

Laboratorio #5: STM32/Arduino: GPIO, Giroscopio, comunicaciones, TinyML

Adrián Avilés Flores, B80835 y Steven Mora Barboza, B95109

I. INTRODUCCIÓN

En este laboratorio se ha desarrollado un reconocedor de actividad humana utilizando un kit Arduino Nano 33 BLE y TensorFlow Lite. El objetivo era capturar datos del giroscopio y utilizarlos para etiquetar tres tipos de movimientos diferentes (estacionario, vueltas y arriba-abajo).

Para lograr esto, se ha creado un programa para el microcontrolador que captura la información del giroscopio y la envía a la computadora por el puerto USB. Luego, se ha creado un script de Python que recibe esta información y la guarda en un archivo CSV etiquetada con el tipo de movimiento correspondiente.

Después, se ha utilizado TensorFlow Lite para crear un modelo de red neuronal de dos capas con varias neuronas y tres salidas (una por cada tipo de movimiento) usando ReLU como función de activación y softmax para la capa de salida. El modelo se ha entrenado con el 60 % de los datos, validado con el 20 % y evaluado con el 20 % restante.

Finalmente, se ha creado un programa que utiliza el modelo construido para detectar el tipo de movimiento realizado y lo registra en la PC.

Como conclusiones importantes, podemos destacar que este laboratorio muestra cómo se puede utilizar TensorFlow Lite para crear un modelo de red neuronal capaz de reconocer diferentes tipos de movimientos humanos. Además, se demuestra cómo la combinación de hardware y software puede ser utilizada para desarrollar soluciones útiles y prácticas en el campo de la actividad humana.

II. NOTA TEÓRICA

II-A. Arduino nano 33 BLE

El Arduino Nano 33 BLE es una placa de desarrollo de la familia de microcontroladores Arduino. Cuenta con un microcontrolador ARM Cortex-M4F a 64 MHz, una memoria flash de 256 KB y una memoria SRAM de 32 KB. La placa cuenta con 14 entradas/salidas digitales y 8 entradas analógicas, y es compatible con interfaces de comunicación como UART, SPI, I2C y USB.

Una de las características más destacables del Arduino Nano 33 BLE es su capacidad de conexión inalámbrica a través de Bluetooth Low Energy (BLE) 5.0, lo que permite la comunicación con otros dispositivos BLE como teléfonos móviles o tabletas. Además, cuenta con un giroscopio y acelerómetro de 3 ejes incorporados (LSM6DS3), y un magnetómetro de 3 ejes en un solo paquete. Esto permite medir la orientación, la velocidad angular, la aceleración y la dirección del campo magnético, lo que es útil para proyectos que involucren la

detección de movimiento, incluyendo aplicaciones de control de movimiento y navegación.

El LSM9DS1 se comunica con el microcontrolador a través de un bus de comunicación I2C, lo que simplifica su conexión y configuración. Además, cuenta con un sistema integrado de filtrado de datos para reducir el ruido y mejorar la precisión de las mediciones.

El sensor tiene una resolución de 16 bits y una tasa de muestreo de hasta 952 Hz para el acelerómetro y de hasta 8.33 kHz para el giroscopio.

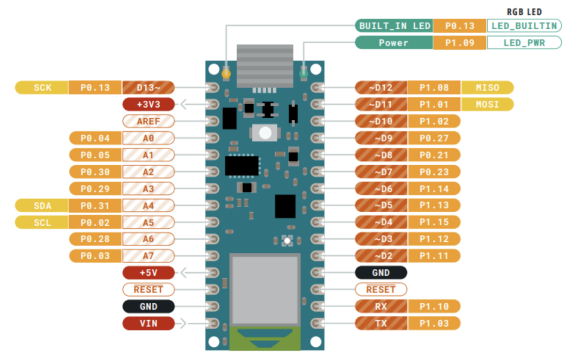


Figura 1. Diagrama de pines Arduino nano 33 BLE [1].

A continuación el diagrama de bloques del MCU:

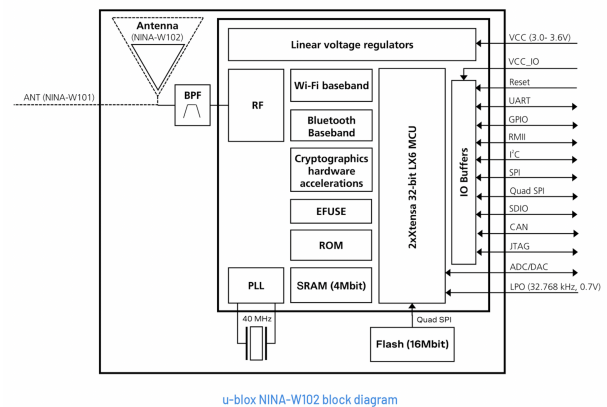


Figura 2. Diagrama de bloques del Arduino nano 33 BLE [1].

