

Universidad de Costa Rica

Facultad de Ingeniería

Escuela de Ingeniería Eléctrica

IE0624 - Laboratorio de Microcontroladores

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

Prof. MSc. Marco Villalta Fallas

Por: Andrés Artavia Solano - B90789

Índice

1. Introducción	1
2. Nota Teórica	2
2.1. Arduino Nano 33 BLE	2
2.2. Giroscopio	3
2.3. Edge Impulse	3
2.4. TensorFlow Lite y redes neuronales	3
3. Desarrollo	5
3.1. Lectura de Giroscopio	5
3.2. Edge Impulse	5
3.2.1. Configuración de datos	5
3.2.2. Estructura del impulso	5
3.2.3. Procesamiento de datos	6
3.2.4. Clasificación	6
3.3. Exportación	6
3.4. Captura de resultados	7
4. Video Demostración	8
5. Análisis de los resultados	8
6. Conclusiones	10
7. Apéndice	12
7.1. Hoja de datos del Arduino Nano 33 BLE	12

1. Introducción

El objetivo de este proyecto es desarrollar un sistema de reconocimiento de actividad humana (HAR) utilizando un Arduino Nano 33 BLE, un microcontrolador versátil y de bajo consumo que integra sensores de movimiento y conectividad inalámbrica, para detectar y clasificar diferentes actividades humanas en tiempo real.

Para lograr este propósito, se empleó la plataforma Edge Impulse, que permite el entrenamiento de modelos de aprendizaje automático diseñados para dispositivos embebidos. Los datos del giroscopio integrado en el Arduino Nano 33 BLE fueron capturados mediante un script de python, procesados y utilizados para entrenar un modelo de clasificación basado en TensorFlow Lite. El modelo resultante fue implementado directamente en el microcontrolador, permitiendo la detección y clasificación de actividades humanas (mover la mano en círculos, dar un golpe o dejar la mano quieta) de manera eficiente y autónoma.

En este laboratorio se recolectaron datos del giroscopio en diferentes actividades humanas utilizando el Arduino Nano 33 BLE utilizando un programa que corre en el arduino y el cual envía los datos por el puerto serial para ser leídos por un script de python que los guarda en archivos ordenados de tipo csv. Los datos fueron procesados y utilizados para entrenar un modelo de clasificación en Edge Impulse. Se generó un modelo de TensorFlow Lite optimizado para el microcontrolador. El modelo entrenado fue cargado en el Arduino Nano 33 BLE, donde realiza la inferencia en tiempo real, detectando actividades en función de los datos del giroscopio.

Este proyecto combina el uso de tecnologías de aprendizaje automático, procesamiento de señales y computación en el borde para construir un sistema de detección inteligente en dispositivos de bajo costo, con aplicaciones potenciales en múltiples sectores. La implementación de esta solución destaca el potencial de la inteligencia artificial en microcontroladores para transformar datos en conocimiento útil y generar soluciones prácticas para problemas del mundo real.

Nota: El código fuente se puede encontrar en la rama "main" del repositorio, en la carpeta "lab5". <https://github.com/artavias/IE0624/tree/main>

2. Nota Teórica

A continuación se muestra un diagrama de pines del Arduino Nano 33 BLE:

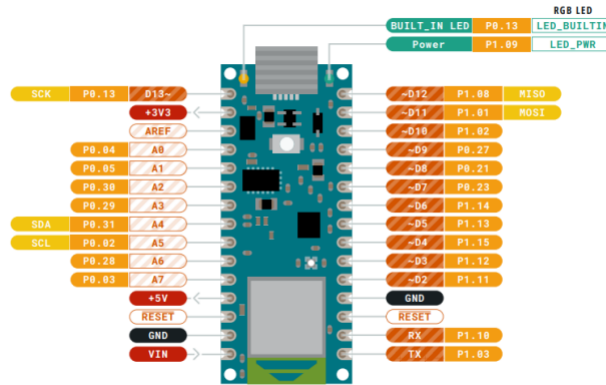


Figura 1: Diagrama de pines. [1]

2.1. Arduino Nano 33 BLE

El Arduino Nano 33 BLE está diseñada para facilitar la creación de aplicaciones de Internet de las Cosas (IoT), dispositivos portátiles y sistemas embebidos que requieren conectividad inalámbrica y capacidades avanzadas de procesamiento.

El Arduino Nano 33 BLE se basa en un microcontrolador de 32 bits ARM Cortex-M4 a 64 MHz. Ofrece una arquitectura eficiente y poderosa que permite el manejo de tareas complejas, como procesamiento de datos en tiempo real y ejecución de modelos de aprendizaje automático optimizados. Soporta comunicación BLE, lo que le permite conectar de manera eficiente dispositivos inalámbricos de bajo consumo, una capacidad clave para aplicaciones que requieren transmisión de datos a smartphones, computadoras u otros dispositivos IoT. Esta placa tiene un sensor de inercia que combina un acelerómetro, un giroscopio y un magnetómetro, proporcionando una solución completa para la detección y el seguimiento del movimiento en tres dimensiones. La integración de estos sensores en una única placa es fundamental para aplicaciones de Reconocimiento de Actividad Humana (HAR), ya que permite medir con precisión la aceleración lineal, la rotación y la orientación en tiempo real. Sin embargo para este laboratorio solo se utilizara el giroscopio el cual registra la velocidad angular, permitiendo detectar rotaciones y movimientos angulares. Magnetómetro: Proporciona datos sobre el campo magnético ambiental, facilitando la orientación espacial del dispositivo.

A diferencia de otras placas Arduino, el Nano 33 BLE cuenta con una capacidad de procesamiento mucho más alta, lo que le permite ejecutar modelos de aprendizaje automático ligeros, como aquellos generados por TensorFlow Lite, directamente en el dispositivo. Esto abre la puerta a aplicaciones de inteligencia artificial en tiempo real en el "borde" de la red, reduciendo la latencia y la necesidad de enviar datos a un servidor externo para su procesamiento.

