 bits

Linux ARM 32 bits

Linux ARM 64 bits

Mac OS X 10.10 or newer

[Release Notes](#)

[Checksums \(sha512\)](#)

Downloads



Arduino IDE 1.8.19

The open-source Arduino Software (IDE) makes it easy to write code and upload it to the board. This software can be used with any Arduino board.

Refer to the [Getting Started](#) page for Installation instructions.

SOURCE CODE

Active development of the Arduino software is [hosted by GitHub](#). See the instructions for [building the code](#). Latest release source code archives are available [here](#). The archives are PGP-signed so they can be verified using [this](#) gpg key.

DOWNLOAD OPTIONS

Windows Win 7 and newer

Windows ZIP file

Windows app Win 8.1 or 10



Linux 32 bits

Linux 64 bits

Linux ARM 32 bits

Linux ARM 64 bits

Mac OS X 10.10 or newer

[Release Notes](#)

[Checksums \(sha512\)](#)

HARDWARE

SOFTWARE

CLOUD

DO

www.arduino.cc/en/software

Downloads



Arduino IDE 1.8.19

The open-source Arduino Software (IDE) makes it easy to write code and upload it to the board. This software can be used with any Arduino board.

Refer to the [Getting Started](#) page for Installation instructions.

SOURCE CODE

Active development of the Arduino software is [hosted by GitHub](#). See the instructions for [building the code](#). Latest release source code archives are available [here](#). The archives are PGP-signed so they can be verified using [this](#) gpg key.

DOWNLOAD OPTIONS

Windows Win 7 and newer

Windows ZIP file

Windows app Win 8.1 or 10



Linux 32 bits

Linux 64 bits

Linux ARM 32 bits

Linux ARM 64 bits

Mac OS X 10.10 or newer

[Release Notes](#)

[Checksums \(sha512\)](#)

Getting Started with Arduino products

Welcome to Arduino! Before you start controlling the world around you, you'll need to set up the software to program your board.

The Arduino Software (IDE) allows you to write programs and upload them to your board. In the [Arduino Software](#) page you will find two options:

1. If you have a reliable Internet connection, you should use the [online IDE](#) (Arduino Web Editor). It will allow you to save your sketches in the cloud, having them available from any device and backed up. You will always have the most up-to-date version of the IDE without the need to install updates or community generated libraries.
2. If you would rather work offline, you should use the latest version of the [desktop IDE](#).

Code online on the Arduino Web Editor

To use the online IDE simply follow [these instructions](#). Remember that boards work out-of-the-box on the [Web Editor](#), you only need to install Arduino Create Agent to get started.

Install the Arduino Desktop IDE

To get step-by-step instructions select one of the following link accordingly to your operating system.

- [Windows](#)
- [Mac OS](#)
- [Linux](#)
- [Portable IDE](#) (Windows and Linux)
- [ChromeOS](#) (Arduino Create App) in the [Chrome Web Store](#)

Choose your board in the list here on the right to learn how to get started with it and how to use it on the Desktop IDE.

Learn Arduino

- Read an [introduction](#) on what is Arduino and why you'd want to use it.
- What is the [Arduino Software \(IDE\)](#) and how do I change the default language?

Arduino IoT Cloud:

[Getting Started with Arduino IoT Cloud](#)

Instructions for our boards:

[Due](#)

[MEGA2560](#)

[MKR1000](#)

[MKR WiFi 1010](#)

[MKR FOX 1200](#)

[MKR WAN 1300](#)

[MKR WAN 1310](#)

[MKR GSM 1400](#)

[MKR NB 1500](#)

[MKR Vidor 4000](#)

[MKRZERO](#)

[Nano](#)

[NANO 33 IoT](#)

[NANO 33 BLE](#)

[NANO 33 BLE Sense](#)

Getting Started with Arduino products

Welcome to Arduino! Before you start controlling the world around you, you'll need to set up the software to program your board.

The Arduino Software (IDE) allows you to write programs and upload them to your board. In the [Arduino Software](#) page you will find two options:

1. If you have a reliable Internet connection, you should use the [online IDE](#) (Arduino Web Editor). It will allow you to save your sketches in the cloud, having them available from any device and backed up. You will always have the most up-to-date version of the IDE without the need to install updates or community generated libraries.
2. If you would rather work offline, you should use the latest version of the [desktop IDE](#).

Code online on the Arduino Web Editor

To use the online IDE simply follow [these instructions](#). Remember that boards work out-of-the-box on the [Web Editor](#), you only need to install Arduino Create Agent to get started.

Install the Arduino Desktop IDE

To get step-by-step instructions select one of the following link accordingly to your operating system.

- [Windows](#)
 - [Mac OS](#)
 - [Linux](#)
- [Portable IDE](#) (Windows and Linux)
 - [ChromeOS](#) (Arduino Create App) in the [Chrome Web Store](#)

Choose your board in the list here on the right to learn how to get started with it and how to use it on the Desktop IDE.

Learn Arduino

- Read an [introduction](#) on what is Arduino and why you'd want to use it.
- What is the [Arduino Software \(IDE\)](#) and how do I change the default language?

Arduino IoT Cloud:

[Getting Started with Arduino IoT Cloud](#)

Instructions for our boards:

[Due](#)

[MEGA2560](#)

[MKR1000](#)

[MKR WiFi 1010](#)

[MKR FOX 1200](#)

[MKR WAN 1300](#)

[MKR WAN 1310](#)

[MKR GSM 1400](#)

[MKR NB 1500](#)

[MKR Vidor 4000](#)

[MKRZERO](#)

[Nano](#)

[NANO 33 IoT](#)

[NANO 33 BLE](#)

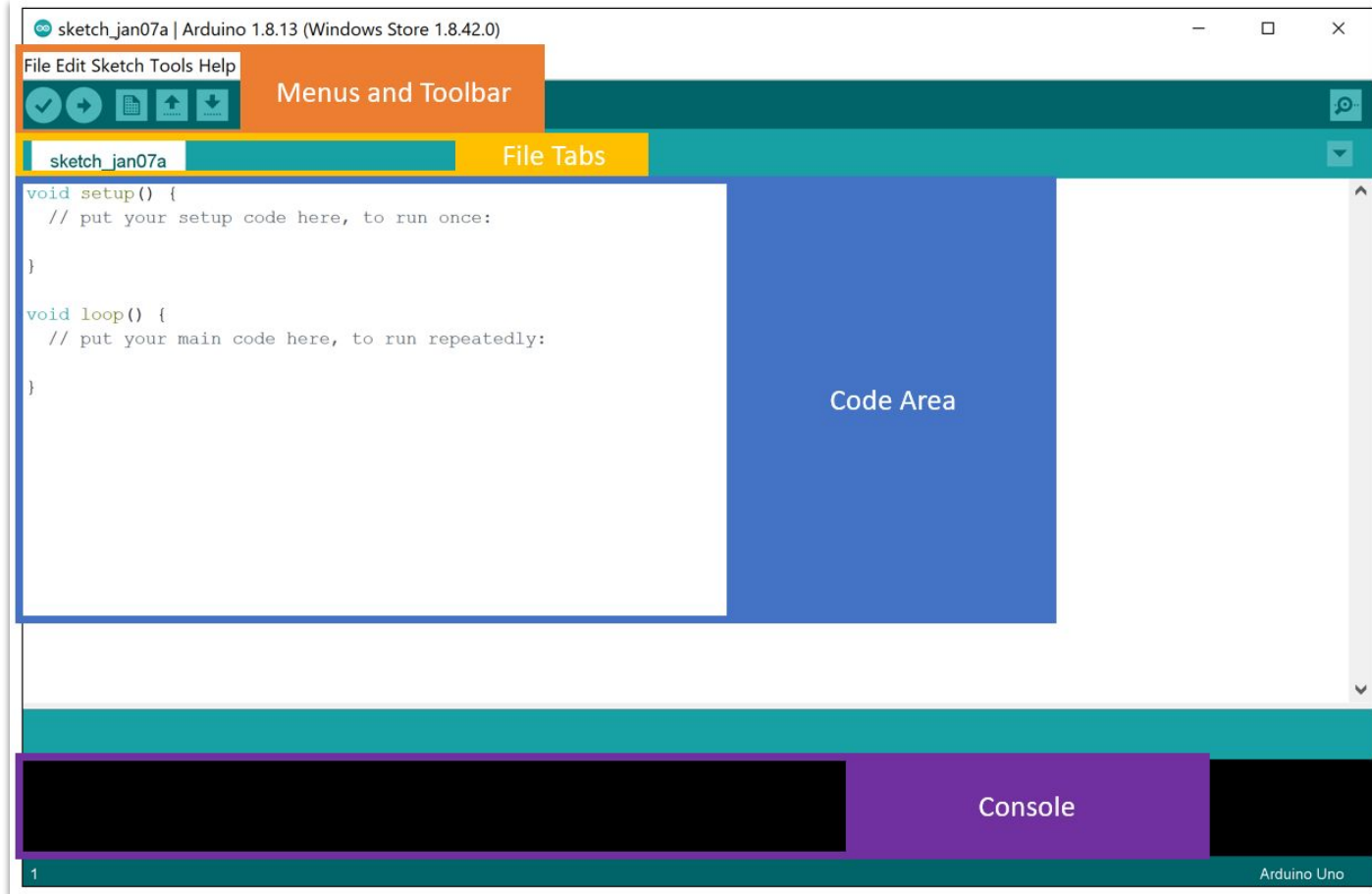
[NANO 33 BLE Sense](#)

head to Arduino's software **website**

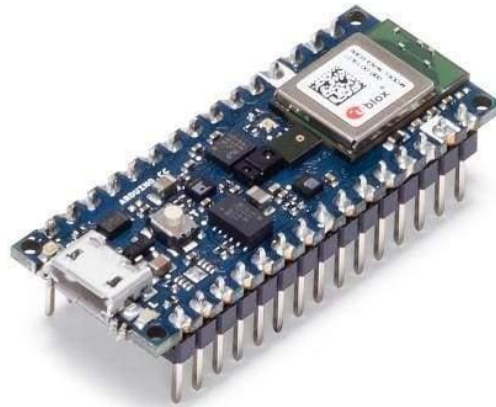
www.arduino.cc/en/software

then **download** and **install**

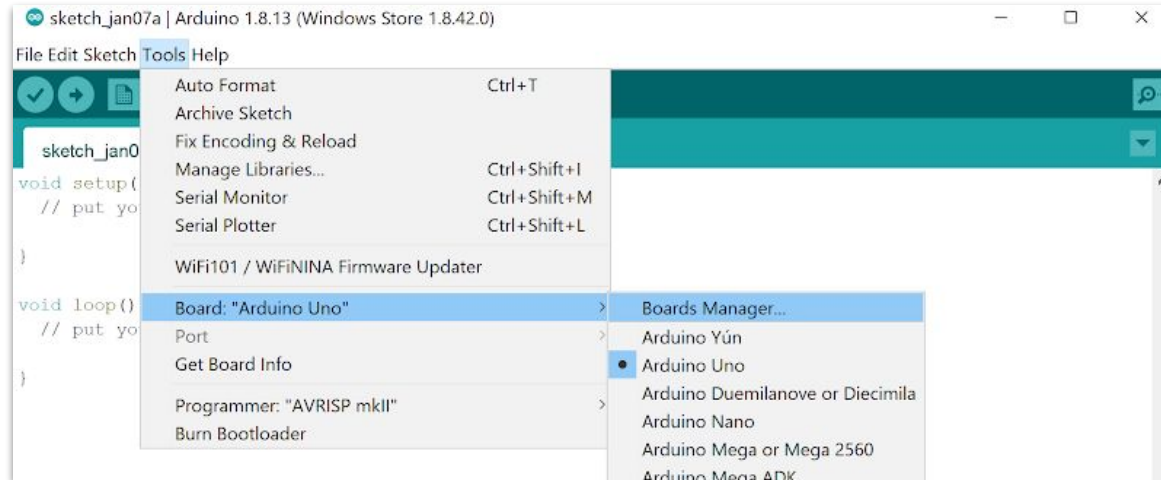
right now!



Install Extras



Tools → Board → Boards Manager



Boards Manager

Type: All Nano 33 BLE

Arduino Mbed OS Nano Boards
by **Arduino**
Boards included in this package:
Arduino Nano 33 BLE, Arduino Nano 33 BLE Sense, Arduino Nano RP2040 Connect.
[Online Help](#)
[More Info](#)

3.1.1 Install

[DEPRECATED - Please install standalone packages] Arduino Mbed OS Boards
by **Arduino DEPRECATED**
Boards included in this package:
Arduino Nano 33 BLE, Arduino Nano 33 BLE Sense, Arduino Nano RP2040 Connect, Arduino Portenta H7, Arduino Edge Control, Raspberry Pi Pico, Nida Sense ME, Arduino Nida Vision.
[Online Help](#)
[More Info](#)

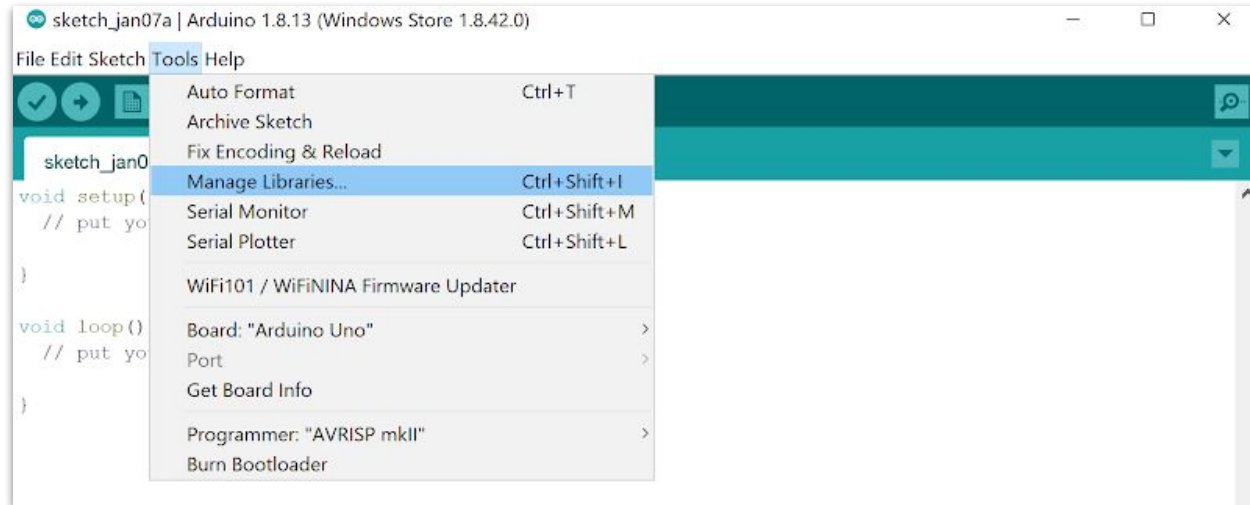
Close

The screenshot shows the Arduino Boards Manager interface. At the top, the title bar reads "Boards Manager". Below it, there is a search bar with "Nano 33 BLE" entered. The main content area displays two packages:

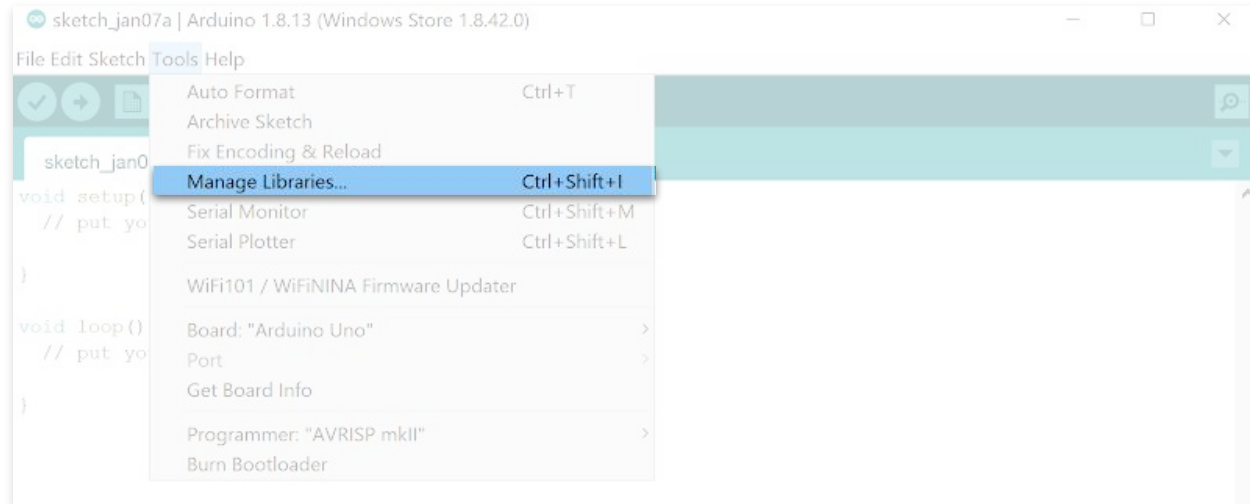
- Arduino Mbed OS Nano Boards** by **Arduino**. It lists boards included in the package: Arduino Nano 33 BLE, Arduino Nano 33 BLE Sense, and Arduino Nano RP2040 Connect. It includes links for [Online Help](#) and [More Info](#). To the right of this package, there is a version dropdown menu set to "3.1.1" and an "Install" button. A green callout box with the text "3.1.1" and an arrow points to the version dropdown.
- [DEPRECATED - Please install standalone packages] Arduino Mbed OS** by **Arduino DEPRECATED**. It lists boards included in this package: Arduino Nano 33 BLE, Arduino Nano 33 BLE Sense, Arduino Nano RP2040 Connect, Arduino Portenta H7, Arduino Edge Control, Raspberry Pi Pico, Nida Sense ME, and Arduino Nida Vision. It also includes links for [Online Help](#) and [More Info](#).