Entre los componentes a usar según el diagrama de bloques, se utilizará el conector USB para programar y alimentar el Arduino, regulador de tensión, que convierte la alimentación externa de hasta 3.3 V.

| | | |
|-------------|-------------------------|------------------------|
| Power | I/O Voltage | 3.3V |
| | Input voltage (nominal) | 5-18V |
| | DC Current per I/O Pin | 10 mA |
| Clock speed | Processor | nRF52840 64MHz |
| Memory | nRF52840 | 256 KB SRAM, 1MB flash |
| Dimensions | Weight | 5gr |
| | Width | 18 mm |
| | Length | 45 mm |

Figura 3. Especificaciones eléctricas del Arduino nano 33 BLE [1].

Además, es importante tener en cuenta las especificaciones eléctricas mostradas en la Figura 3, donde se puede observar que la tensión máxima de operación es de 3.3 V y la corriente máxima por pin input/output es de 10 mA, esto se tomará en cuenta con la elaboración del circuito según los valores de resistencias y fuentes. [2]

¿Por qué utilizar microcontroladores para aprendizaje automático e inteligencia artificial? Los microcontroladores son una excelente opción para el aprendizaje automático e inteligencia artificial debido a su bajo consumo de energía, lo que los hace eficientes en comparación con las computadoras y GPU de alta potencia que consumen mucha energía. Esto es especialmente importante en aplicaciones que dependen de baterías o recolección de energía, como en dispositivos IoT y wearables. Además, los microcontroladores son más asequibles que las estaciones de trabajo de aprendizaje automático de alta gama, lo que los hace más accesibles para los estudiantes y desarrolladores que quieren experimentar con el aprendizaje automático. La flexibilidad también es un gran beneficio, ya que los microcontroladores están en todas partes, por lo que hay muchas posibilidades de agregar inteligencia artificial a diversos dispositivos sin depender de la conectividad de red. Por último, la privacidad es una preocupación importante en el mundo actual, y los microcontroladores ofrecen una ventaja en este sentido, ya que los datos no tienen que salir del dispositivo, lo que reduce la exposición a posibles brechas de seguridad y pérdida de datos. [3]

¿Qué es la inteligencia artificial (Artificial Intelligence, AI)? En términos simples, inteligencia artificial (IA) se refiere a sistemas o máquinas que imitan la inteligencia humana para realizar tareas y pueden mejorar iterativamente a partir de la información que recopilan.[4]

¿Qué es una red neuronal? Una red neuronal es un método de la inteligencia artificial que enseña a las computadoras a procesar datos de una manera que está inspirada en la forma en que lo hace el cerebro humano. Se trata de un tipo de proceso de machine learning llamado aprendizaje profundo, que utiliza los nodos o las neuronas interconectados en una estructura de capas que se parece al cerebro humano. Crea un sistema adaptable que las computadoras utilizan para aprender de sus errores y mejorar continuamente. De esta forma, las redes neuronales artificiales intentan resolver problemas complicados, como la realización de resúmenes de documentos o el reconocimiento de rostros, con mayor precisión.[5]

II-B. Diseño del circuito

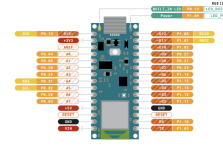


Figura 4. Esquemático del circuito completo.

En este caso, se puede apreciar solamente el Arduino nano 33 BLE, esto debido que para este laboratorio, lo único que se utiliza es este dispositivo, debido a que el sensor de movimiento lo tiene integrado en la misma placa.

A continuación la lista de componentes utilizados con sus respectivos precios en dólares:

| Componente | Cantidad | Precio \$ |
|---------------------|----------|-----------|
| Arduino nano 33 BLE | 1 | 35 |

TABLA I
PRECIO DE LOS COMPONENTES

Como se puede esperar, en este caso el único componente utilizado es el arduino.