En resumen, el Arduino Nano 33 BLE, con su potente microcontrolador y sus sensores, es una plataforma ideal para aplicaciones de inteligencia artificial embebida. En el contexto de un sistema de Reconocimiento de Actividad Humana, su capacidad de procesar datos en tiempo real y ejecutarlos en modelos de aprendizaje automático en el "borde" demuestra su versatilidad y potencial para resolver problemas del mundo real con una solución compacta y de bajo costo.

[2]

2.2. Giroscopio

El giroscopio incluido en el Arduino Nano 33 BLE mide la velocidad angular del dispositivo en torno a sus tres ejes principales (X, Y, Z). Al registrar cambios en la velocidad de rotación, es posible identificar patrones de movimiento característicos que pueden asociarse con actividades específicas. En aplicaciones de HAR, los datos del giroscopio son utilizados como características de entrada para modelos de aprendizaje automático para luego realizar una inferencia y detectar los movimientos en tiempo real.

2.3. Edge Impulse

Edge Impulse es una plataforma que facilita la creación, entrenamiento e implementación de modelos de aprendizaje automático en dispositivos embebidos. Con Edge Impulse, es posible capturar datos directamente desde un microcontrolador, etiquetar muestras, entrenar modelos de clasificación y generar modelos optimizados para ejecutarse en hardware con recursos limitados, como el Arduino Nano 33 BLE. El modelo entrenado puede exportarse como un modelo de TensorFlow Lite, el cual está diseñado para ser ejecutado eficientemente en dispositivos con baja capacidad de cómputo. [3]

2.4. TensorFlow Lite y redes neuronales

A continuación se muestra el diagrama de una neurona típica en una red neuronal.

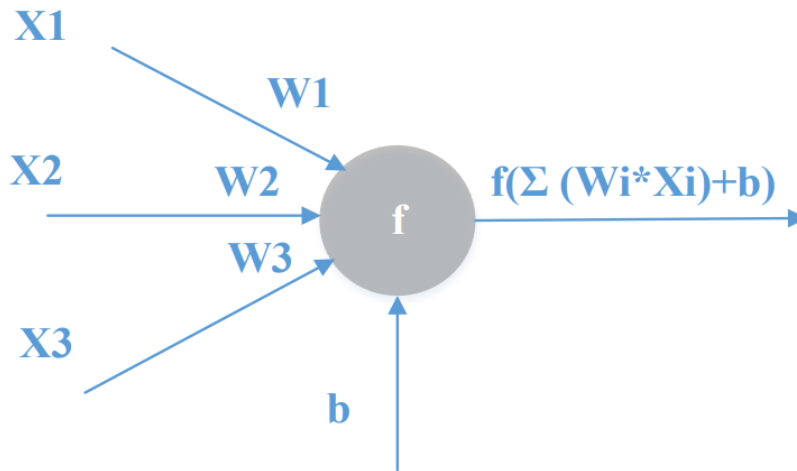


Figura 2: Neurona básica de una red neuronal.[4]

Como se muestra en la figura, la función de cada neurona es realizar el siguiente cálculo matemático:

$$output = f \left(\sum (W_i * X_i) + b \right) \quad (1)$$

En donde W_i representa el vector de pesos (weight) y X_i el vector de entradas a la neurona, b sería el sesgo (bias) y la función f es una función de activación como ReLU, tanh entre otras.

Estas neuronas se interconectan entre ellas para formar una red neuronal como se muestra a continuación.

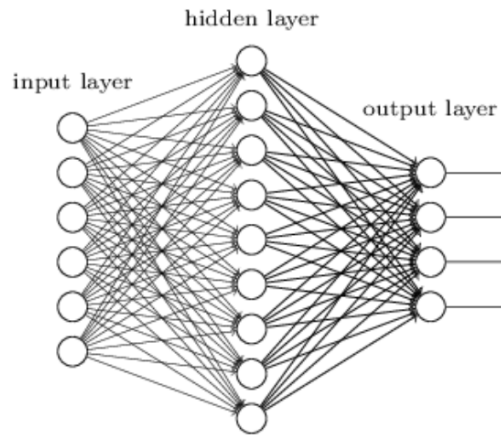


Figura 3: Red Neuronal [4]

Para este laboratorio se tiene una red neuronal de 3 capas, una capa de entrada y una capa escondida que utilizan una función de activación ReLU. Esta función retorna 0 si el valor de entrada es menor o igual a 0 o retorna el mismo valor de entrada si este es mayor a 0. La capa de salida posee solo tres neuronas, cada una corresponde a un tipo de movimiento que se desea detectar. En este caso se entreno a la red neuronal para detectar un movimiento circular, un movimiento de golpe y un caso donde el dispositivo se encuentra estacionario. Esta capa de salida aplica una función softmax, la cual toma los tres valores de salida y transforma cada uno en un valor de 0 a 1 de manera que la suma de los tres valores sumen 1, es decir se puede tomar este valor como un porcentaje de certeza de la red neuronal.

TensorFlow Lite es una versión optimizada de TensorFlow para dispositivos móviles y sistemas embebidos. Permite realizar inferencias en tiempo real con baja latencia y bajo consumo de recursos, lo que resulta fundamental para aplicaciones de HAR en microcontroladores. Para este laboratorio se exporta un modelo entrenado de TensorFlow Lite desde la plataforma Edge Impulse para luego correr en el Arduino Nano 33 BLE y hacer inferencias en tiempo real sobre el movimiento que se esta realizando.