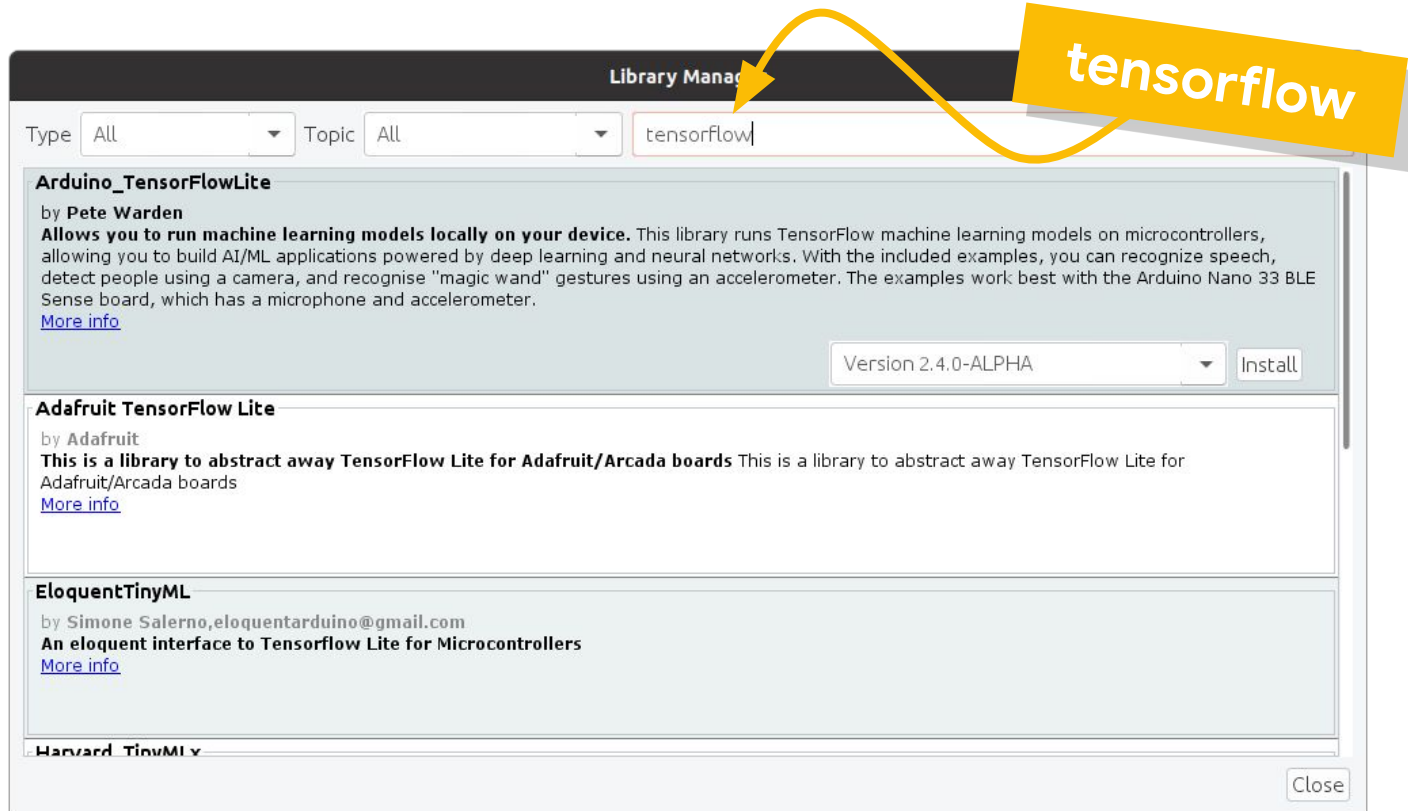
A yellow callout box with the text "Nano 33 BLE" and an arrow points to the package name in the search results. A "Close" button is located in the bottom right corner of the window.

Tools → Manage Libraries...



Tools → Manage Libraries...





Library Manager

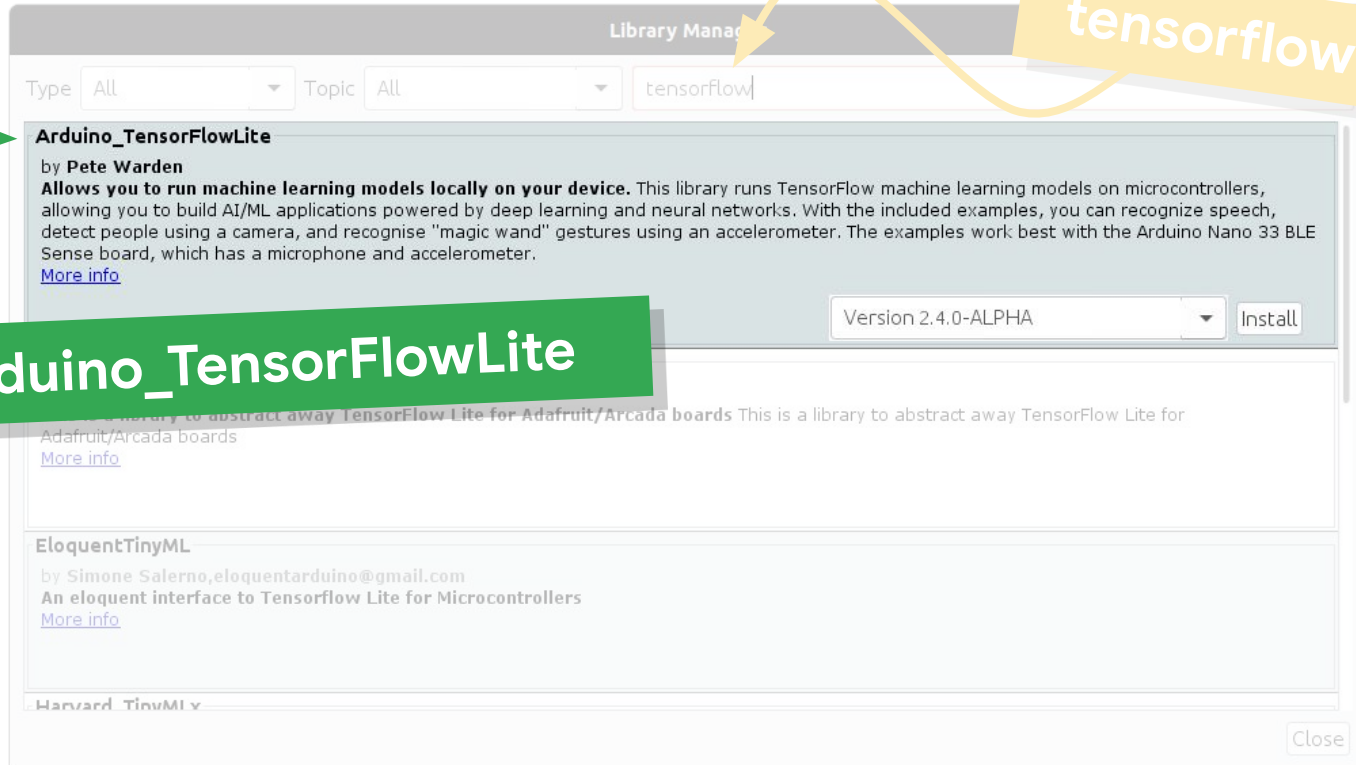
Type: All Topic: All

Arduino_TensorFlowLite
by Pete Warden
Allows you to run machine learning models locally on your device. This library runs TensorFlow machine learning models on microcontrollers, allowing you to build AI/ML applications powered by deep learning and neural networks. With the included examples, you can recognize speech, detect people using a camera, and recognise "magic wand" gestures using an accelerometer. The examples work best with the Arduino Nano 33 BLE Sense board, which has a microphone and accelerometer.
[More info](#)
Version 2.4.0-ALPHA

Adafruit TensorFlow Lite
by Adafruit
This is a library to abstract away TensorFlow Lite for Adafruit/Arcada boards This is a library to abstract away TensorFlow Lite for Adafruit/Arcada boards
[More info](#)

EloquentTinyML
by Simone Salerno, eloquentarduino@gmail.com
An eloquent interface to Tensorflow Lite for Microcontrollers
[More info](#)

Harvard_TinyML v



The screenshot shows the Arduino Library Manager interface. At the top, the title is "Library Manager". Below it, there are filters for "Type" (set to "All") and "Topic" (set to "All"). A search bar contains the text "tensorflow". The search results are displayed in a list. The first result is "Arduino_TensorFlowLite" by Pete Warden. The description for this library states: "Allows you to run machine learning models locally on your device. This library runs TensorFlow machine learning models on microcontrollers, allowing you to build AI/ML applications powered by deep learning and neural networks. With the included examples, you can recognize speech, detect people using a camera, and recognise 'magic wand' gestures using an accelerometer. The examples work best with the Arduino Nano 33 BLE Sense board, which has a microphone and accelerometer." Below the description is a "More info" link. To the right of the description, there is a dropdown menu showing "Version 2.4.0-ALPHA" and an "Install" button. Below this result, there is a partially visible result for "Library to abstract away TensorFlow Lite for Adafruit/Arcada boards" with a "More info" link. The second result is "EloquentTinyML" by Simone Salerno, with the description "An eloquent interface to Tensorflow Lite for Microcontrollers" and a "More info" link. The third result is "Harvard_TinyML" which is partially cut off. At the bottom right of the window, there is a "Close" button.

tensorflow

Arduino_TensorFlowLite

Library Manager

Type All Topic All tensorflow

Arduino_TensorFlowLite
by Pete Warden
Allows you to run machine learning models locally on your device. This library runs TensorFlow machine learning models on microcontrollers, allowing you to build AI/ML applications powered by deep learning and neural networks. With the included examples, you can recognize speech, detect people using a camera, and recognise "magic wand" gestures using an accelerometer. The examples work best with the Arduino Nano 33 BLE Sense board, which has a microphone and accelerometer.
[More info](#)

Version 2.4.0-ALPHA Install

Adafruit TensorFlow Lite
by Adafruit
This is a library to abstract away TensorFlow Lite for Adafruit/Arcada boards This is a library to abstract away TensorFlow Lite for Adafruit/Arcada boards
[More info](#)

EloquentTinyML
by Simone Salerno, eloquentarduino@gmail.com
An eloquent interface to Tensorflow Lite for Microcontrollers
[More info](#)

Harvard_TinyML v

Close

The screenshot shows the Arduino Library Manager interface. At the top, the search bar contains the text 'tensorflow'. Below the search bar, the first result is 'Arduino_TensorFlowLite' by Pete Warden. The description for this library states: 'Allows you to run machine learning models locally on your device. This library runs TensorFlow machine learning models on microcontrollers, allowing you to build AI/ML applications powered by deep learning and neural networks. With the included examples, you can recognize speech, detect people using a camera, and recognise "magic wand" gestures using an accelerometer. The examples work best with the Arduino Nano 33 BLE Sense board, which has a microphone and accelerometer.' Below the description, there is a 'More info' link. To the right of the description, a dropdown menu shows 'Version 2.4.0-ALPHA' and an 'Install' button. Below this, there is a partial description for another library: 'Library to abstract away TensorFlow Lite for Adafruit/Arcada boards This is a library to abstract away TensorFlow Lite for Adafruit/Arcada boards' with a 'More info' link. Below that is 'EloquentTinyML' by Simone Salerno, eloquentarduino@gmail.com, described as 'An eloquent interface to Tensorflow Lite for Microcontrollers' with a 'More info' link. At the bottom, there is a partial entry for 'Harvard_TinyML'. A 'Close' button is located in the bottom right corner of the window.

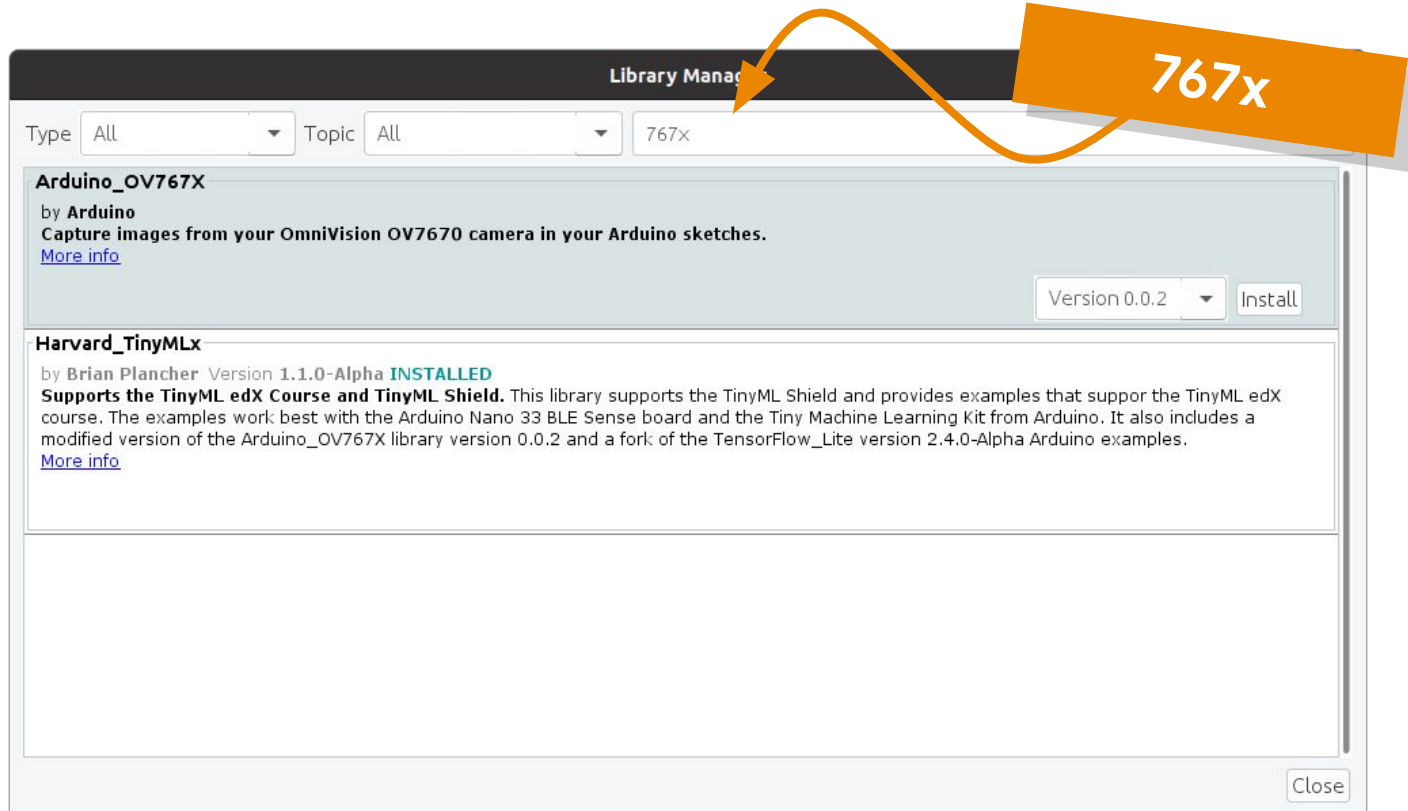
tensorflow

Arduino_TensorFlowLite

Version 2.4.0-ALPHA

Install Extras





The screenshot shows the Arduino IDE Library Manager interface. At the top, the title bar reads "Library Manager". Below it, there are search filters: "Type" set to "All", "Topic" set to "All", and a search box containing "767x". An orange arrow points from a callout box to the search box. The callout box is orange and contains the text "767x".