III. ANÁLISIS DE RESULTADOS

El presente laboratorio se diferencia de los pasados en el hecho de que ahora se ocupa de 2 programas independientes para cumplir con los objetivos de la práctica. El primer script que se debe cargar en el Arduino NANO 33 BLE corresponde a uno que sea capaz de registrar los datos del giroscopio integrado y los envíe por el puerto serial para que sean procesados por el archivo datasaver.py y los almacene en archivos .csv, el flujo de este primer programa se presenta mediante la Figura 5 y fue implementado gracias al ejemplo mostrado para trabajar con TensorFlowLite en Arduino [6].

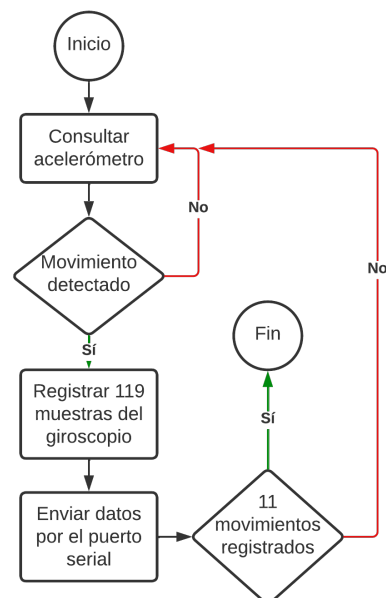


Figura 5. Funcionamiento de IMU_Capture.ino

Este programa fue correctamente ejecutado para 3 tipos de movimientos: Estacionario, Vueltas y Arriba-Abajo, un archivo .csv fue creado para cada uno de estos movimientos con un total de 1309 muestras, esto debido a que se procesaron 119 muestras en cada una de las 11 iteraciones para cada movimiento. El registro de estos datos fue exitoso y sirvió para entrenar y validar el modelo creado a partir de un jupyter notebook [6] proporcionado por los mismos creadores del archivo que captura datos.

Fue posible entrenar y obtener el modelo que permite reconocer movimientos al utilizar el 60 % de los datos para entrenamiento, 20 % para validación y el restante 20 % para pruebas. El archivo generado posee el nombre model.h y fue utilizado junto al segundo programa independiente que se debe cargar al Arduino llamado IMU_Classifie.ino. Este programa posee la lógica mostrada en la Figura 6.

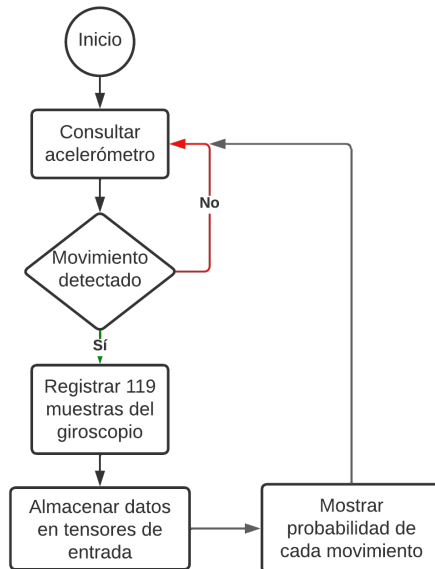


Figura 6. Funcionamiento de IMU_Classifier.ino

Funciona de forma similar en el sentido en que espera a que se registre un movimiento considerable del Arduino al utilizar el acelerómetro, si detecta el inicio de un nuevo movimiento entonces toma 119 muestras de este y los almacena en los tensores de entrada. Al final de recolectar todas las muestras, despliega el valor de probabilidad de detección para cada movimiento (estacionario, vueltas y Arriba-Abajo) en una escala de 0 a 1, siendo 1 la máxima confianza de detección del movimiento.

Para probar el correcto funcionamiento de la detección de los movimientos se cargó el programa y se comenzaron a probar uno a uno los 3 tipos de movimientos, el resultado de la identificación de ellos se muestra mediante la Figura 7.

```

2 Accelerometer sample rate = 119.00 Hz
3 Gyroscope sample rate = 119.00 Hz
4
5 Estacionario: 0.258001
6 Vueltas: 0.407079
7 Arriba-Abajo: 0.334920
8
9 Estacionario: 0.258461
10 Vueltas: 0.405647
11 Arriba-Abajo: 0.335892
12
13 Estacionario: 0.260021
14 Vueltas: 0.403321
15 Arriba-Abajo: 0.336659
16
17 Estacionario: 0.267434
18 Vueltas: 0.381068
19 Arriba-Abajo: 0.351498
20
21 Estacionario: 0.257989
22 Vueltas: 0.406749
23 Arriba-Abajo: 0.335263
24
25 Estacionario: 0.274860
26 Vueltas: 0.379800
27 Arriba-Abajo: 0.345340
28
29 Estacionario: 0.257345
30 Vueltas: 0.401570
31 Arriba-Abajo: 0.341085

```

Figura 7. Detección de movimientos.