3. Desarrollo

3.1. Lectura de Giroscopio

El primer punto que se debe resolver es realizar la lectura de los valores en los ejes X, Y y Z del giroscopio. Para esto se implementó un pequeño programa de Arduino que lee los datos del giroscopio y los envía por el puerto serial para luego ser leídos por un script de Python, el cual primero indica al usuario qué movimiento realizar con la placa y posteriormente lee los datos, los cuales ordena en archivos csv con su debida etiqueta y marca de tiempo. La etiqueta y la marca de tiempo son columnas en los archivos csv que se utilizan para entrenar el modelo e indicarle cuál es el movimiento que se está realizando. El código de Arduino para esta parte se puede encontrar en la carpeta “gyro” del repositorio y el código de Python corresponde al archivo llamado “captura_datos.py”. Cabe mencionar que esta parte del proyecto es esencial para garantizar una captura de datos lo mas limpia y real posible, de manera ordenada y que facilite la creación de datos para el posterior entrenamiento de la red neuronal.

3.2. Edge Impulse

3.2.1. Configuración de datos

Una vez capturados los datos (24 archivos csv por cada movimiento) se procedió a subir los datos a la plataforma de Edge Impulse para entrenar el modelo. Edge Impulse ofrece un procesamiento de datos automático en donde se configura primero un “CSV Wizard” para que la herramienta pueda leer los archivos CSV, en esta parte se indican las columnas de los datos, el tiempo y la etiqueta del movimiento que se está realizando.

A la hora de subir los datos se clasifican en datos para entrenamiento o validación, en este caso se optó por utilizar aproximadamente 80 % de los datos para entrenamiento y 20 % para validaciones.

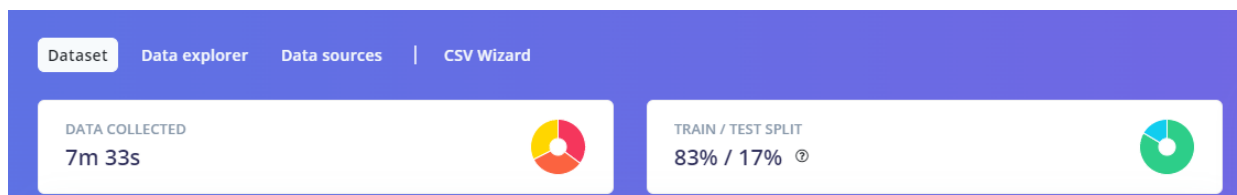


Figura 4: Distribución de los datos

3.2.2. Estructura del impulso

Habiendo subido los datos el siguiente paso es crear un “impulse”. Un impulso o “impulse^{en} Edge Impulse es un conjunto de bloques de procesamiento de datos y modelos de aprendizaje que toma datos sin procesar y retorna la clasificación de estos.

Primero se configuro el procesamiento de datos el cual detecta ventanas de 2000ms con pequeños incrementos de 90ms, esto es asumiendo que se toma aproximadamente 2 segundos en realizar por completo cada acción de mover en círculos o dar un golpe (o en su defecto permanecer estacionario). Estas ventanas de 2000ms son alimentadas a un bloque de analisis espectral. Un bloque de análisis espectral en Edge Impulse se utiliza para transformar datos crudos de una señal en información representativa de las frecuencias que componen esa señal. Este tipo de análisis es especialmente útil cuando se trabaja con datos que presentan patrones periódicos o de variación temporal, como los datos de audio, vibración o movimiento obtenidos de sensores como acelerómetros y giroscopios y como se mostrara mas adelante resultado bastante útil para detectar actividad humana (HAR).

El siguiente bloque corresponde al bloque de clasificación, o lo que es propiamente el modelo de aprendizaje automático utilizando una red neuronal que se encargara de clasificar los datos en los tres movimientos previamente mencionados, para dar a la salida tres valores diferentes. El valor mas alto después de realizar la inferencia es el que se considera como la predicción de los datos procesados otorgada por la red neuronal.

A continuación se muestra la ventana de configuración del impulso:

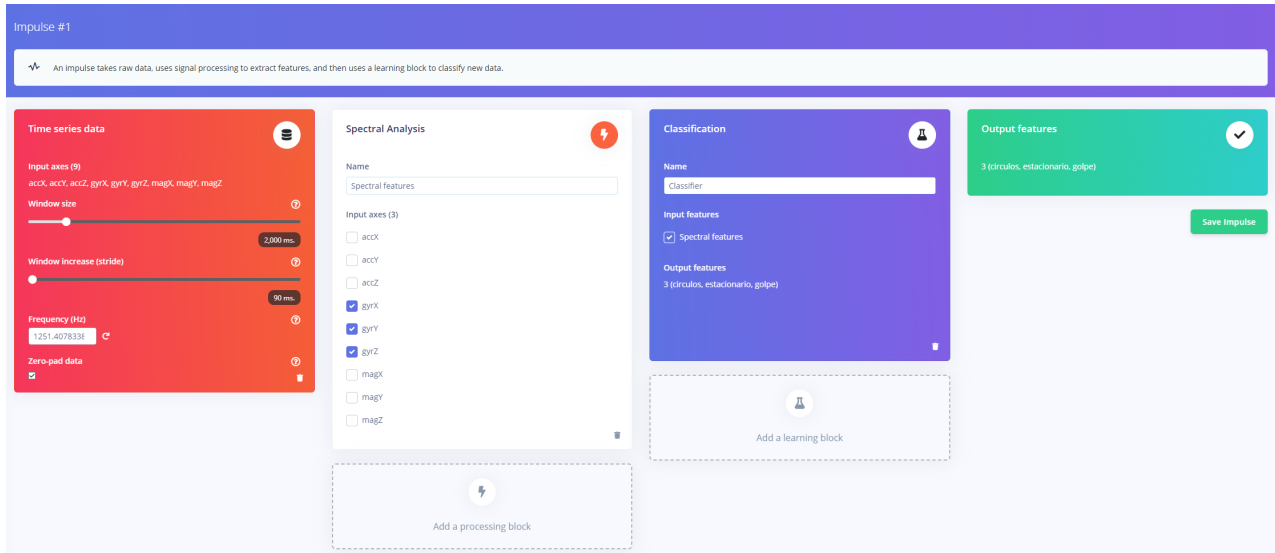


Figura 5: Configuración del impulso.