The search results are displayed in a list:

- Arduino_OV767X**
by **Arduino**
Capture images from your OmniVision OV7670 camera in your Arduino sketches.
[More info](#)
Version 0.0.2
- Harvard_TinyMLx**
by **Brian Plancher** Version 1.1.0-Alpha **INSTALLED**
Supports the TinyML edX Course and TinyML Shield. This library supports the TinyML Shield and provides examples that support the TinyML edX course. The examples work best with the Arduino Nano 33 BLE Sense board and the Tiny Machine Learning Kit from Arduino. It also includes a modified version of the Arduino_OV767X library version 0.0.2 and a fork of the TensorFlow_Lite version 2.4.0-Alpha Arduino examples.
[More info](#)

A "Close" button is located at the bottom right of the window.

Library Manager

Type All Topic All 767x

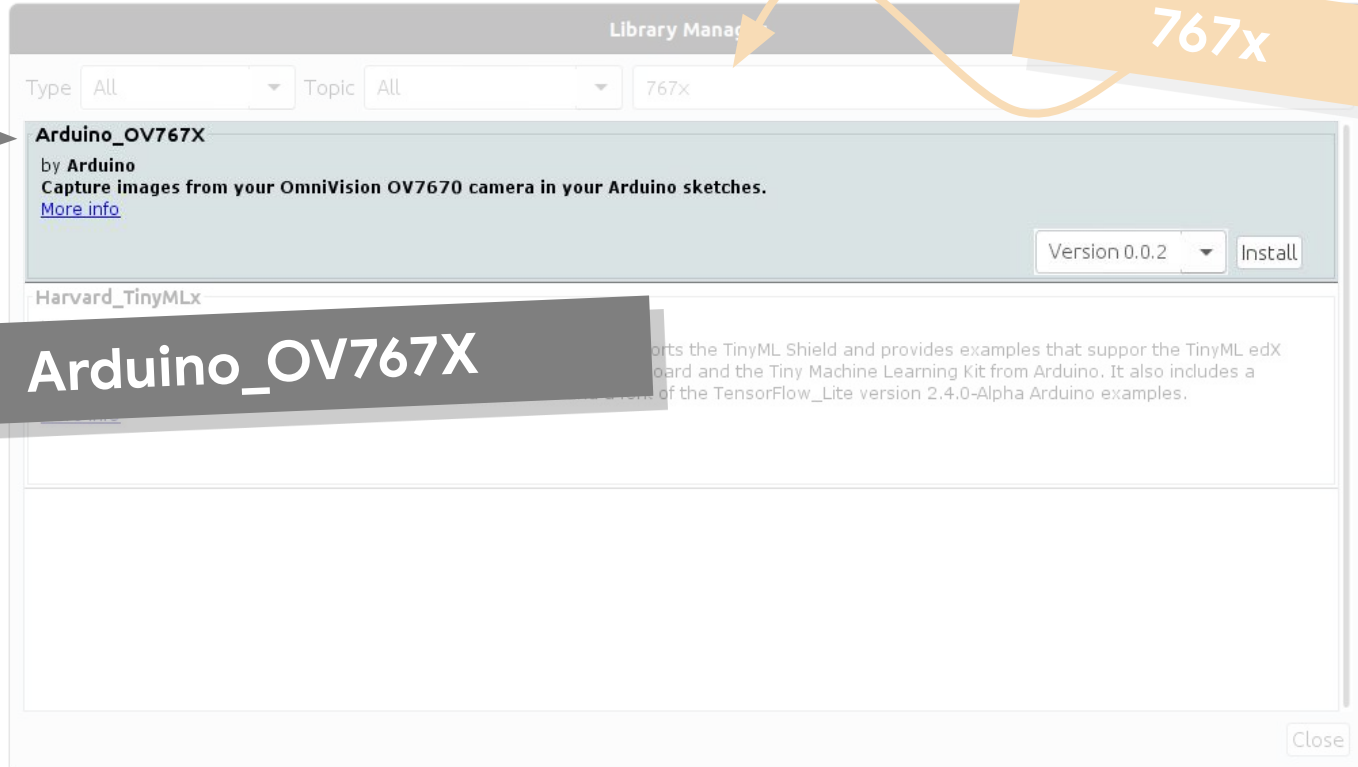
Arduino_OV767X
by **Arduino**
Capture images from your OmniVision OV7670 camera in your Arduino sketches.
[More info](#)

Version 0.0.2 Install

Harvard_TinyMLx

Supports the TinyML Shield and provides examples that support the TinyML edX board and the Tiny Machine Learning Kit from Arduino. It also includes a selection of the TensorFlow_Lite version 2.4.0-Alpha Arduino examples.

Close



767x

Arduino_OV767X

Library Manager

Type All Topic All 767x

Arduino_OV767X
by **Arduino**
Capture images from your OmniVision OV7670 camera in your Arduino sketches.
[More info](#)

Version 0.0.2 Install

Harvard_TinyMLx
by **Brian Plancher** Version 1.1.0-Alpha **INSTALLED**
Supports the **TinyML edX Course** and **TinyML Shield**. This library supports the TinyML Shield and provides examples that support the TinyML edX course. The examples work best with the Arduino Nano 33 BLE Sense board and the Tiny Machine Learning Kit from Arduino. It also includes a modified version of the Arduino_OV767X library version 0.0.2 and a fork of the TensorFlow_Lite version 2.4.0-Alpha Arduino examples.
[More info](#)

Close

Library Manager

Type All Topic All 767x

Arduino_OV767X
by **Arduino**
Capture images from your OmniVision OV7670 camera in your Arduino sketches.
[More info](#)

Version 0.0.2 Install

Harvard_TinyMLx

ports the TinyML Shield and provides examples that support the TinyML edX board and the Tiny Machine Learning Kit from Arduino. It also includes a selection of the TensorFlow_Lite version 2.4.0-Alpha Arduino examples.

Close

767x

Arduino_OV767X

Version 0.0.2

The screenshot shows the Arduino IDE Library Manager window. At the top, the title bar reads "Library Manager". Below the title bar, there are filters for "Type" (set to "All"), "Topic" (set to "All"), and a search box containing "767x".

Two library entries are visible:

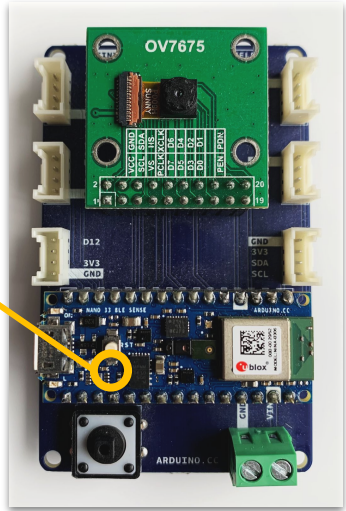
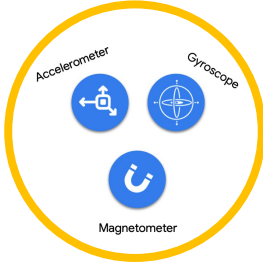
- Arduino_OV767X**
by Arduino Version 0.0.2 **INSTALLED**
Capture images from your OmniVision OV7670 camera in your Arduino sketches.
[More info](#)
- Harvard_TinyMLx**
by Brian Plancher
Supports the TinyML edX Course and TinyML Shield. This library supports the TinyML Shield and provides examples that support the TinyML edX course. The examples work best with the Arduino Nano 33 BLE Sense board and the Tiny Machine Learning Kit from Arduino. It also includes a modified version of the Arduino_OV767X library version 0.0.2 and a fork of the TensorFlow_Lite version 2.4.0-Alpha Arduino examples.
[More info](#)

At the bottom right of the Harvard_TinyMLx entry, there is a dropdown menu showing "Version 1.1.0-Alp..." and an "Install" button.

Annotations in the image:

- An orange arrow points from an orange box labeled "767x" to the search filter.
- A red arrow points from a red box labeled "Harvard_TinyMLx" to the library name.
- A yellow arrow points from a yellow box labeled "Version 1.1.0-ALPHA" to the version dropdown menu.

Install Extras



Library Manager

Type All Topic All 9DS1

Arduino_LSM9DS1
by Arduino
Allows you to read the accelerometer, magnetometer and gyroscope values from the LSM9DS1 IMU on your Arduino Nano 33 BLE Sense.
[More info](#)

Version 1.1.0 Install

Adafruit LSM9DS1 Library
LSM9DS1 9-DOF sensor board.

Melopero LSM9DS1
by Leonardo La Rocca
A driver library for the LSM9DS1 9-DOF IMU. This library allows an Arduino board reading the accelerometer, gyroscope and magnetometer data. Accelerometer / gyroscope / magnetometer data.
[More info](#)

SmartEverything LSM9DS1
by axelelectronica
Library code for LSM9DS1 iNEMO inertial module: 3D accelerometer, 3D gyroscope, 3D magnetometer The LSM9DS1 is a system-in-package

Close

9DS1

Arduino_LSM9DS1

Version 1.1.0

The screenshot shows the Arduino Library Manager interface. At the top, the title bar reads "Library Manager". Below it, there are search filters: "Type" set to "All", "Topic" set to "All", and a search box containing "ArduinoBLE". The search results list several libraries. The "ArduinoBLE" library is highlighted in a darker grey. Its details are as follows:

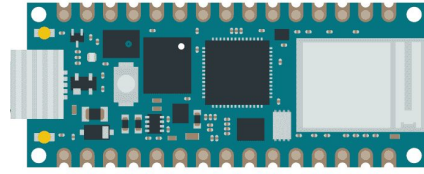
- ArduinoBLE**
- by **Arduino**
- Enables Bluetooth® Low Energy connectivity on the Arduino MKR WiFi 1010, Arduino UNO WiFi Rev.2, Arduino Nano 33 IoT, Arduino Nano 33 BLE and Nicla Sense ME.** This library supports creating a Bluetooth® Low Energy peripheral & central mode.
- [More info](#)
- Version 1.3.1 (dropdown menu)
- Install button

Other visible libraries include "Arduino_ScienceJournal" and "BeaconNano" by Petruzzella. A "Close" button is visible at the bottom right of the window.

ArduinoBLE

ArduinoBLE

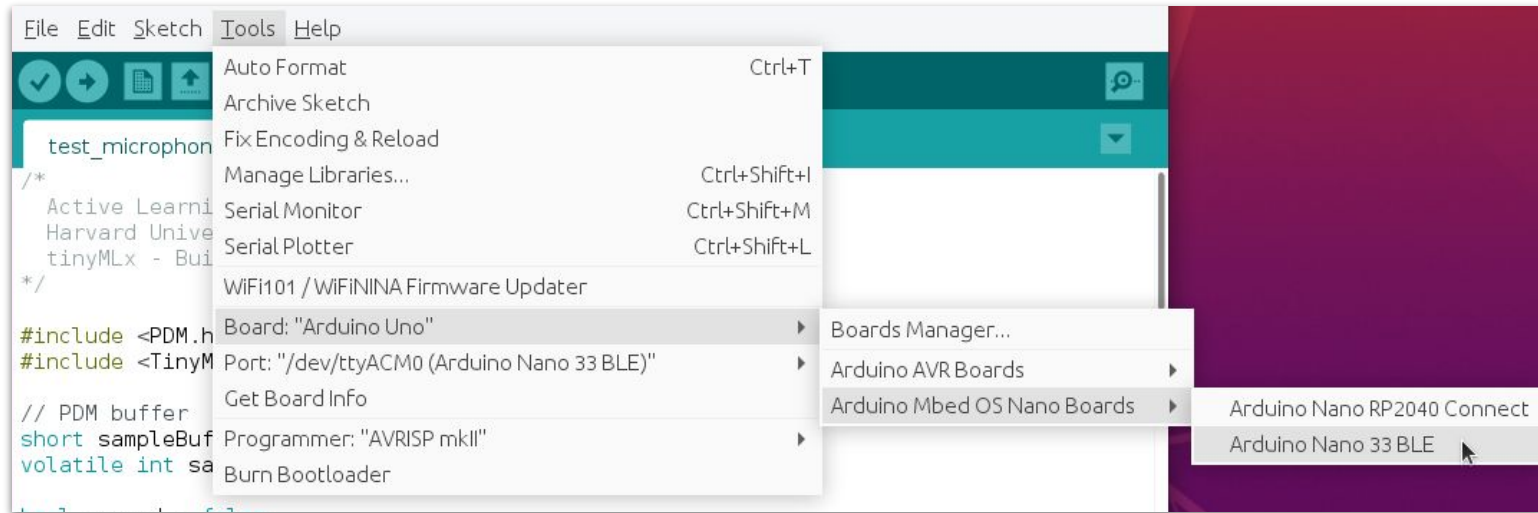
Version 1.3.1



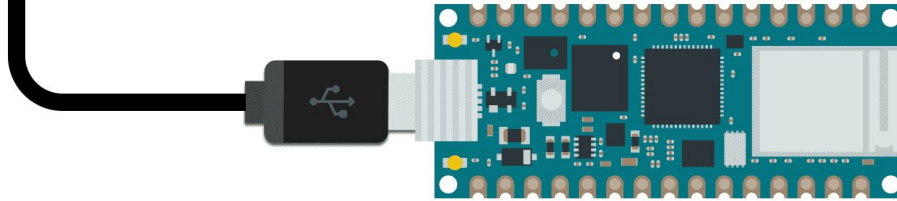
Tools → Board: “[something]”

→ Arduino Mbed OS Nano Boards

→ Arduino Nano 33 BLE



MICRO USB CABLE



PLUG IN via MicroUSB

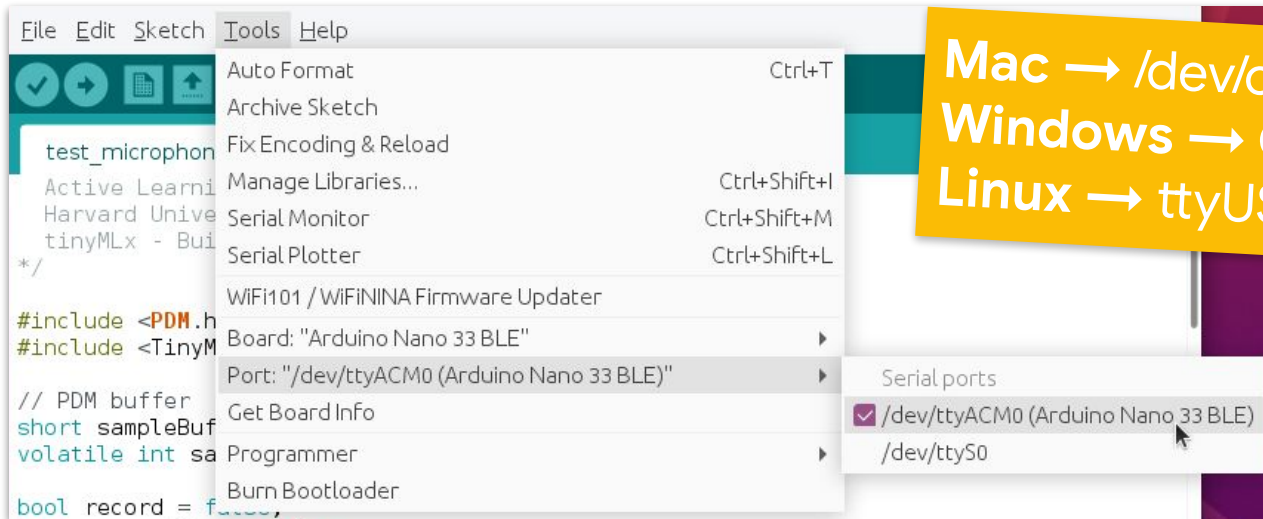
to your computer

right now!