Como se puede apreciar, los distintos movimientos no son claramente identificados por el Arduino NANO pese a completar el proceso de entrenamiento y obtener un modelo de forma correcta. El programa resulta confundido en esta identificación y, pese a producir múltiples movimientos con la tarjeta, se comunica que Vueltas es el más probable, seguido de Arriba-Abajo y por último Estacionario.

IV. CONCLUSIONES

- Fue posible entrenar un modelo a partir de la recolección de muestras de 3 tipos de movimientos distintos y utilizarlo dentro del dispositivo Arduino NANO 33 BLE.
- La comunicación entre el Arduino y la computadora fue utilizada con éxito para transmitir los datos de movimientos en formato csv y obtener la información necesaria para construir un modelo entrenado.
- El microcontrolador no fue capaz de reconocer los movimientos efectuados en la fase de pruebas con precisión, pese a realizar los mismos tipos de movimientos que se utilizaron para la fase de entrenamiento.
- La biblioteca TensorFlow Lite resulta bastante provechosa para integrar proyectos de ML en dispositivos Edge como resultan ser microcontroladores.

- Google Colab es una herramienta importante para poder ejecutar programas que demanden altos recursos, esto ya que se ejecuta en la web y brinda gran accesibilidad para desarrollar proyectos e incluso entrenar modelos de ML, como ocurrió en este laboratorio.

El repositorio con el desarrollo de este laboratorio se encuentra en: <https://github.com/Steven1mb/Lab-Microcontroladores/tree/Lab5>

IV-A. Bibliografía

REFERENCIAS

- [1] Arduino. (2023) Arduino nano 33 ble. [Online]. Available: <https://docs.arduino.cc/resources/datasheets/ABX00030-datasheet.pdf>
- [2] A. A. Steven Mora, "Laboratorio #3: Arduino: Gpio,adc y comunicaciones," 2023.
- [3] Roboticoss. Arduinotensorflowlitetutorials. [Online]. Available: <https://roboticoss.com/microcontroladores-para-aprendizaje-automatico-e-inteligencia-artificial/>
- [4] Oracle. ¿qué es la inteligencia artificial? obtén más información sobre la inteligencia artificial. [Online]. Available: <https://www.oracle.com/mx/artificial-intelligence/what-is-ai/>
- [5] Amazon. ¿qué es la inteligencia artificial? obtén más información sobre la inteligencia artificial. [Online]. Available: <https://aws.amazon.com/es/what-is/neural-network/>
- [6] S. Mistry. (2021) Arduinotensorflowlitetutorials. [Online]. Available: <https://github.com/arduino/ArduinoTensorFlowLiteTutorials>

IV-B. Anexos



1 The Board

As all Nano form factor boards, Nano 33 BLE does not have a battery charger but can be powered through USB or headers.

NOTE: Arduino Nano 33 BLE only supports 3.3V I/Os and is **NOT** 5V tolerant so please make sure you are not directly connecting 5V signals to this board or it will be damaged. Also, as opposed to Arduino Nano boards that support 5V operation, the 5V pin does NOT supply voltage but is rather connected, through a jumper, to the USB power input.

1.1 Application Examples

Sound spectrum: Create a sound spectrum to visualize sound frequencies. Connect an Arduino 33 Nano BLE and a microphone or amplifier.

Social distancing sensor: Keeping the social distance has become more important than ever to ensure your own, as well as others health. By connecting an Arduino Nano 33 BLE with a sensor and a LED display, you can create a wearable band that alerts you when you get too close to other people.

Healthy plant scanner: Watering your plants isn't always enough to keep them happy. Diseases, lack of sunlight etc. could also be vital factors for unhealthy plants. Keep your plants happy by creating a detector and train it to detect any diseases, all with an Arduino Nano 33 BLE

1.2 Ratings

1.2.1 Recommended Operating Conditions

| Symbol | Description | Min | Max |
|--------|--|-----------------|----------------|
| | Conservative thermal limits for the whole board: | -40 °C (40 °F) | 85°C (185 °F) |