3.2.3. Procesamiento de datos

En el bloque de análisis espectral mencionado anteriormente, inicialmente se configuro para realizar una Transformada Rápida de Fourier (FFT) con 16 puntos pero resulto mas eficiente realizarla con 32 puntos. De esta manera el bloque de análisis espectral convierte datos crudos en información de la frecuencia esencial, facilitando la clasificación y el análisis de señales con patrones periódicos o complejos.

3.2.4. Clasificación

El bloque de clasificación se configuro con tres capas, la primera capa es de 20 neuronas con una función de activación ReLU, la segunda capa se configuro con 10 neuronas igualmente con una función de activación ReLU para finalmente pasar a una tercera capa de salida de 3 neuronas con una función de activación softmax para dar una predicción final mapeada en un valor de 0 a 1, que ademas sumados los valores de las tres neuronas se obtiene un resultado de 1. El código de la configuración de la red neuronal se puede encontrar en el repositorio del proyecto, en la carpeta "lab5" como "arquitectura_modelo.py".

3.3. Exportación

Una vez teniendo la configuración de todos los bloques lista se procedió a realizar el entrenamiento del modelo en la misma plataforma de Edge Impulse. Como se muestra en la imagen a continuación, se obtuvo una exactitud del 98.5 %.

Last training performance (validation set)

ACCURACY
98.5%LOSS
0.04

Confusion matrix (validation set)

	CIRCULOS	ESTACIONARIO	GOLPE
CIRCULOS	97.0%	0%	3.0%
ESTACIONARIO	0%	100%	0%
GOLPE	1.3%	0%	98.7%
F1 SCORE	0.98	1.00	0.98

Figura 6: Exactitud del modelo.

Habiendo obtenido este resultado se procedió a exportar el modelo como una librería de Arduino llamada “ei-har_lab5-arduino” que utiliza TensorFlow Lite para posteriormente poder correr el modelo en el Arduino Nano 33 Ble. El código de Arduino propiamente que corre el modelo se encuentra en el directorio “modeloz” esta basado en el ejemplo que se crea automáticamente en la librería exportada de Edge Impulse. Para compilarlo hay que instalar antes la biblioteca de Arduino exportada anteriormente.

3.4. Captura de resultados

Finalmente se creó un script de Python llamado “captura_modelo.py” diseñado para capturar los datos del arduino cuando está corriendo el modelo y realizando inferencias. Similar a la captura de datos para el entrenamiento del modelo, el script lee los datos enviados por el microcontrolador por el puerto serial para guardarlos en un archivo de texto, hasta que se recibe una señal de interrupción.

4. Video Demostración

Un video de la demostración del proyecto se puede encontrar en el siguiente enlace: <https://www.youtube.com/shorts/Ub2hezA1R5U>

5. Análisis de los resultados

La siguiente imagen presenta una muestra de los resultados propiciados por el arduino, capturados por el script de Python:

```
resultado_modelo.txt
1  Inferencia de actividad humana utilizando giroscopio
2
3  gyrX Inicializacion de sensor de eje exitosa.
4  gyrY Inicializacion de sensor de eje exitosa.
5  gyrZ Inicializacion de sensor de eje exitosa.
6
7  Comenzando inferencia en 1 segundo:
8  Tomando muestras...
9  Predicciones:
10 circulos: 0.01172
11 estacionario: 0.00000
12 golpe: 0.98828
13
14 Comenzando inferencia en 1 segundo:
15 Tomando muestras...
16 Predicciones:
17 circulos: 0.96484
18 estacionario: 0.00000
19 golpe: 0.03516
20
21 Comenzando inferencia en 1 segundo:
22 Tomando muestras...
23 Predicciones:
24 circulos: 0.00000
25 estacionario: 0.00000
26 golpe: 0.99609
27
```

Figura 7: Resultados del modelo.

En la figura 7 se puede observar como el Arduino primero indica la inicialización de los sensores del giroscopio para proceder a tomar datos inmediatamente y poder realizar la predicción del movimiento. Se muestran tres predicciones distintas las cuales corresponden a un movimiento de golpe, un movimiento en circulos y finalmente otro de golpe respectivamente, para los tres casos se logra una predicción con una confianza mayor al 96 %.

El modelo desarrollado y ejecutado en el Arduino Nano 33 BLE demuestra un desempeño notable en términos de precisión y eficiencia, logrando una clasificación precisa de las actividades humanas detectadas por el giroscopio. El modelo alcanzó una exactitud del 98.5 % durante la etapa de pruebas. Este alto nivel de precisión refleja la capacidad del modelo, el entrenamiento realizado con Edge Impulse permitió identificar patrones claros en los datos provenientes del giroscopio, asociándolos correctamente a las clases definidas de “circulos”, “estacionario.” “golpe”.

Aquí es importante recalcar la relevancia de entrenar el modelo con datos limpios y confiables. La captura de datos del sensor utilizando el programa de arduino junto con el script de Python permite obtener una gran cantidad de datos ordenados y con su respectiva etiqueta y

distribuidos equitativamente (igual cantidad de datos para cada movimiento o clase) en cuestión de pocos minutos, lo que facilita la creación de estos y aumenta la capacidad de alimentar al modelo con una gran cantidad de datos. Además, el bloque de procesamiento aplicado (análisis espectral) logró transformar los datos crudos del sensor en características relevantes para la clasificación, minimizando la influencia del ruido en las inferencias.

Como se observa en la imagen 7, los resultados obtenidos son concisos y demuestran la exactitud del modelo, además de su capacidad para tomar datos y realizar la inferencia en tiempo real, clasificando el movimiento realizado en cuestión de milésimas de segundos.

6. Conclusiones

El desarrollo de un sistema de reconocimiento de actividad humana (HAR) utilizando el Arduino Nano 33 BLE, Edge Impulse y un script de Python para capturar resultados demuestra la efectividad de combinar tecnologías de inteligencia artificial embebida con herramientas de procesamiento de datos. El modelo creado, con una exactitud del 98.5 %, evidencia un alto nivel de desempeño, mostrando la capacidad del sistema para clasificar actividades humanas (HAR) de manera eficiente.