set board and port for later

Tools → Port: “[???” (Arduino Nano 33 BLE)”

→ <???” (Arduino Nano 33 BLE)



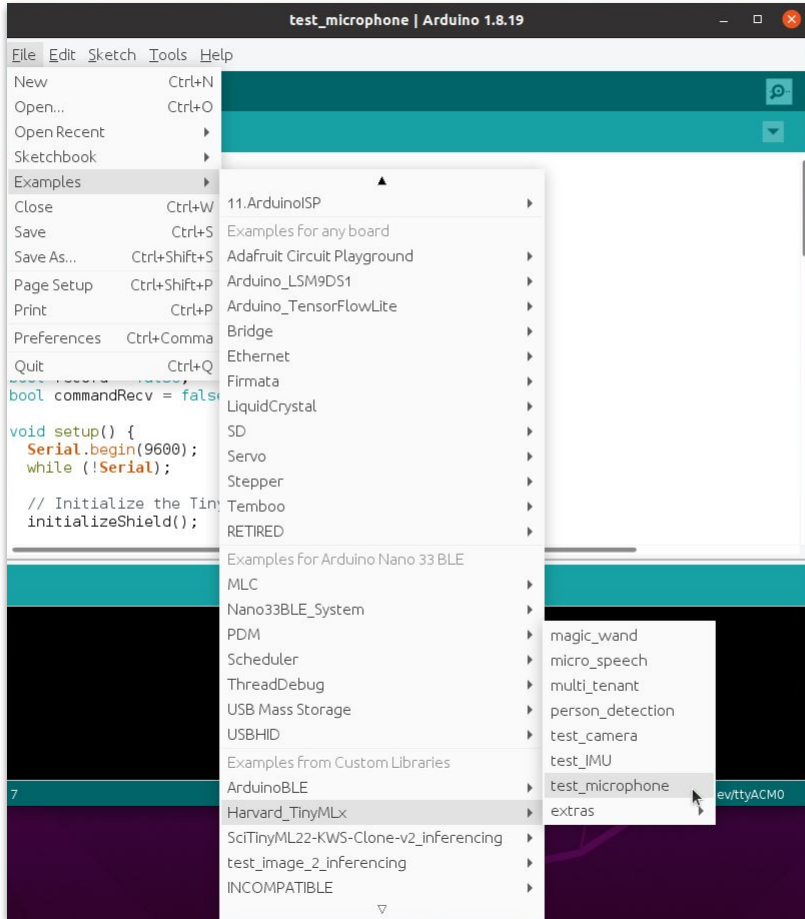
Mac → /dev/cu.usbmodem<#>
Windows → COM<#>
Linux → ttyUSB<#> or ttyACM<#>

Today's Agenda

- Review + Why Tiny?
- Hardware Basics
- Installing and Starting the Arduino IDE

Testing Your TinyML Kit

- Deploying KWS model onto Arduino
- Summary & Next Steps

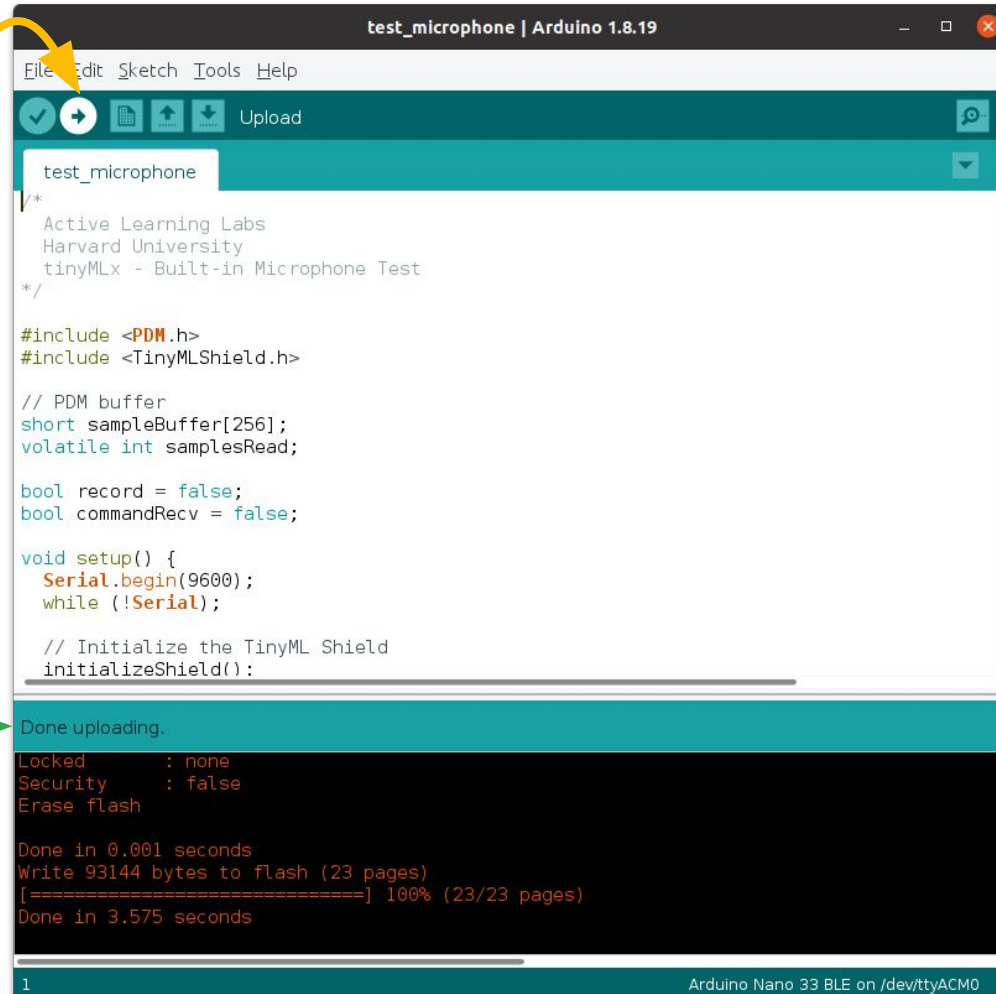


File → **Examples**

→ **Harvard_TinyMLx**

→ **test_microphone**

upload



```
test_microphone | Arduino 1.8.19
File Edit Sketch Tools Help
Upload
test_microphone
/*
 * 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();
}

Done uploading.
Locked      : none
Security    : false
Erase flash

Done in 0.001 seconds
Write 93144 bytes to flash (23 pages)
[=====] 100% (23/23 pages)
Done in 3.575 seconds

1 Arduino Nano 33 BLE on /dev/ttyACM0
```

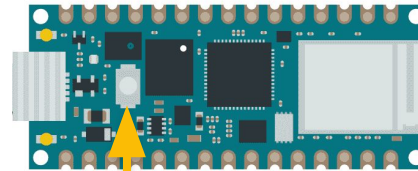
done?

An error occurred while uploading the sketch

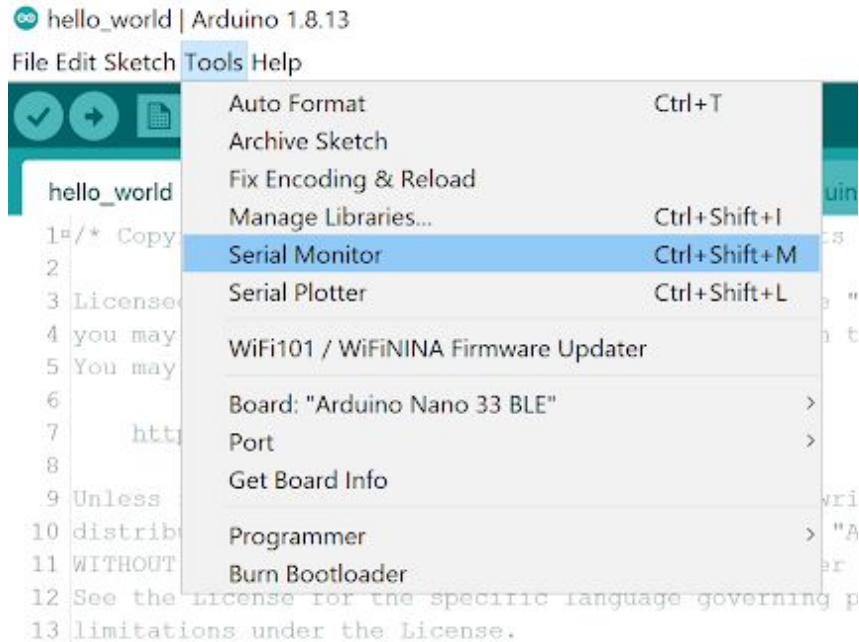
```
/home/plancher/Arduino/libraries/test_image_2_inferencing/src/edge-impulse-sdk/CMSIS/NN/Source/PoolingFunctions/arm_pool_q7_HWC
    ^__SIMD32(pCnt)++ = __QADD16(vo2, in);
/home/plancher/Arduino/libraries/test_image_2_inferencing/src/edge-impulse-sdk/tensorflow/lite/core/api/op_resolver.cpp: In fun
/home/plancher/Arduino/libraries/test_image_2_inferencing/src/edge-impulse-sdk/tensorflow/lite/core/api/op_resolver.cpp:34:20:
    builtin_code < BuiltinOperator_MIN) {
    ~~~~~^~~~~~
Sketch uses 224024 bytes (22%) of program storage space. Maximum is 983040 bytes.
Global variables use 58672 bytes (22%) of dynamic memory, leaving 203472 bytes for local variables. Maximum is 262144 bytes.
An error occurred while uploading the sketch

Device unsupported
```

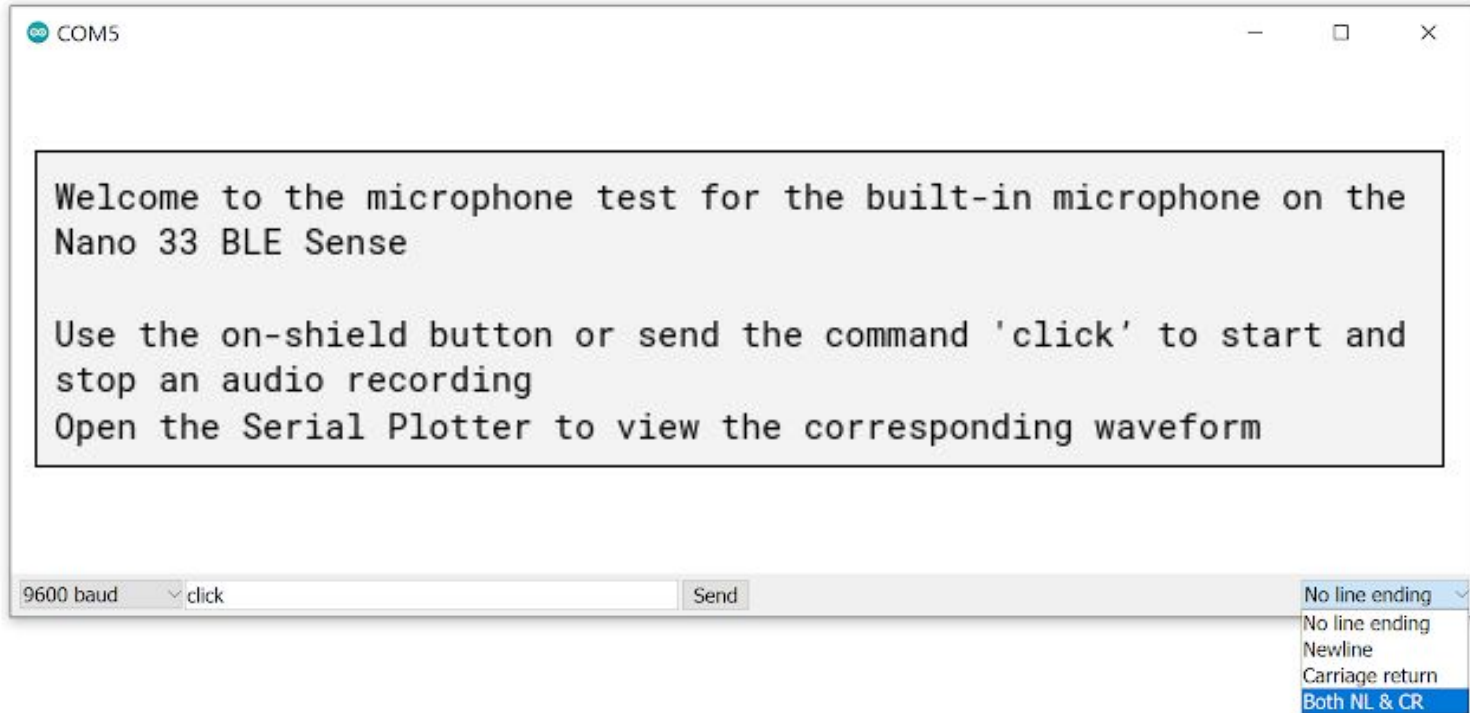
**Double Tap Reset for
Bootloader Mode!**



Tools → Serial Monitor



Tools → Serial Monitor



Tools → Serial Monitor

The image shows a screenshot of a serial monitor window titled "COM5". The window contains the following text:

```
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
```

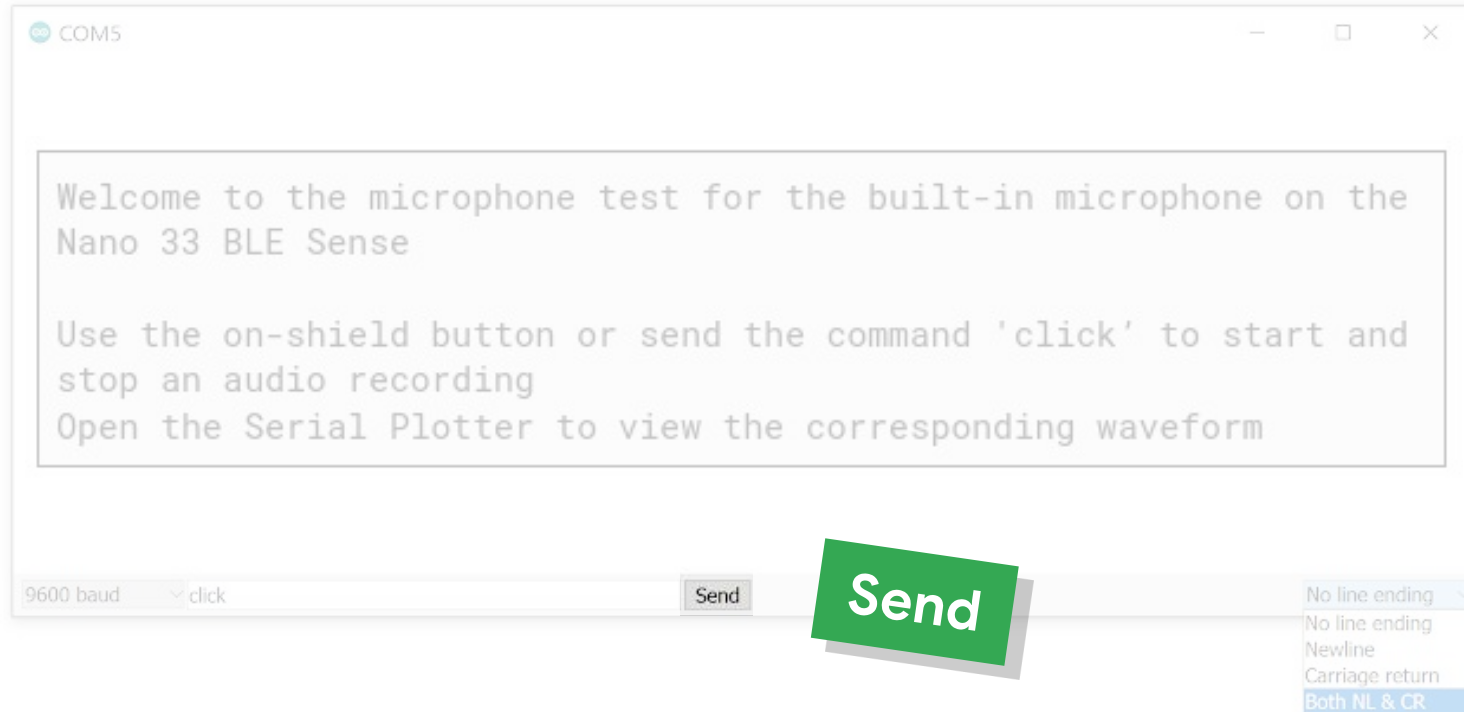
At the bottom of the window, there is an input field containing "9600 baud" and "click", and a "Send" button. A dropdown menu is open on the right side of the input field, showing the following options:

- No line ending
- No line ending
- Newline
- Carriage return
- Both NL & CR

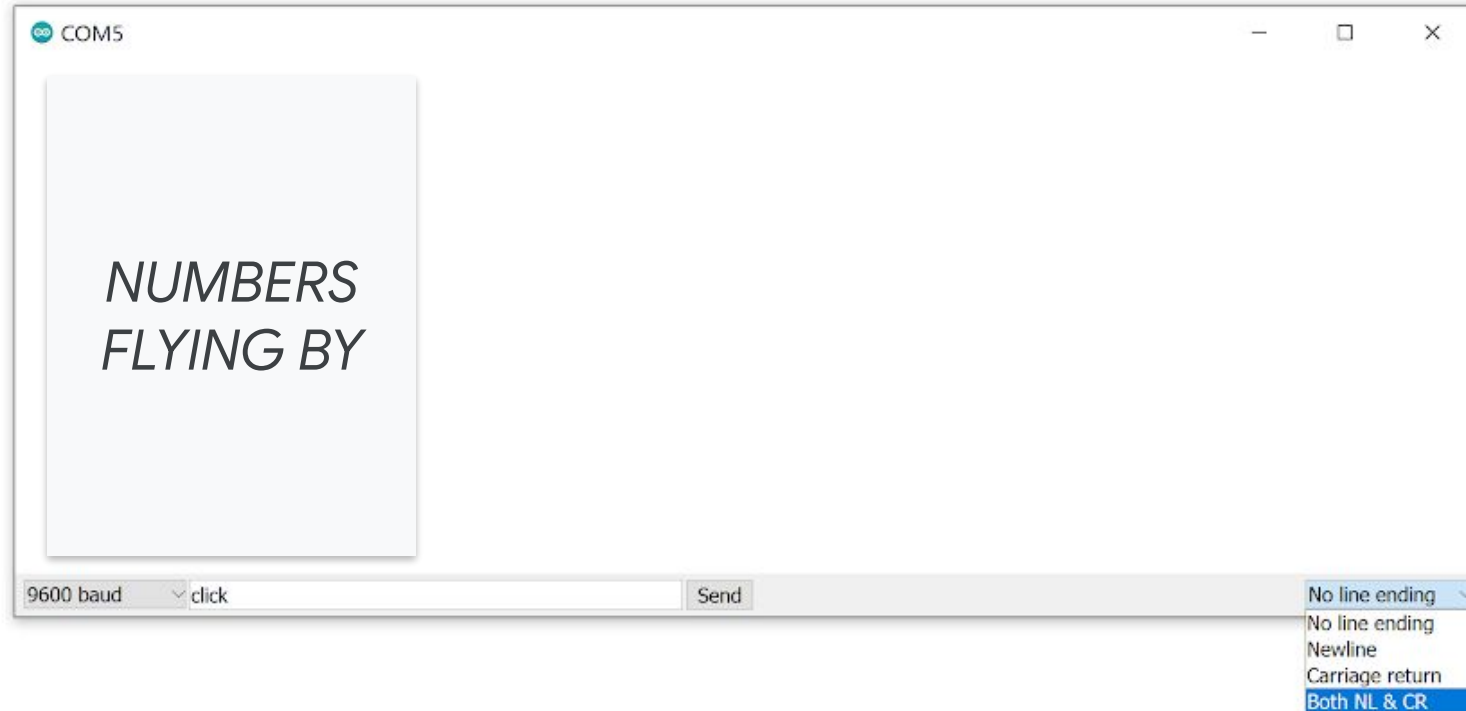
Annotations in the image highlight the following elements:

- A blue box labeled "9600 baud" with an arrow pointing to the baud rate dropdown.
- A red box labeled "click" with an arrow pointing to the command input field.
- A yellow box labeled "Both! NL & CR" with an arrow pointing to the "Both NL & CR" option in the dropdown menu.

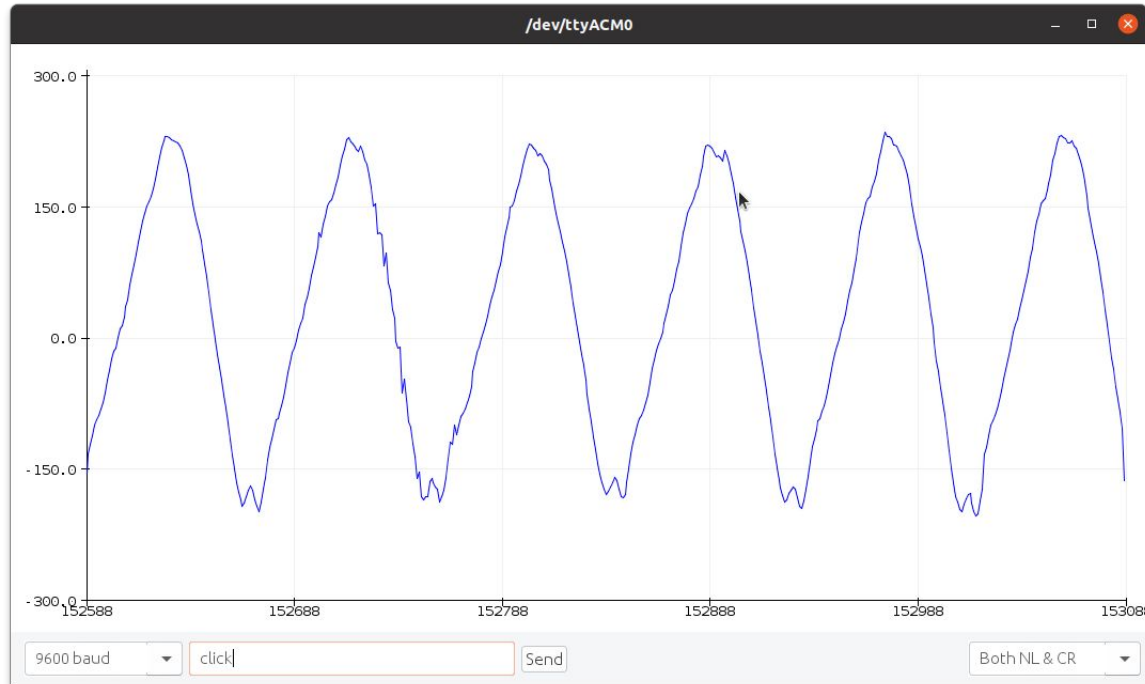
Tools → Serial Monitor



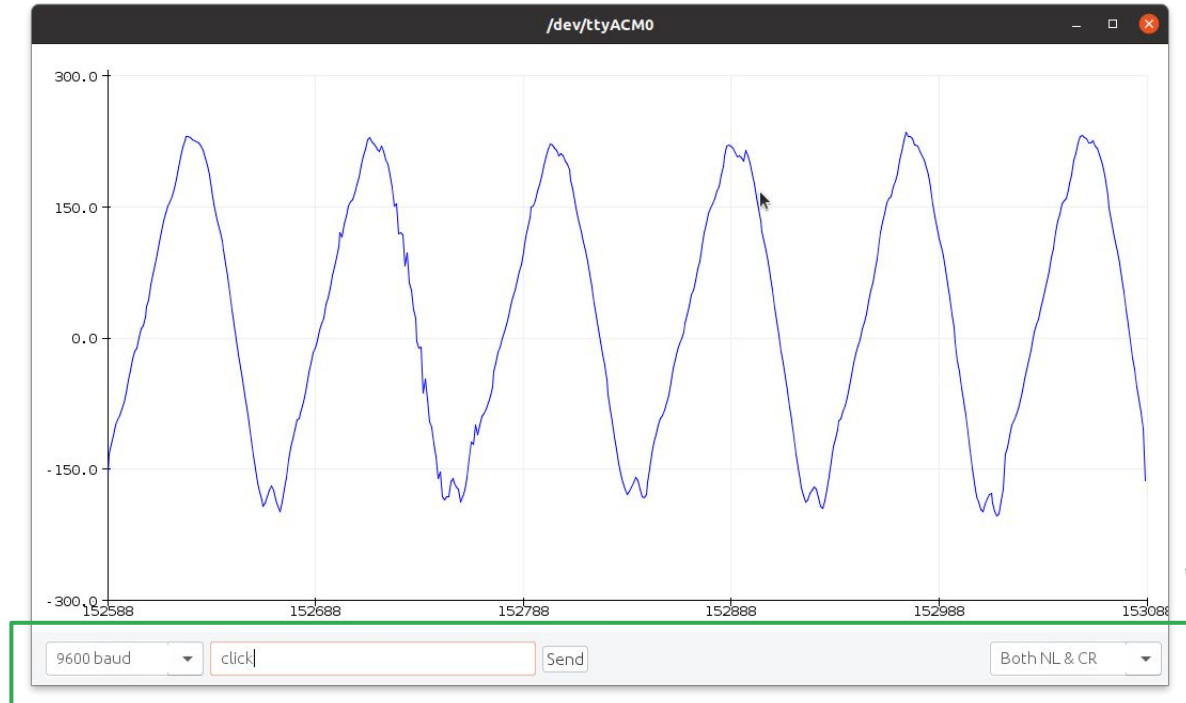
Tools → Serial Monitor



Tools → *Serial Plotter*



Tools → **Serial Plotter**



How clean of a wave can you get?

Today's Agenda

- Review + Why Tiny?
- Hardware Basics
- Installing and Starting the Arduino IDE
- Testing Your TinyML Kit

Deploying KWS model onto Arduino

- Summary & Next Steps

- Dashboard
- Devices
- Data acquisition
- Impulse design
 - Create impulse
 - Image
 - Transfer learning
- EON Tuner
- Retrain model
- Live classification
- Model testing
- Versioning
- Deployment**

GETTING STARTED

- Documentation
- Forums







DEPLOYMENT (TEST IMAGE 2)

Deploy your impulse

You can deploy your impulse to any device. This makes the model run without an internet connection, minimizes latency, and runs with minimal power consumption. [Read more.](#)




Create library

Turn your impulse into optimized source code that you can run on any device.

 C++ library	 Arduino library	 Cube.MX CMSIS-PACK
 WebAssembly	 TensorRT library	 OpenMV library

Build firmware

Get a ready-to-go binary for your development board that includes your impulse.

 Arduino Nano 33 BLE Sense	 Arduino Portenta H7	 Himax WE-I Plus
--	--	--

- Dashboard
- Devices
- Data acquisition
- Impulse design
 - Create impulse
 - Image
 - Transfer learning
- EON Tuner
- Retrain model
- Live classification
- Model testing
- Versioning
- Deployment

GETTING STARTED

- Documentation
- Forums



Select optimizations *(optional)*

Model optimizations can increase on-device performance but may reduce accuracy. Click below to analyze optimizations and see the recommended choices for your target. Or, just click Build to use the currently selected options.



Enable EON™ Compiler

Same accuracy, up to 50% less memory. Open source.



Available optimizations for Transfer learning

Quantized (int8) Currently selected	RAM USAGE	LATENCY
	66.1K	58 ms
Unoptimized (float32) Click to select	FLASH USAGE	ACCURACY
	108.1K	-
	RAM USAGE	LATENCY
	155.6K	43 ms
	FLASH USAGE	ACCURACY
	193.8K	-

Analyze optimizations

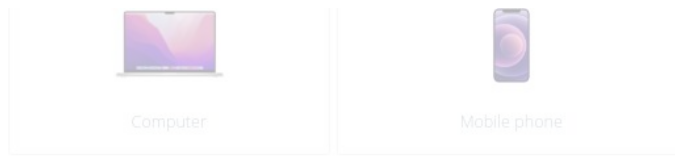
Estimate for Arduino Portenta H7 (Cortex-M7 480MHz)

Build

- Dashboard
- Devices
- Data acquisition
- Impulse design
 - Create impulse
 - Image
 - Transfer learning
- EON Tuner
- Retrain model
- Live classification
- Model testing
- Versioning
- Deployment

GETTING STARTED

- Documentation
- Forums



Select optimizations *(optional)*

Model optimizations can increase on-device performance but may reduce accuracy. Click below to analyze optimizations and see the recommended choices for your target. Or, just click Build to use the currently selected options.



Enable EON™ Compiler

Same accuracy, up to 50% less memory. Open source.



Available optimizations for Transfer learning

	RAM USAGE	LATENCY
Quantized (int8) Currently selected	66.1K	58 ms
	FLASH USAGE	ACCURACY
	108.1K	-
Unoptimized (float32) Click to select	RAM USAGE	LATENCY
	155.6K	43 ms
	FLASH USAGE	ACCURACY
	193.8K	-

Analyze optimizations

Estimate for Arduino Portenta H7 (Cortex-M7 480MHz)

Build



Quantization

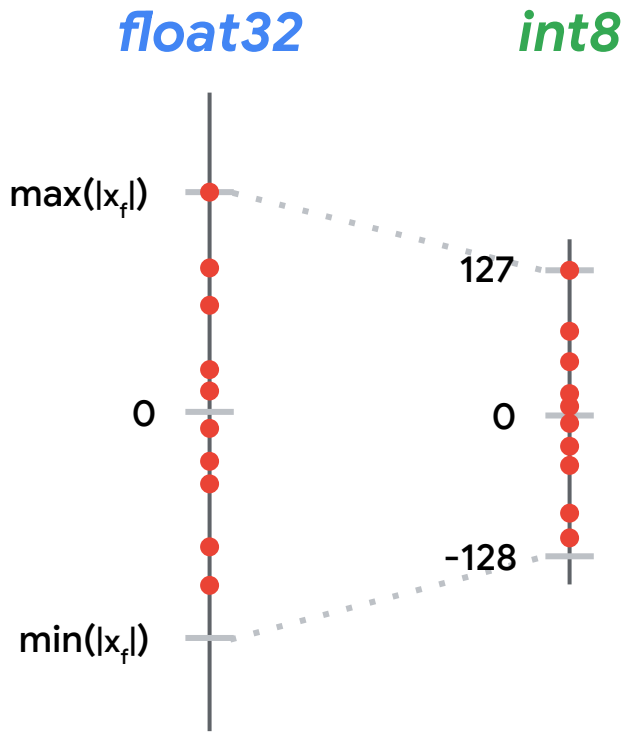
Reduces the precision of numbers used in a model which results in:

- **smaller** model size
- **faster** computation

max: 3.40282e+38
min: 1.17549e-38

Reducing the Precision

4 bytes per model parameter

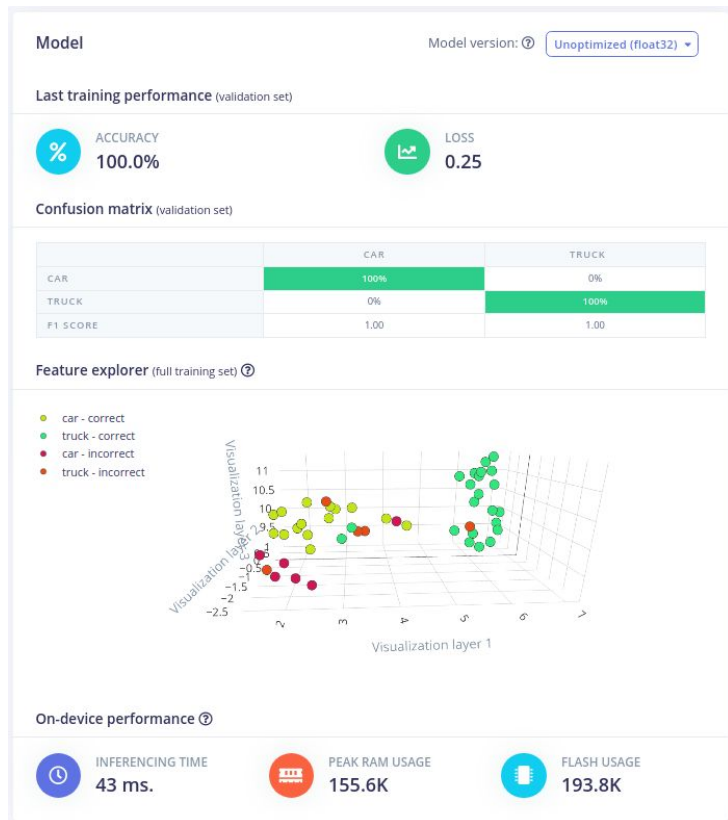


1 byte per model parameter

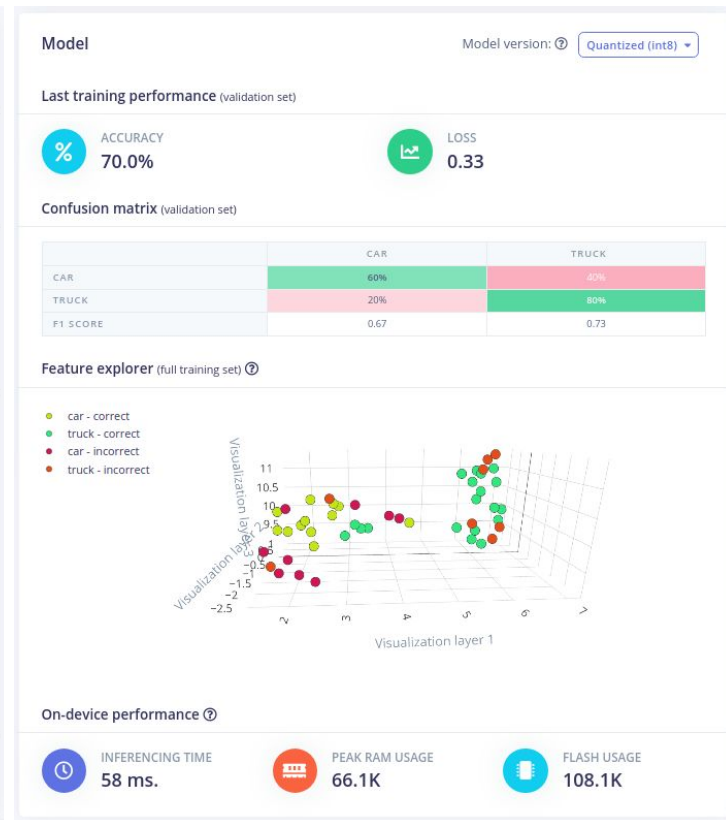
Tradeoff

	Floating-point Baseline	After Quantization	Accuracy Drop
MobileNet v1 1.0 224	71.03%	69.57%	▼1.46%
MobileNet v2 1.0 224	70.77%	70.20%	▼0.57%
Resnet v1 50	76.30%	75.95%	▼0.35%

float32



int8



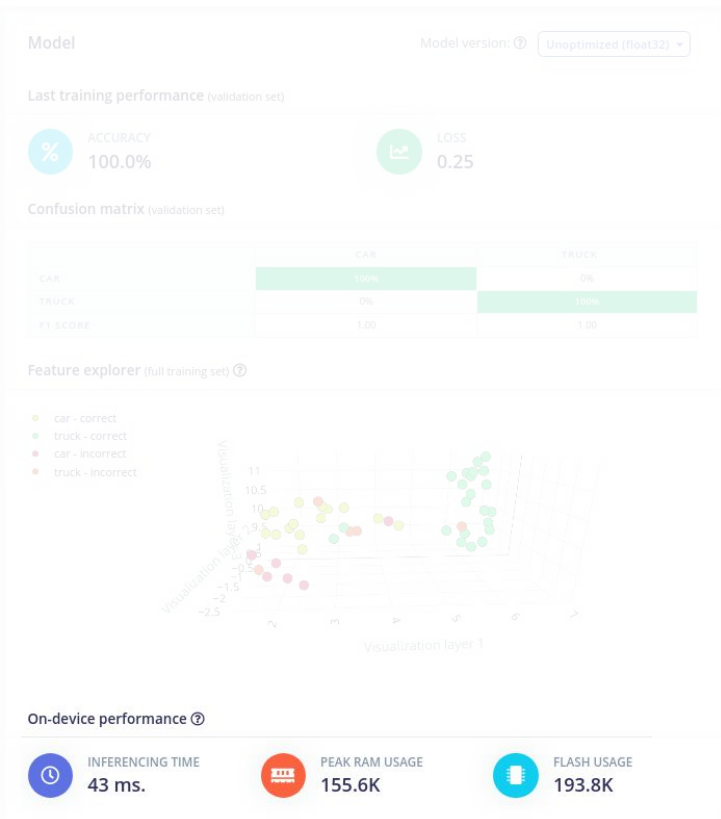
float32



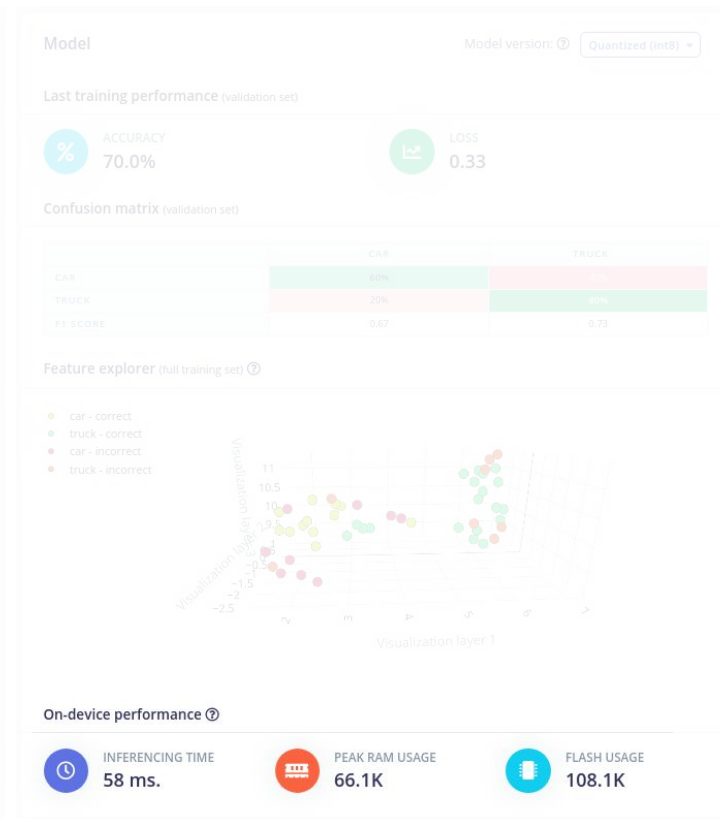
int8



float32



int8





Built Arduino library

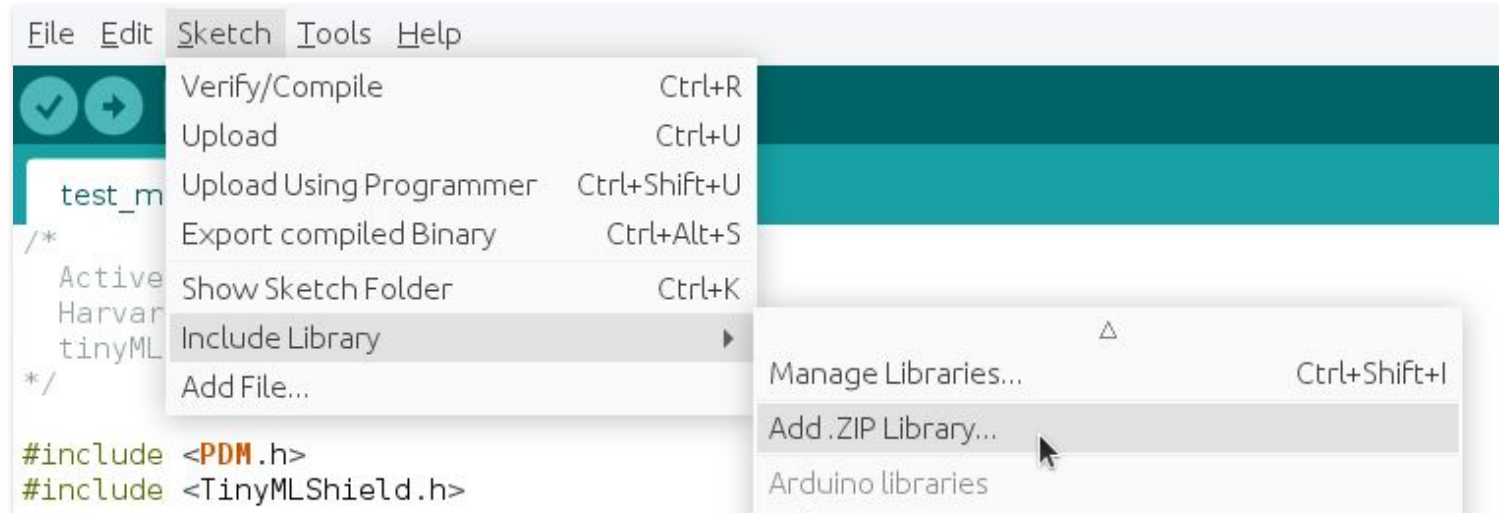
Add this library through the Arduino IDE via:

Sketch > Include Library > Add .ZIP Library...

Examples can then be found under:

File > Examples > YOUR_PROJECT_NAME

Sketch → Include Library → Add .ZIP Library





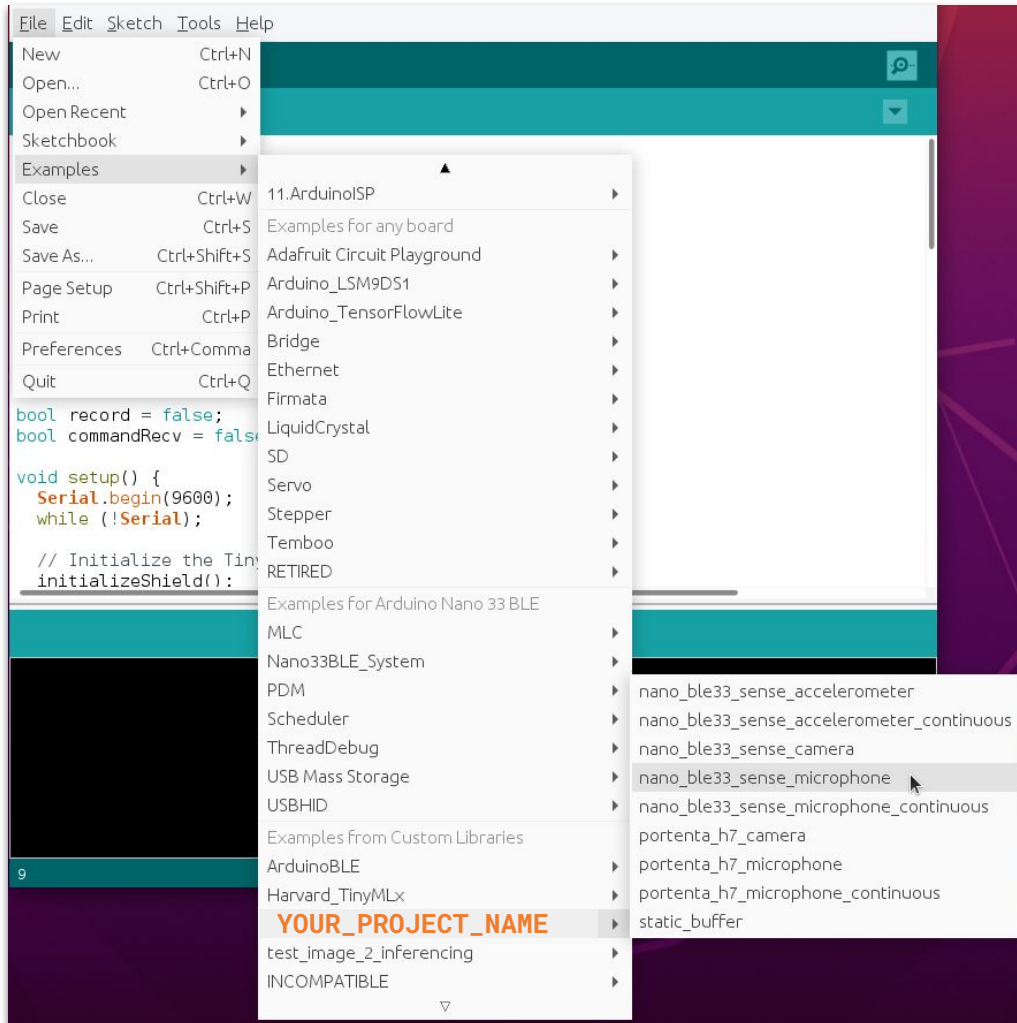
Built Arduino library

Add this library through the Arduino IDE via:

Sketch > Include Library > Add .ZIP Library...

Examples can then be found under:

File > Examples > YOUR_PROJECT_NAME

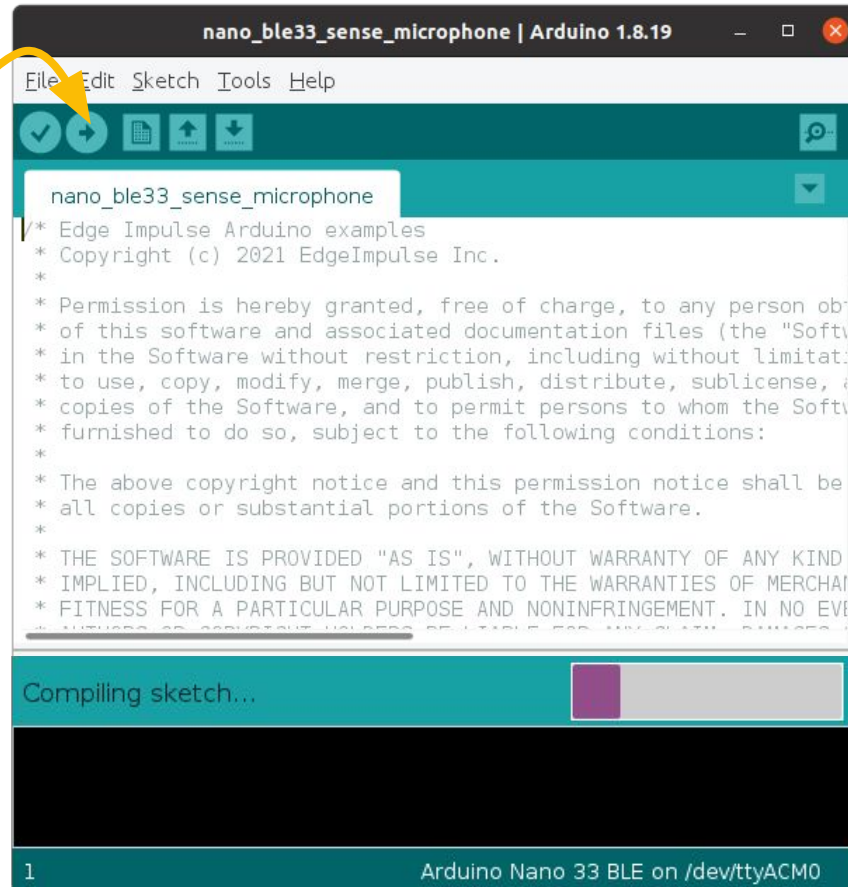


File → Examples →

YOUR_PROJECT_NAME →

nano_ble33_sense_microphone

upload



```
File Edit Sketch Tools Help
nano_ble33_sense_microphone
/* Edge Impulse Arduino examples
 * Copyright (c) 2021 EdgeImpulse Inc.
 *
 * Permission is hereby granted, free of charge, to any person obtaining
 * copies of this software and associated documentation files (the "Software"),
 * to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
 * copies of the Software, and to permit persons to whom the Software is
 * furnished to do so, subject to the following conditions:
 *
 * The above copyright notice and this permission notice shall be included in
 * all copies or substantial portions of the Software.
 *
 * THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
 * IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
 * FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
 * AUTHORS OR CONTRIBUTORS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY,
 * WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR
 * IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
 */

Compiling sketch...

1 Arduino Nano 33 BLE on /dev/ttyACM0
```

```
nano_ble33_sense_microphone | Arduino 1.8.19
File Edit Sketch Tools Help
nano_ble33_sense_microphone
/* Edge Impulse Arduino examples
 * Copyright (c) 2021 EdgeImpulse Inc.
 *
 * Permission is hereby granted, free of charge, to any person obtaining
 * copies of this software and associated documentation files (the "Software"),
 * to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
 * copies of the Software, and to permit persons to whom the Software is
 * furnished to do so, subject to the following conditions:
 *
 * The above copyright notice and this permission notice shall be included in
 * all copies or substantial portions of the Software.
 *
 * THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
 * IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
 * FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
 * AUTHORS OR CONTRIBUTORS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY,
 * WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR
 * IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
 */

Done uploading.
Locked      : none
Security    : false
Erase flash

Done in 0.000 seconds
Write 173792 bytes to flash (43 pages)
[=====] 100% (43/43 pages)
Done in 6.694 seconds

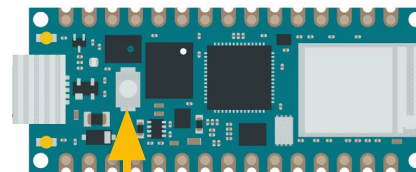
1 Arduino Nano 33 BLE on /dev/ttyACM0
```

done?