1.3 Power Consumption

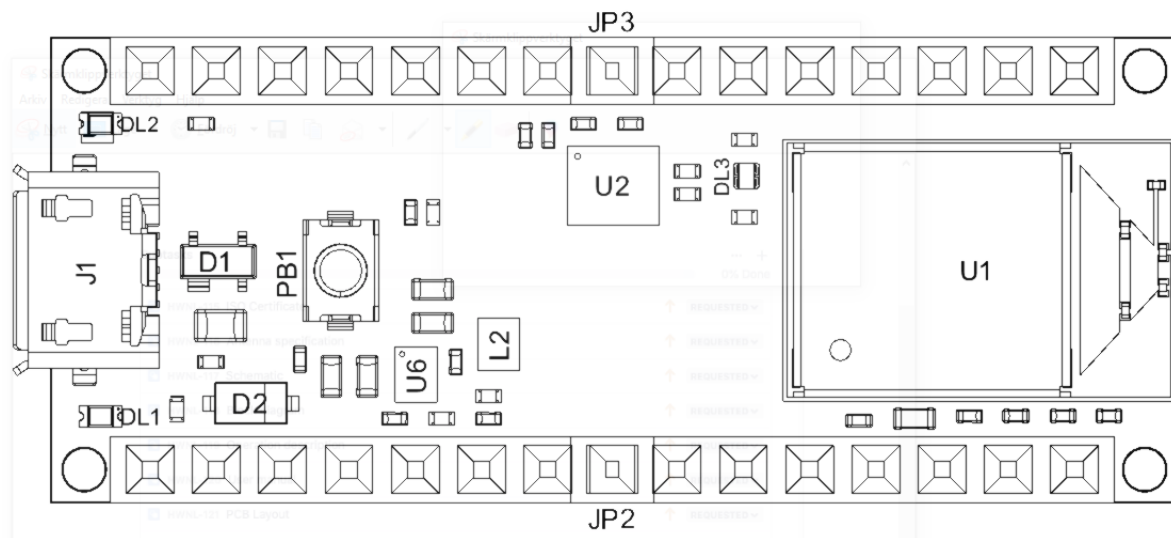
| Symbol | Description | Min | Typ | Max | Unit |
|--------|-------------------------------------|-----|-----|-----|------|
| PBL | Power consumption with busy loop | | TBC | | mW |
| PLP | Power consumption in low power mode | | TBC | | mW |
| PMAX | Maximum Power Consumption | | TBC | | mW |

2 Functional Overview

2.1 Board Topology



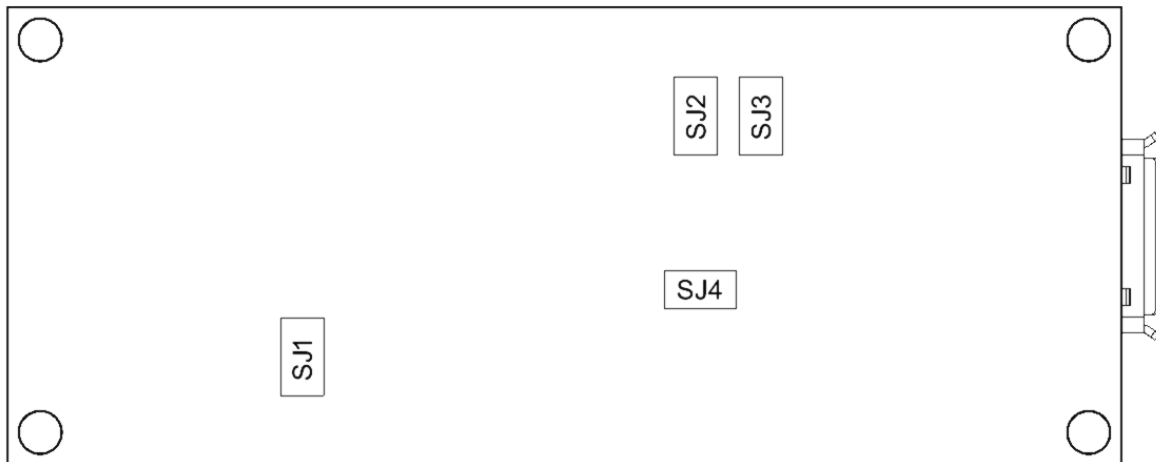
2.1.1 Top



Board topology Top

| Ref. | Description | Ref. | Description |
|------|---|------|----------------------------------|
| U1 | NINA-B306 Module Bluetooth® Low Energy 5.0 Module | U6 | MP2322GQH Step Down Converter |
| U2 | LSM9DS1TR Sensor IMU | PB1 | IT-1185AP1C-160G-GTR Push button |
| DL1 | Led L | DL2 | Led Power |