El uso de Edge Impulse facilitó el proceso de recopilación y procesamiento de datos, así como la generación de un modelo optimizado para el hardware. Por su parte, el script en Python permitió la integración del sistema con una computadora para capturar las inferencias en tiempo real y almacenarlas en un archivo de texto, mejorando la trazabilidad y el análisis posterior.

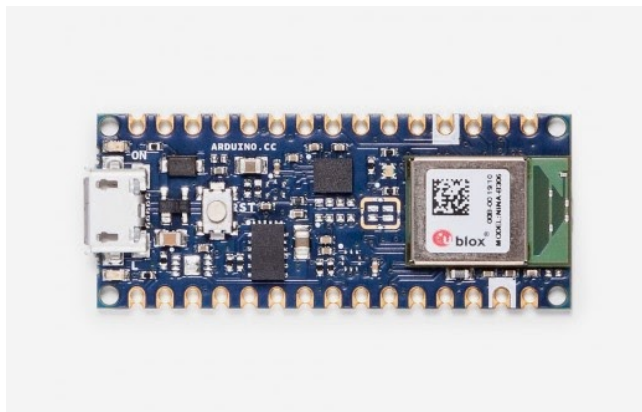
Este proyecto no solo destacó la potencia del Arduino Nano 33 BLE como plataforma para aplicaciones de inteligencia artificial, sino que también reforzó el conocimiento en la integración de herramientas de aprendizaje automático. En resumen, el laboratorio logró implementar un sistema HAR funcional y preciso, demostrando la viabilidad de construir soluciones avanzadas con dispositivos de bajo costo y recursos limitados.

Referencias

- [1] Arduino. Arduino nano 33 ble datasheet. <https://docs.arduino.cc/resources/datasheets/ABX00030-datasheet.pdf>.
- [2] Arduino. Nano 33 ble features. <https://docs.arduino.cc/hardware/nano-33-ble/#features>.
- [3] 330ohms. Que es edge impulse. https://www.330ohms.com/blogs/blog/que-es-edge-impulse?srsltid=AfmB0oqzuG0eke_PyzcCa6E1arFXbaJoE6I4apMDuUHtDRqeQckrLUoZ, 2024.
- [4] Michael Nielsen. Why are deep neural networks hard to train? <http://neuralnetworksanddeeplearning.com/chap5.html>, 2019.

7. Apéndice

7.1. Hoja de datos del Arduino Nano 33 BLE



Description

Arduino® Nano 33 BLE is a miniature sized module containing a NINA B306 module, based on Nordic nRF52480 and containing an Arm® Cortex®-M4F and a 9-axis IMU. The module can either be mounted as a DIP component (when mounting pin headers), or as a SMT component, directly soldering it via the castellated pads.

Target areas:

Maker, enhancements, basic IoT application scenarios

Features

- **NINA B306 Module**
 - **Processor**
 - 64 MHz Arm® Cortex®-M4F (with FPU)
 - 1 MB Flash + 256 kB RAM
 - **Bluetooth® 5 multiprotocol radio**
 - 2 Mbps
 - CSA #2
 - Advertising Extensions
 - Long Range
 - +8 dBm TX power
 - -95 dBm sensitivity
 - 4.8 mA in TX (0 dBm)
 - 4.6 mA in RX (1 Mbps)
 - Integrated balun with 50 Ω single-ended output
 - IEEE 802.15.4 radio support
 - Thread
 - Zigbee®
- **Peripherals**
 - Full-speed 12 Mbps USB
 - NFC-A tag
 - Arm CryptoCell CC310 security subsystem
 - QSPI/SPI/TWI/I²S/PDM/QDEC
 - High speed 32 MHz SPI
 - Quad SPI interface 32 MHz
 - EasyDMA for all digital interfaces
 - 12-bit 200 ksps ADC
 - 128 bit AES/ECB/CCM/AAR co-processor
- **LSM9DS1** (9-axis IMU)
 - 3 acceleration channels, 3 angular rate channels, 3 magnetic field channels
 - $\pm 2/\pm 4/\pm 8/\pm 16$ g linear acceleration full scale
 - $\pm 4/\pm 8/\pm 12/\pm 16$ gauss magnetic full scale
 - $\pm 245/\pm 500/\pm 2000$ dps angular rate full scale
 - 16-bit data output
- **MPM3610** DC-DC
 - Regulates input voltage from up to 21V with a minimum of 65% efficiency @minimum load
 - More than 85% efficiency @12V



Contents

1 The Board	4
1.1 Application Examples	4
1.2 Ratings	4
1.2.1 Recommended Operating Conditions	4
1.3 Power Consumption	4
2 Functional Overview	4
2.1 Board Topology	4
2.1.1 Top	4
2.1.2 Bottom	5
2.2 Processor	6
2.3 Power Tree	6
3 Board Operation	7
3.1 Getting Started - IDE	7
3.2 Getting Started - Arduino Cloud Editor	7
3.3 Getting Started - Arduino Cloud	7
3.4 Sample Sketches	7
3.5 Online Resources	7
3.6 Board Recover	7
4 Connector Pinouts	8
4.1 USB	8
4.2 Headers	8
4.3 Debug	9
5 Mechanical Information	9
5.1 Board Outline and Mounting Holes	9
6 Certifications	10
6.1 Declaration of Conformity CE DoC (EU)	10
6.2 Declaration of Conformity to EU RoHS & REACH 211 01/19/2021	10
6.3 Conflict Minerals Declaration	11
7 FCC Caution	11
8 Company Information	12
9 Reference Documentation	12
10 Revision History	12



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: 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 Nano 33 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 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)