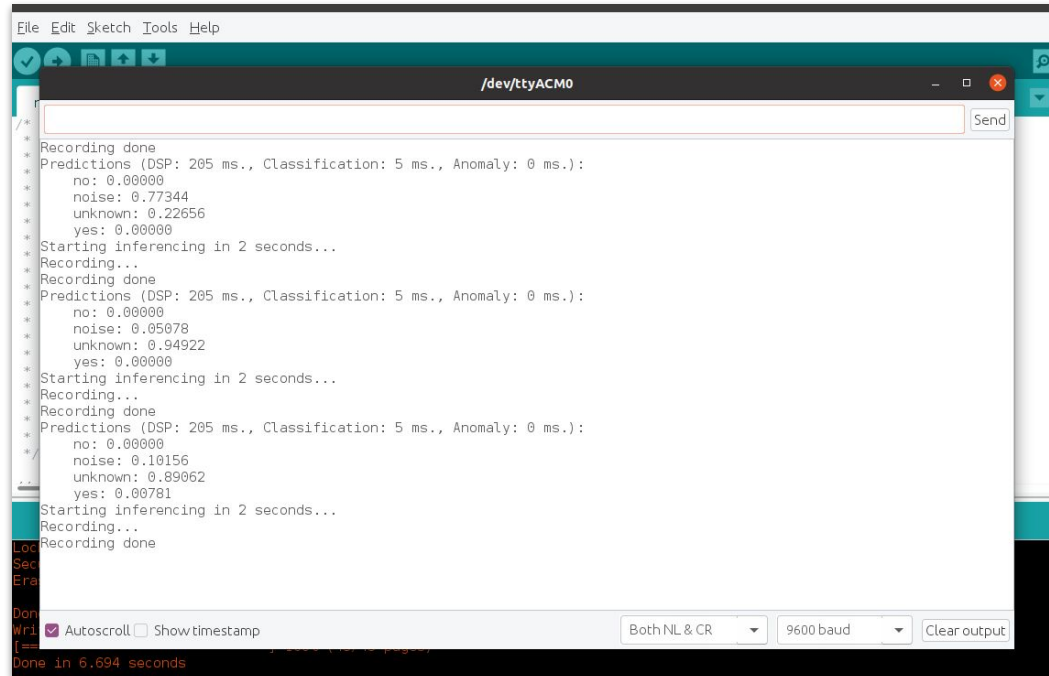
An error occurred while uploading the sketch

```
/home/plancher/Arduino/libraries/test_image_2_inferencing/src/edge-impulse-sdk/CMSIS/NN/Source/PoolingFunctions/arm_pool_q7_HW0
    ^__SIMD32(pCnt)++ = __QADD16(vo2, in);
/home/plancher/Arduino/libraries/test_image_2_inferencing/src/edge-impulse-sdk/tensorflow/lite/core/api/op_resolver.cpp: In fun
/home/plancher/Arduino/libraries/test_image_2_inferencing/src/edge-impulse-sdk/tensorflow/lite/core/api/op_resolver.cpp:34:20:
    builtin_code < BuiltinOperator_MIN) {
    ~~~~~^~~~~~
Sketch uses 224024 bytes (22%) of program storage space. Maximum is 983040 bytes.
Global variables use 58672 bytes (22%) of dynamic memory, leaving 203472 bytes for local variables. Maximum is 262144 bytes.
An error occurred while uploading the sketch
Device unsupported
```

**Double Tap Reset for
Bootloader Mode!**



Tools → Serial Monitor



```
File Edit Sketch Tools Help
/dev/ttyACM0
Recording done
Predictions (DSP: 205 ms., Classification: 5 ms., Anomaly: 0 ms.):
no: 0.00000
noise: 0.77344
unknown: 0.22656
yes: 0.00000
Starting inferencing in 2 seconds...
Recording...
Recording done
Predictions (DSP: 205 ms., Classification: 5 ms., Anomaly: 0 ms.):
no: 0.00000
noise: 0.05078
unknown: 0.94922
yes: 0.00000
Starting inferencing in 2 seconds...
Recording...
Recording done
Predictions (DSP: 205 ms., Classification: 5 ms., Anomaly: 0 ms.):
no: 0.00000
noise: 0.10156
unknown: 0.89062
yes: 0.00781
Starting inferencing in 2 seconds...
Recording...
Recording done
Done in 6.694 seconds
```

Autoscroll Show timestamp Both NL & CR 9600 baud Clear output

File Edit Sketch Tools Help

/dev/ttyACM0

Send

```
*  
* Recording done  
* Predictions (DSP: 205 ms., Classification: 5 ms., Anomaly: 0 ms.):  
*   no: 0.00000  
*   noise: 0.77344  
*   unknown: 0.22656  
*
```

Predictions (DSP: 205 ms., Classification: 5 ms., Anomaly: 0 ms.)
no: 0.00000
noise: 0.10156
unknown: 0.89062
yes: 0.00781

**Confidence that the audio
is one of the choices (0-1 scale)**

```
* Predictions (DSP: 205 ms., Classification:  
*   no: 0.00000  
*   noise: 0.10156  
*   unknown: 0.89062  
*   yes: 0.00781
```

```
Starting inferencing in 2 seconds...  
Recording...  
Recording done
```

Loc
Sec
Era

Done
wri Autoscroll Show timestamp

Both NL & CR

9600 baud

Clear output

Done in 6.694 seconds

Today's Agenda

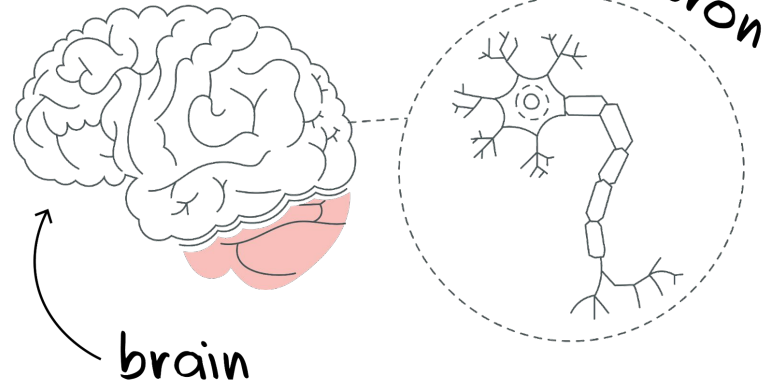
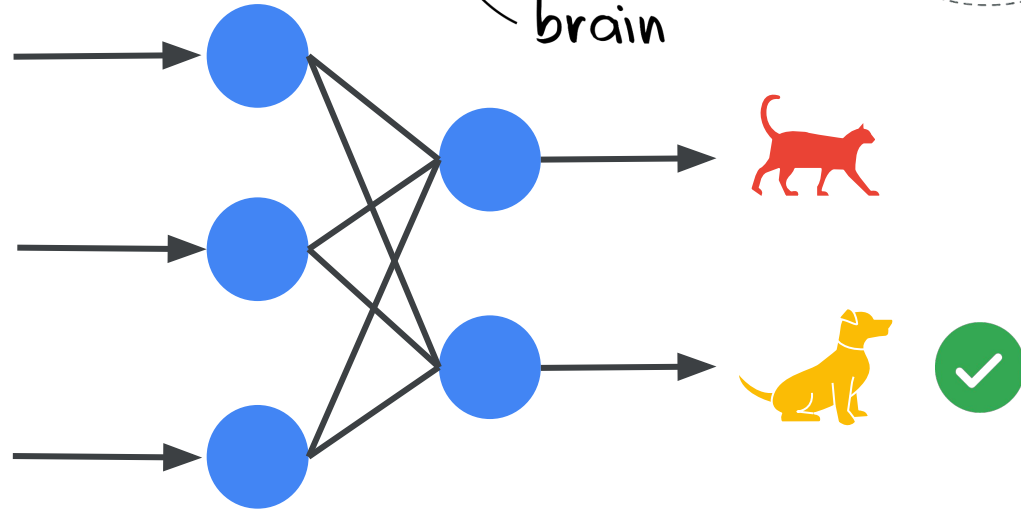
- Review + Why Tiny?
- Hardware Basics
- Installing and Starting the Arduino IDE
- Testing Your TinyML Kit
- Deploying KWS model onto Arduino

Summary & Next Steps

Machine Learning



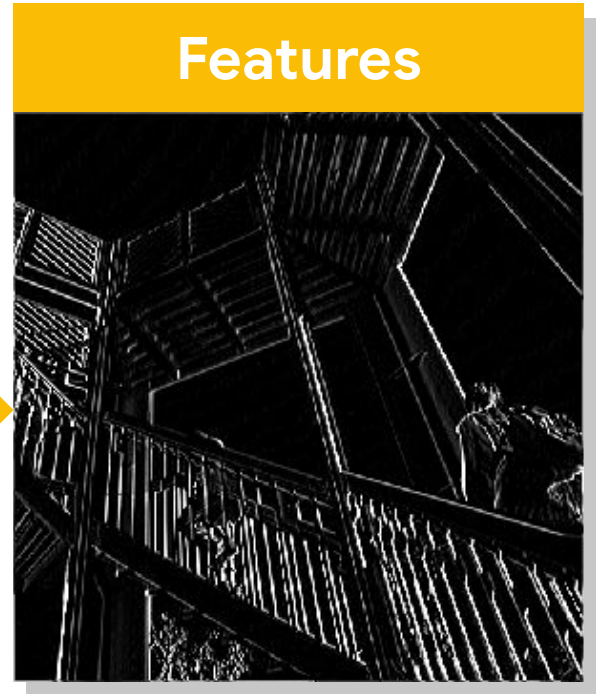
Deep Learning with **Neural Networks**



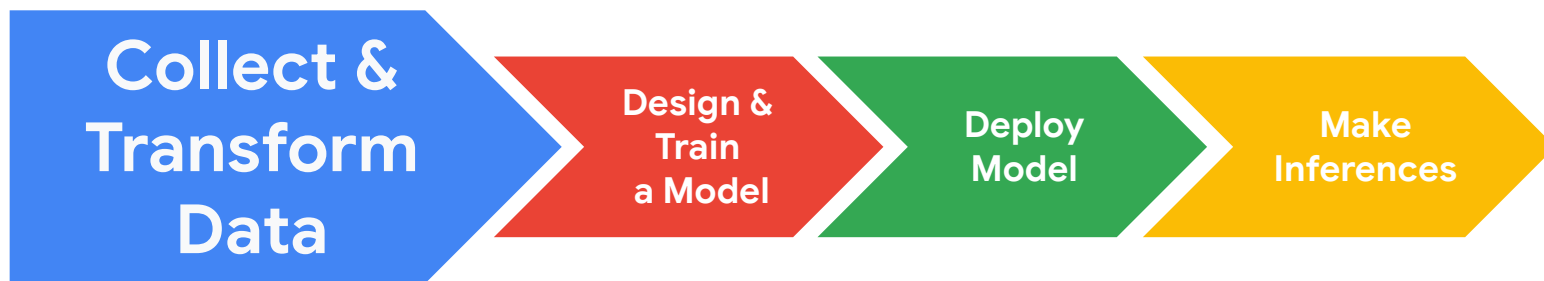
Features can be found with **Convolutions**



-1	0	1
-2	0	2
-1	0	1

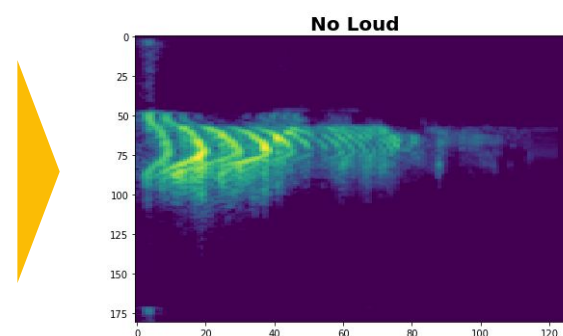
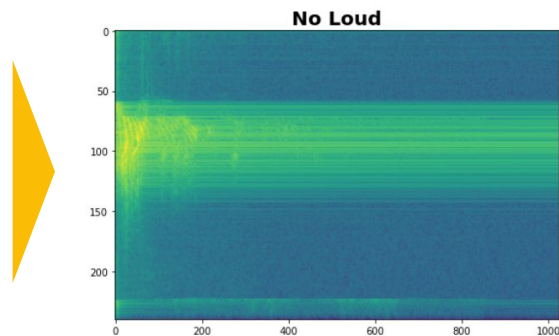
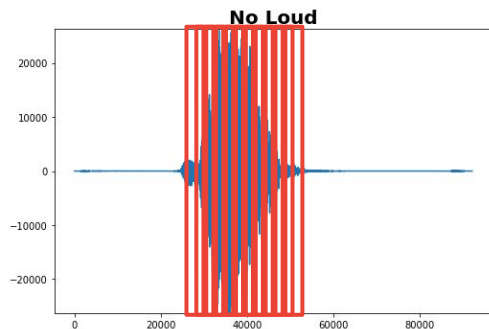
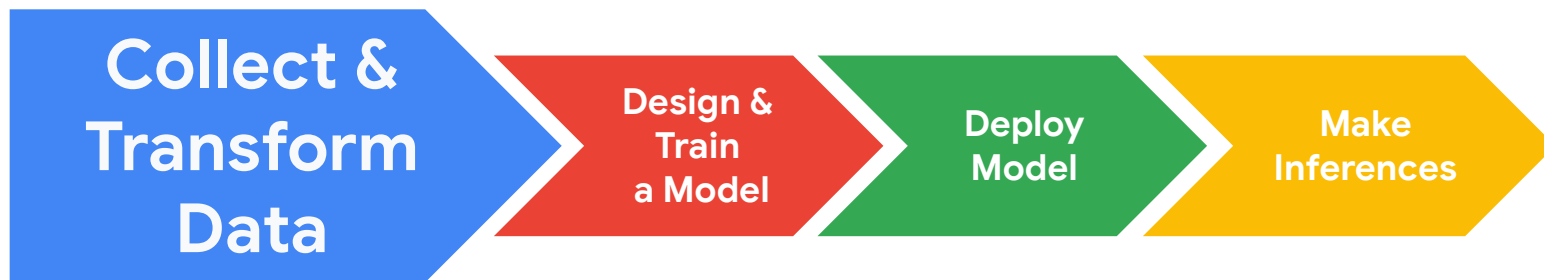