2.1.2 Bottom



Board topology bottom

| Ref. | Description | Ref. | Description |
|------|-------------|------|-------------|
| SJ1 | VUSB Jumper | SJ2 | D7 Jumper |
| SJ3 | D8 Jumper | SJ4 | 3v3 Jumper |



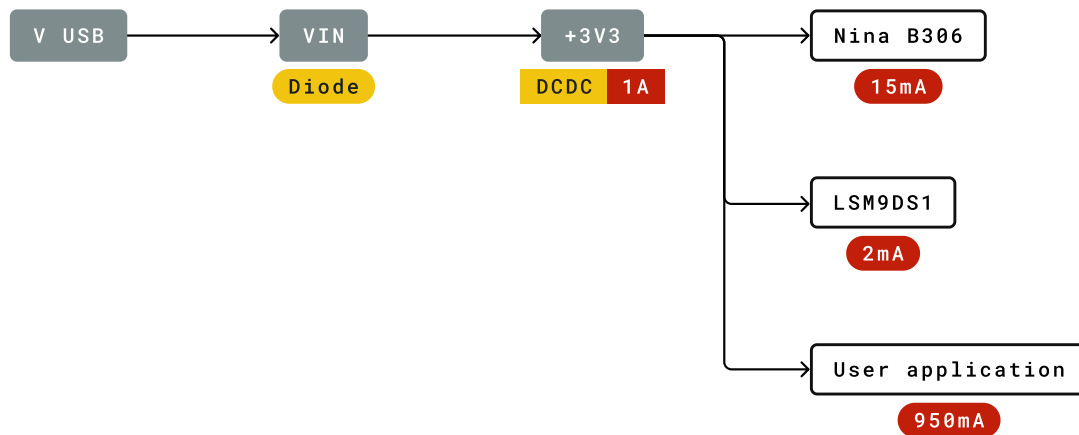
2.2 Processor

The Main Processor is a Cortex M4F running at up to 64MHz. Most of its pins are connected to the external headers, however some are reserved for internal communication with the wireless module and the on-board internal I²C peripherals (IMU and Crypto).

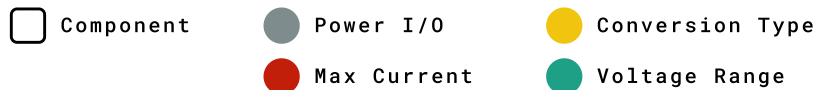
NOTE: As opposed to other Arduino Nano boards, pins A4 and A5 have an internal pull up and default to be used as an I²C Bus so usage as analog inputs is not recommended.

2.3 Power Tree

The board can be powered via USB connector, V_{IN} or V_{USB} pins on headers.



Legend :



Power tree

NOTE: Since V_{USB} feeds V_{IN} via a Schottky diode and a DC-DC regulator specified minimum input voltage is 4.5V the minimum supply voltage from USB has to be increased to a voltage in the range between 4.8V to 4.96V depending on the current being drawn.



3 Board Operation

3.1 Getting Started - IDE

If you want to program your Arduino Nano 33 BLE while offline you need to install the Arduino Desktop IDE [1] To connect the Arduino Nano 33 BLE to your computer, you'll need a Micro-B USB cable. This also provides power to the board, as indicated by the LED.

3.2 Getting Started - Arduino Web Editor

All Arduino boards, including this one, work out-of-the-box on the Arduino Web Editor [2], by just installing a simple plugin.

The Arduino Web Editor is hosted online, therefore it will always be up-to-date with the latest features and support for all boards. Follow [3] to start coding on the browser and upload your sketches onto your board.

3.3 Getting Started - Arduino IoT Cloud

All Arduino IoT enabled products are supported on Arduino IoT Cloud which allows you to Log, graph and analyze sensor data, trigger events, and automate your home or business.

3.4 Sample Sketches

Sample sketches for the Arduino Nano 33 BLE can be found either in the "Examples" menu in the Arduino IDE or in the "Documentation" section of the Arduino Pro website [4]

3.5 Online Resources

Now that you have gone through the basics of what you can do with the board you can explore the endless possibilities it provides by checking exciting projects on ProjectHub [5], the Arduino Library Reference [6] and the online store [7] where you will be able to complement your board with sensors, actuators and more

3.6 Board Recover

All Arduino boards have a built-in bootloader which allows flashing the board via USB. In case a sketch locks up the processor and the board is not reachable anymore via USB it is possible to enter bootloader mode by double-tapping the reset button right after power up.

4 Connector Pinouts