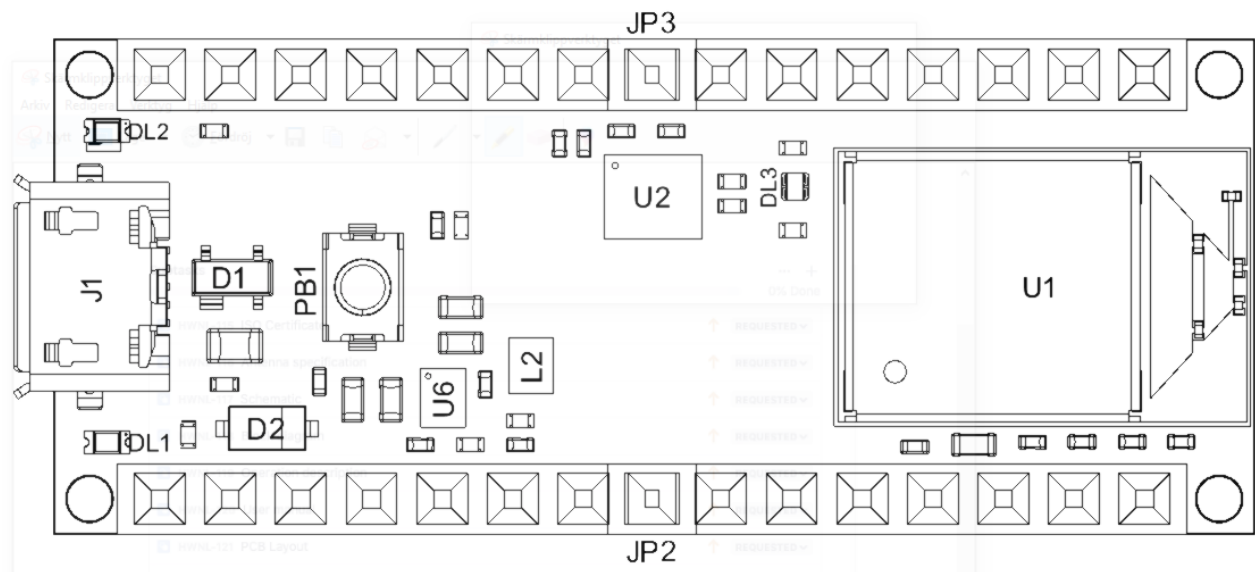
1.3 Power Consumption

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

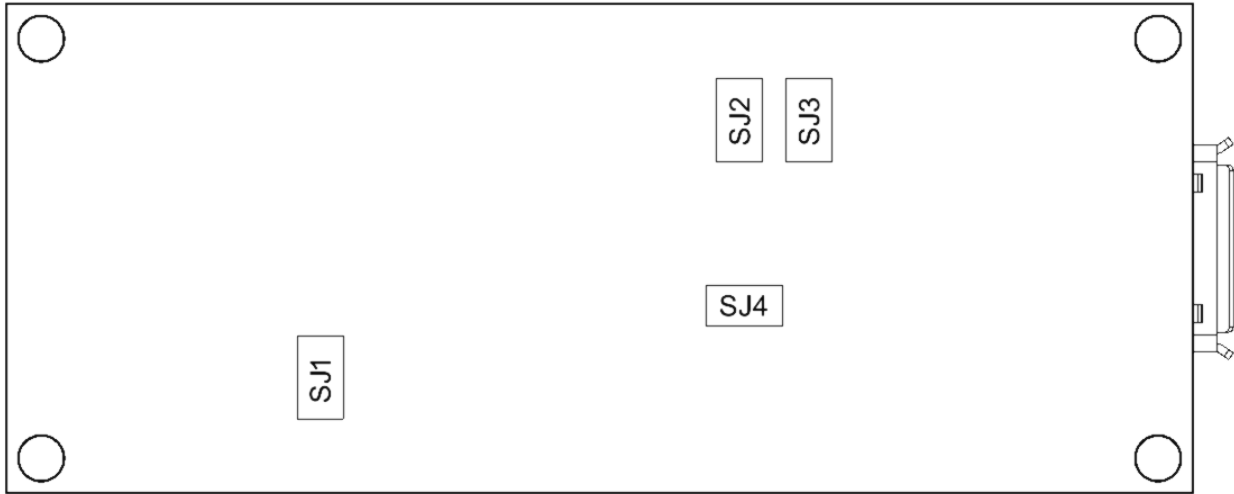
2.1.1 Top



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

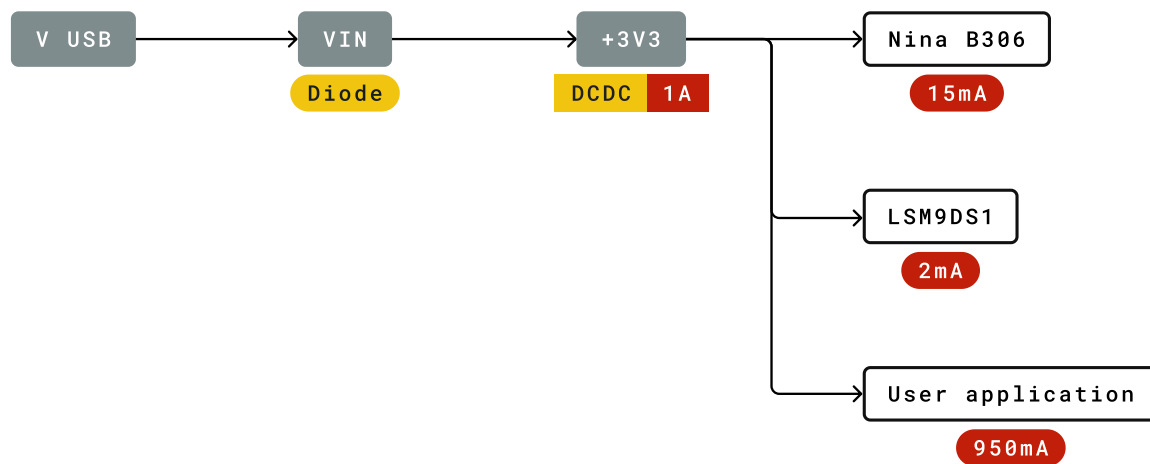
2.2 Processor

The Main Processor is a Arm® 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).