The (Tiny) Machine Learning **Workflow**

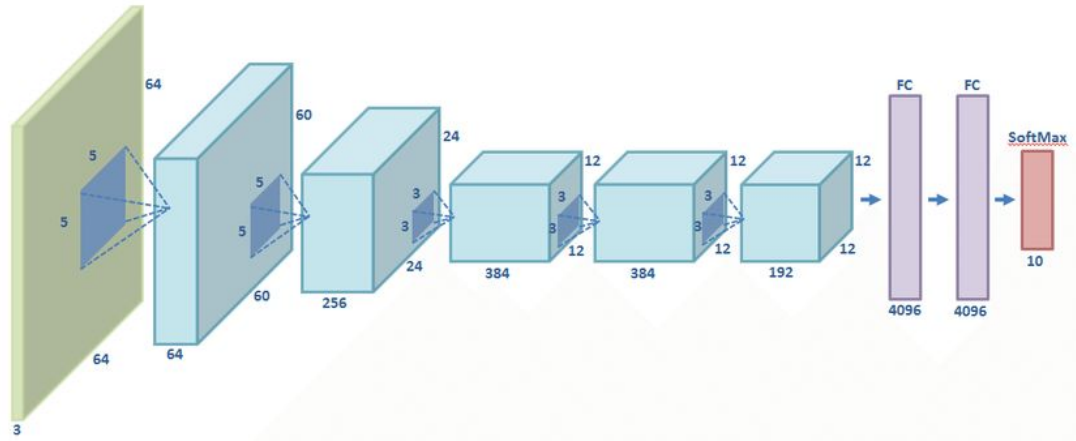
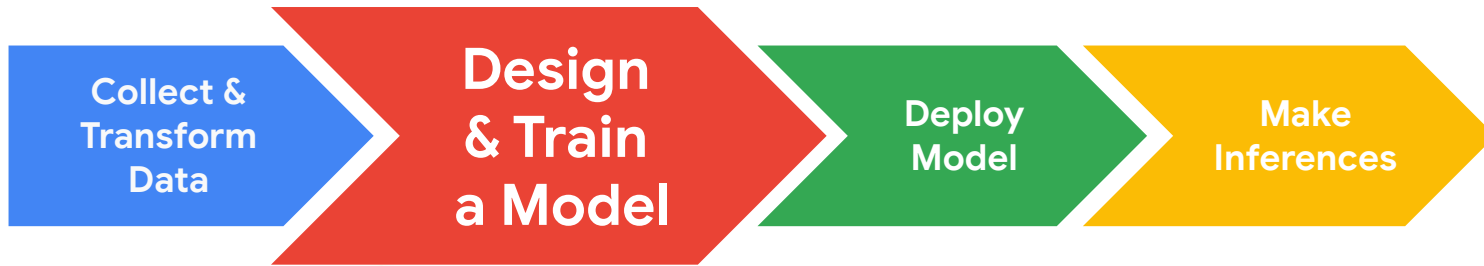


ML is everywhere → collect **GOOD**
data **RESPONSIBLY**

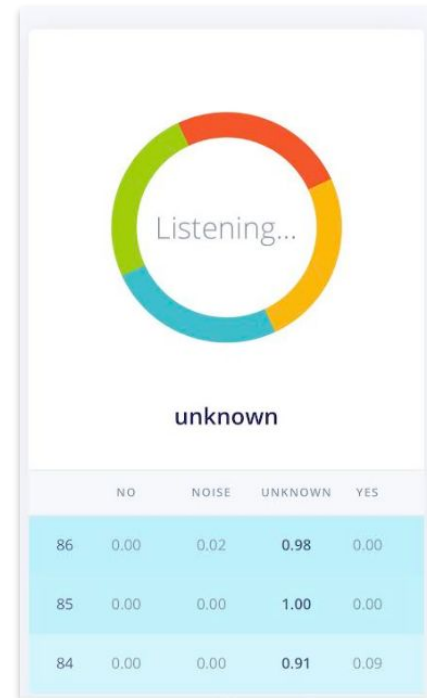
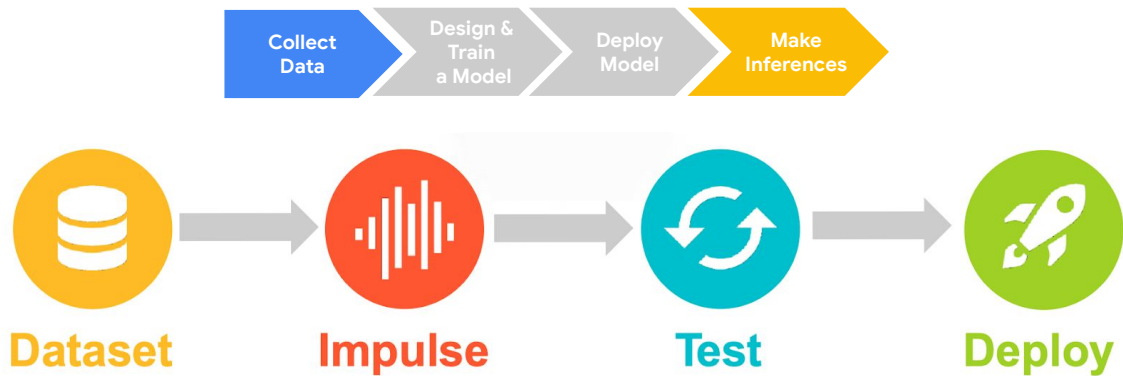
The (Tiny) Machine Learning **Workflow**



The (Tiny) Machine Learning **Workflow**

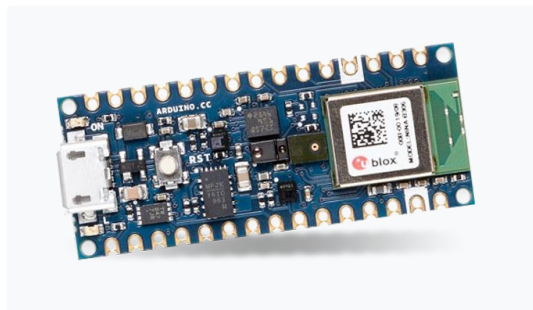
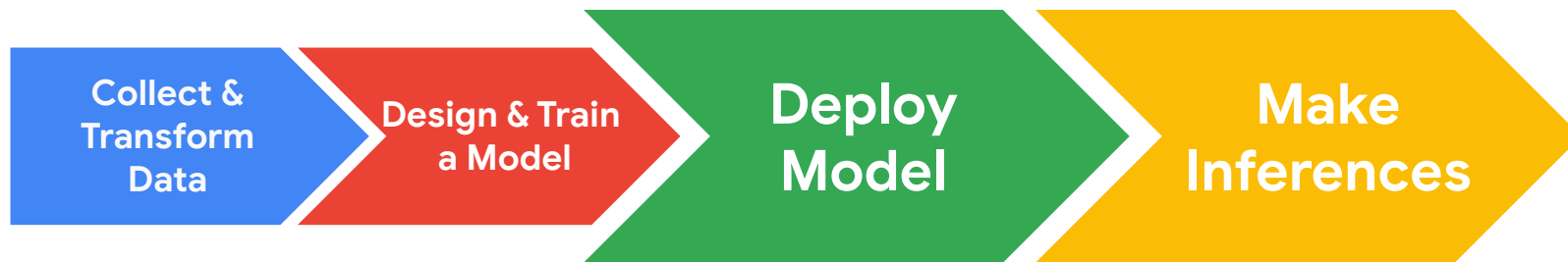


The (Tiny) Machine Learning **Workflow**



**Edge Impulse Simplifies
Training and Deployment**

The (Tiny) Machine Learning **Workflow**



Even less memory

Even less compute power

Also, only focused on *inference*

File Edit Sketch Tools Help

/dev/ttyACM0

Send

```
*  
* Recording done  
* Predictions (DSP: 205 ms., Classification: 5 ms., Anomaly: 0 ms.):  
*   no: 0.00000  
*   noise: 0.77344  
*   unknown: 0.22656  
*
```

Predictions (DSP: 205 ms., Classification: 5 ms., Anomaly: 0 ms.)
no: 0.00000
noise: 0.10156
unknown: 0.89062
yes: 0.00781

**Confidence that the audio
is one of the choices (0-1 scale)**

```
* Predictions (DSP: 205 ms., Classification:  
*   no: 0.00000  
*   noise: 0.10156  
*   unknown: 0.89062  
*   yes: 0.00781
```

```
Starting inferencing in 2 seconds...  
Recording...  
Recording done
```

Loc
Sec
Era

Done
wri Autoscroll Show timestamp

Both NL & CR

9600 baud

Clear output

Done in 6.694 seconds

Workshop **Agenda**

Day 1

Introduction to AI and (Tiny)ML

Cloud ML

Day 2

Keyword Spotting for the Navajo Language

Mobile ML

Day 3

Bringing AI/ML from the Cloud to the Edge

Embedded ML

Workshop **Agenda**

Day 1

Introduction to AI and (Tiny)ML

Cloud ML

Day 2

Keyword Spotting for the Navajo Language

Mobile ML

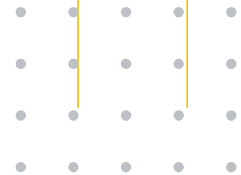
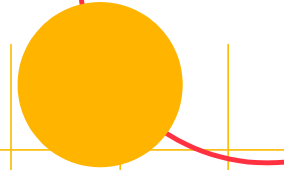
Day 3

Bringing AI/ML from the Cloud to the Edge

Embedded ML



Practice with more
machine learning!



Experiments with Google

Collections ▾

Experiments



Search

SUBMIT EXPERIMENT

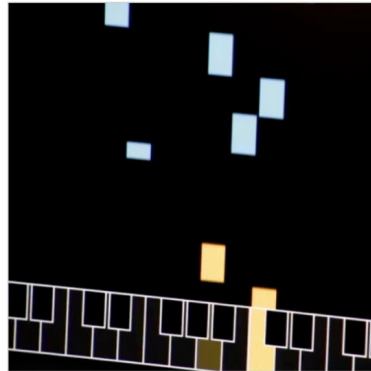
AI + MUSIC



FREDDIEMETER

by Google Research, Google Creative Lab,
YouTube Music

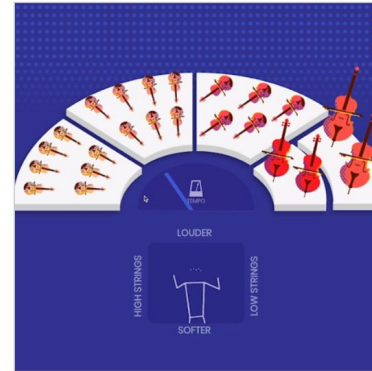
An AI-powered singing challenge that rates
how closely your singing matches the voice



AI DUET

by Yotam Mann

A piano that responds to you.



SEMI-CONDUCTOR

by Google Creative Lab

Conduct your own orchestra in the browser by
moving your arms

experiments.withgoogle.com



Teachable Machine

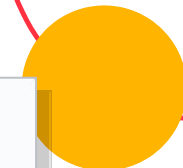
Train a computer to recognize your own images, sounds, & poses.

A fast, easy way to create machine learning models for your sites, apps, and more – no expertise or coding required.

Get Started



teachablemachine.withgoogle.com



The screenshot displays the Edge Impulse web interface. On the left is a navigation sidebar with options like Dashboard, Devices, Data acquisition, Impulse design, Retrain model, Live classification, Model testing, Versioning, and Deployment. The main content area is titled "Classification result" and is divided into several sections:

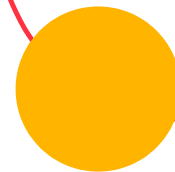
- Summary:** Shows the file name "helloworld.jan5.wav.1ncrr7qm.s17" and the expected outcome "helloworld".
- Category Count Table:**

CATEGORY	COUNT
helloworld	0
noise	0
unknown	1
uncertain	0
- Detailed result:** Includes a checkbox for "Show only unknowns" and a table with classification probabilities:

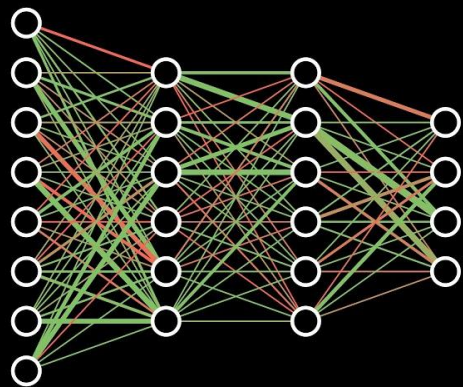
TIMESTAMP	HELLOWORLD	NOISE	UNKNOWN
0	0.36	0.01	0.62
- Raw Data:** A section titled "RAW DATA" for the file "helloworld.jan5.wav.1ncrr7qm.s17" showing an audio waveform and a play button. Below it, "Raw features" are listed as a series of numbers: "37, 34, 42, 36, 14, 1, -3, -9, -7, -10, -20, -29, -26, -21, -23, -".
- MFCC (1,649 samples):** A 3D scatter plot visualization with X, Y, and Z axes. A legend on the left identifies the data points: "helloworld" (orange), "noise" (green), "unknown" (red), and "classification 0" (blue). The plot shows a dense cluster of orange and red points.

edgeimpulse.com





Neural Networks



From the
ground up

3Blue1Brown



Introduction to TensorFlow for Artificial Intelligence, Machine Learning, and Deep Learning

★★★★★ 4.7 15,457 ratings | 👍 96%



Laurence Moroney

Enroll for Free

Starts Jun 24

Financial aid available

coursera

Browse > Data Science > Machine Learning

Offered By



Introduction to Embedded Machine Learning

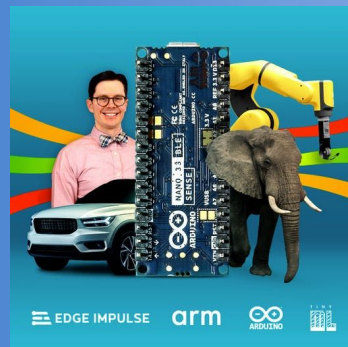
★★★★★ 4.8 335 ratings • 83 reviews



Shawn Hymel [+1 more instructor](#)

Enroll for Free
Starts Jun 16

Financial aid available



18,297 already enrolled

coursera



The Future of ML is Tiny and Bright



Professional Certificate in
Tiny Machine Learning (TinyML)

I'm interested ✓

Courses in this program



HarvardX's Tiny Machine Learning (TinyML) Professional Certificate



Fundamentals of TinyML

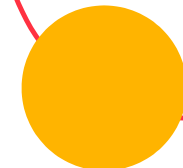


Applications of TinyML



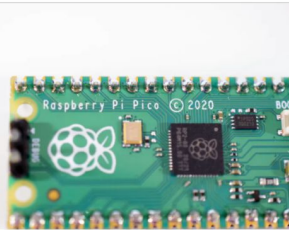
Deploying TinyML

edx.org



Explore projects

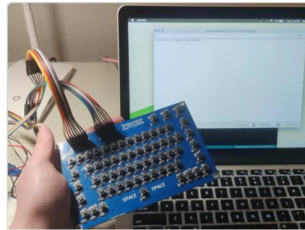
Trending ▾ All difficulties ▾ Featured ▾ Any type ▾



Getting Started with the Raspberry Pi Pico

Arduino "having11" Guy

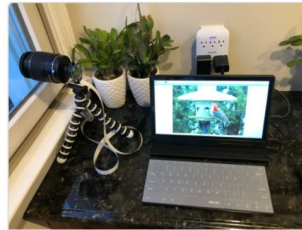
👍 15 👁 2.5K



64-Key Prototyping Keyboard Matrix for Arduino

Cameron Coward

👍 19 👁 6.8K



ML-Based Bird and Squirrel Detector (Raspberry Pi and AWS)

Mike Sadowski

👍 31 👁 5.3K



Self-Playing Melodica 🎹 🤖

touchmysound

👍 30 👁 3.9K



hackster.io



Our website!

tinyMLedu.org/EASI-22

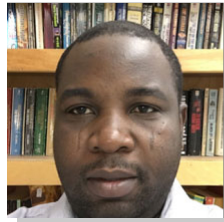
home base for **all information!**

survey!

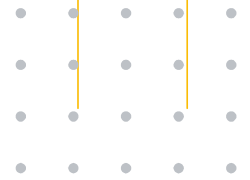
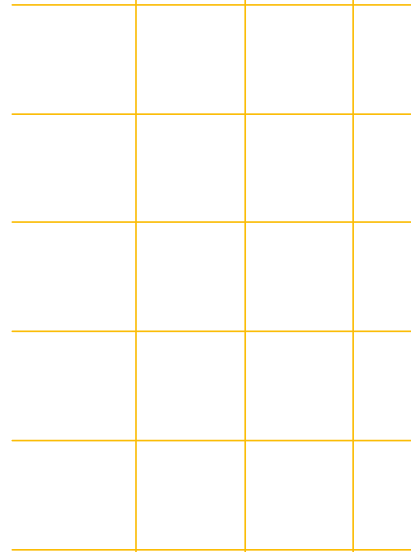
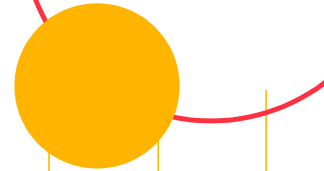
bit.ly/EAS122-Post

help us make this workshop **better!**

Our team!



with help from **many more**





hágoónee' 🖐️

thank you and **keep exploring with ML**