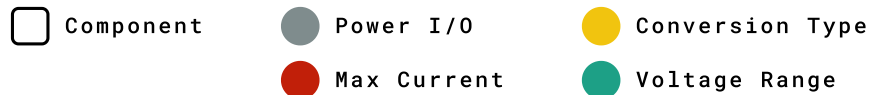
NOTE: As opposed to other Arduino Nano boards, pins A4 and A5 have an internal pull up and default to be used as an I²C Bus so usage as analog inputs is not recommended.

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 Nano 33 BLE while offline you need to install the Arduino Desktop IDE [1] To connect the 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 Cloud Editor

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

The Arduino Cloud 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 Cloud

All Arduino IoT enabled products are supported on Arduino 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 Nano 33 BLE can be found either in the "Examples" menu in the Arduino IDE or in Arduino Docs [4].

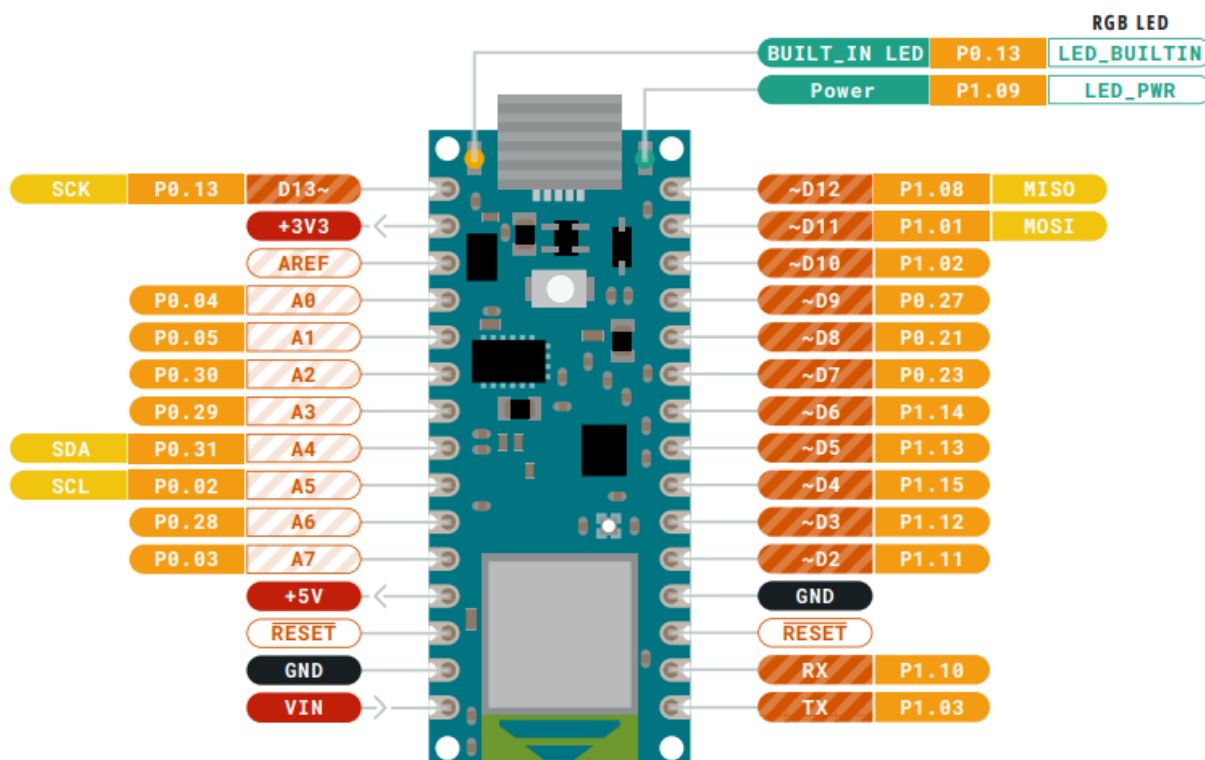
3.5 Online Resources

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 Arduino Project Hub [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



Pinout

4.1 USB

Pin	Function	Type	Description
1	VUSB	Power	Power Supply Input. If board is powered via VUSB from header this is an Output (1)
2	D-	Differential	USB differential data -
3	D+	Differential	USB differential data +
4	ID	Analog	Selects Host/Device functionality
5	GND	Power	Power Ground

4.2 Headers

The board exposes two 15 pin connectors which can either be assembled with pin headers or soldered through castellated vias.

Pin	Function	Type	Description
1	D13	Digital	GPIO/Built-in LED
2	+3V3	Power Out	Internally generated power output to external devices
3	AREF	Analog	Analog Reference; can be used as GPIO
4	A0/DAC0	Analog	ADC in/DAC out; can be used as GPIO
5	A1	Analog	ADC in; can be used as GPIO
6	A2	Analog	ADC in; can be used as GPIO
7	A3	Analog	ADC in; can be used as GPIO
8	A4/SDA	Analog	ADC in; I2C SDA; Can be used as GPIO (1)
9	A5/SCL	Analog	ADC in; I2C SCL; Can be used as GPIO (1)
10	A6	Analog	ADC in; can be used as GPIO
11	A7	Analog	ADC in; can be used as GPIO
12	VUSB	Power In/Out	Normally NC; can be connected to VUSB pin of the USB connector by shorting a jumper

Pin	Function	Type	Description
13	RST	Digital In	Active low reset input (duplicate of pin 18)
14	GND	Power	Power Ground
15	VIN	Power In	Vin Power input
16	TX	Digital	USART TX; can be used as GPIO
17	RX	Digital	USART RX; can be used as GPIO
18	RST	Digital	Active low reset input (duplicate of pin 13)
19	GND	Power	Power Ground
20	D2	Digital	GPIO
21	D3/PWM	Digital	GPIO; can be used as PWM
22	D4	Digital	GPIO
23	D5/PWM	Digital	GPIO; can be used as PWM
24	D6/PWM	Digital	GPIO, can be used as PWM
25	D7	Digital	GPIO
26	D8	Digital	GPIO
27	D9/PWM	Digital	GPIO; can be used as PWM
28	D10/PWM	Digital	GPIO; can be used as PWM
29	D11/MOSI	Digital	SPI MOSI; can be used as GPIO
30	D12/MISO	Digital	SPI MISO; can be used as GPIO

4.3 Debug

On the bottom side of the board, under the communication module, debug signals are arranged as 3x2 test pads with 100 mil pitch with pin 4 removed. Pin 1 is depicted in Figure 3 – Connector Positions

Pin	Function	Type	Description
1	+3V3	Power Out	Internally generated power output to be used as voltage reference
2	SWD	Digital	nRF52480 Single Wire Debug Data
3	SWCLK	Digital In	nRF52480 Single Wire Debug Clock
5	GND	Power	Power Ground
6	RST	Digital In	Active low reset input
1	+3V3	Power Out	Internally generated power output to be used as voltage reference

5 Mechanical Information

5.1 Board Outline and Mounting Holes

The board measures are mixed between metric and imperial. Imperial measures are used to maintain 100 mil pitch grid between pin rows to allow them to fit a breadboard whereas board length is Metric





REACH regulations) and Substances of Very High Concern (SVHC) in any significant amounts as specified by the Annex XVII of Candidate list published by ECHA (European Chemical Agency) 1907 /2006/EC.

6.3 Conflict Minerals Declaration

As a global supplier of electronic and electrical components, Arduino is aware of our obligations with regards to laws and regulations regarding Conflict Minerals, specifically the Dodd-Frank Wall Street Reform and Consumer Protection Act, Section 1502. Arduino does not directly source or process conflict minerals such as Tin, Tantalum, Tungsten, or Gold. Conflict minerals are contained in our products in the form of solder, or as a component in metal alloys. As part of our reasonable due diligence Arduino has contacted component suppliers within our supply chain to verify their continued compliance with the regulations. Based on the information received thus far we declare that our products contain Conflict Minerals sourced from conflict-free areas.

7 FCC Caution

Any Changes or modifications not expressly approved by the party responsible for compliance could void the user's authority to operate the equipment.

This device complies with part 15 of the FCC Rules. Operation is subject to the following two conditions:

- (1) This device may not cause harmful interference
- (2) this device must accept any interference received, including interference that may cause undesired operation.

FCC RF Radiation Exposure Statement:

1. This Transmitter must not be co-located or operating in conjunction with any other antenna or transmitter.
2. This equipment complies with RF radiation exposure limits set forth for an uncontrolled environment.
3. This equipment should be installed and operated with minimum distance 20cm between the radiator & your body.

English: User manuals for license-exempt radio apparatus shall contain the following or equivalent notice in a conspicuous location in the user manual or alternatively on the device or both. This device complies with Industry Canada license-exempt RSS standard(s). Operation is subject to the following two conditions:

- (1) this device may not cause interference
- (2) this device must accept any interference, including interference that may cause undesired operation of the device.

French: Le présent appareil est conforme aux CNR d'Industrie Canada applicables aux appareils radio exempts de licence. L'exploitation est autorisée aux deux conditions suivantes :

- (1) l' appareil n' doit pas produire de brouillage
- (2) l' utilisateur de l' appareil doit accepter tout brouillage radioélectrique subi, même si le brouillage est susceptible d'en compromettre le fonctionnement.

IC SAR Warning:

English This equipment should be installed and operated with minimum distance 20 cm between the radiator and your body.

French: Lors de l' installation et de l' exploitation de ce dispositif, la distance entre le radiateur et le corps est d' au moins 20 cm.

Important: The operating temperature of the EUT can't exceed 85°C and shouldn't be lower than -40°C.

Hereby, Arduino S.r.l. declares that this product is in compliance with essential requirements and other relevant provisions of Directive 2014/53/EU. This product is allowed to be used in all EU member states.

Frequency bands	Maximum output power (ERP)
863-870Mhz	5.47 dBm



8 Company Information

Company name	Arduino S.r.l
Company Address	Via Andrea Appiani 25 20900 MONZA Italy

9 Reference Documentation

Reference	Link
Arduino IDE (Desktop)	https://www.arduino.cc/en/software
Arduino Cloud Editor	https://create.arduino.cc/editor
Arduino Cloud Editor - Getting Started	https://docs.arduino.cc/arduino-cloud/guides/editor/
Arduino Documentation	https://docs.arduino.cc
Arduino Project Hub	https://create.arduino.cc/projecthub?by=part&part_id=11332&sort=trending
Library Reference	https://www.arduino.cc/reference/en/
Arduino Store	https://store.arduino.cc/
Forum	http://forum.arduino.cc/
SAMD21G18	https://ww1.microchip.com/downloads/aemDocuments/documents/MCU32/ProductDocuments/DataSheets/SAM-D21DA1-Family-Data-Sheet-DS40001882G.pdf
NINA W102	https://content.u-blox.com/sites/default/files/NINA-W10_DataSheet_UBX-17065507.pdf
ECC608	https://ww1.microchip.com/downloads/aemDocuments/documents/SCBU/ProductDocuments/DataSheets/ATECC608A-CryptoAuthentication-Device-Summary-Data-Sheet-DS40001977B.pdf
MPM3610	https://www.monolithicpower.com/pub/media/document/MPM3610_r1.01.pdf
NINA Firmware	https://github.com/arduino/nina-fw
ECC608 Library	https://github.com/arduino-libraries/ArduinoECCX08
LSM6DSL Library	https://github.com/stm32duino/LSM6DSL

10 Revision History

Date	Revision	Changes
25/04/2024	3	Updated link to new Cloud Editor
03/08/2022	2	Reference documentation links updates
21/04/2021	1	General datasheet updates