

Universidad de Costa Rica

Facultad de Ingeniería

Escuela de Ingeniería Eléctrica

IE0624 - Proyecto

III ciclo 2023

Título: Smart Door Lock

Estudiantes:

Kevin Campos Castro

Josué Salmerón Córdoba

Grupo 1

Profesor: Marco Villalta

21 de febrero de 2024

Índice

1. Resumen	1
2. Objetivos	2
2.1. Objetivos General	2
2.2. Objetivos Específicos	2
3. Alcances	3
4. Nota teórica	3
4.0.1. Características generales	6
4.0.2. Diagrama de bloques	6
4.0.3. Diagrama de pines	7
4.0.4. Características eléctricas	7
5. Desarrollo/Análisis	11
6. Conclusiones y recomendaciones	18
7. Anexos	20

Índice de figuras

1.	Ubicación del micrófono. Tomado de [1].	3
2.	Diagrama de bloques del Nano BLE 33 Sense . Tomado de [2].	4
3.	Diagrama de pines del Nano BLE 33 Sense . Tomado de [3].	5
4.	Características eléctricas de nRF52480. Tomado de [2].	5
5.	Características eléctricas de la placa. Tomado de [3].	5
6.	Diagrama de bloques del arduino mega 2560 REV3 [4].	6
7.	Diagrama de pines del arduino mega 2560 REV3 [4].	7
8.	Condiciones de operación del arduino mega 2560 REV3 [4].	7
9.	Keypad	8
10.	Módulo I2C.	8
11.	Pantalla lcd.	9
12.	Servomotor.	9
13.	Habilitación de 5 V.	10
14.	Diagrama de bloques smart door lock.	11
15.	Conjunto de datos.	12
16.	Espectro señal abrir.	12
17.	Espectro señal cierra.	12
18.	Espectro del ruido.	13
19.	Espectro de sonido desconocido.	13
20.	Clasificación de los datos.	13
21.	Resultado fina del entrenamiento.	14
22.	Reconocimiento de la palabra abrir.	14
23.	Reconocimiento de la palabra cierra.	14
24.	Accionamiento del servo con la palabra abrir.	15
25.	Accionamiento del servo con la palabra cierra.	15
26.	Funcionamiento completo con la palabra abrir.	15
27.	Funcionamiento completo con la palabra cierra.	15
28.	Cerradura final ensamblado parte frontal.	16
29.	Cerradura final ensamblado parte trasera.	17

Índice de tablas

1. Lista de equipos 10

1. Resumen

Este trabajo se implementó un smart door lock que consta de dos cerraduras y está dividido en las siguientes secciones. En la primera se hace uso de TinyML con un Arduino Nano 33 Ble Sense para el reconocimiento de voz. La segunda sección se hace uso de un microcontrolador Arduino Mega2560 un keypad, una pantalla lcd para una mejorar interacción con el usuario. Ambas secciones tienen un servo motor en común y se encargarán de realizar la misma tarea, es decir, abrir y cerrar la puerta de dos maneras distintas. Donde se obtuvieron los resultados esperados cumpliendo satisfactoriamente los objetivos del proyecto.

2. Objetivos

2.1. Objetivos General

- Crear un sistema inteligente que permita abrir y cerrar una puerta por medio de un Keypad y voz.

2.2. Objetivos Específicos

- Implementar el uso de TinyML con Edge Impulse para grabar las palabras claves.
- Encender LEDs para indicar el estado de la puerta (abierta o cerrada).
-

3. Alcances

Las limitaciones de este proyecto fueron las siguientes:

- Los materiales disponibles para la construcción de la puerta no fueron los mejores, son muy pesados y no tan portátiles en transporte público.
- Haber tenido una pequeña noción de la existencia de Edge Impulse al inicio del curso hubiese sido de gran ayuda ya que el proyecto estaba para más aplicaciones como el reconocimiento facial.
- Realizar un entrenamiento más estricto, esto incluye cantidad de muestras, diferentes voces en distintas horas del día.
- El costo total es bien alto debido al uso de dos placas, esto pudo haberse evitado con el hecho de haber revisado el funcionamiento del Arduino Nano Ble 33 Sense con más anticipación.

4. Nota teórica

Arduino Nano 33 Ble Sense

El arduino Nano 33 BLE Sense es un módulo miniatura que contiene un módulo NINA B306, basado en Nordic nRF52480 y contiene un M4F Cortex, un cripto chip el cual puede almacenar certificados de forma segura y pre-compartir llaves y un IMU de 9 ejes. El módulo puede ser montado como un componente DIP o como componente SMT, directamente soldado por la vía de los pads. Es importante mencionar que la parte de TinyML forma parte de este proyecto, ya que implica reconocimiento de voz, por tanto, hay entrenamiento previo con la grabación de las acciones que se desean realizar, es decir, abrir y cerrar más dos complementos adicionales como sonidos desconocidos y ruido esto porque un modelo de ML no tiene idea sobre las cosas que están bien o mal, y solo aprende con los datos que se le dan, en ese sentido, entre más variado sean los datos, el modelo trabajará mejor [5]. Ahora, también es importante mencionar que la librería PDM y el sensor MP34DT05 ayudaron a la captura de las muestras esto porque los micrófonos se encargan de hacer la conversión de de sonidos en datos digitales.

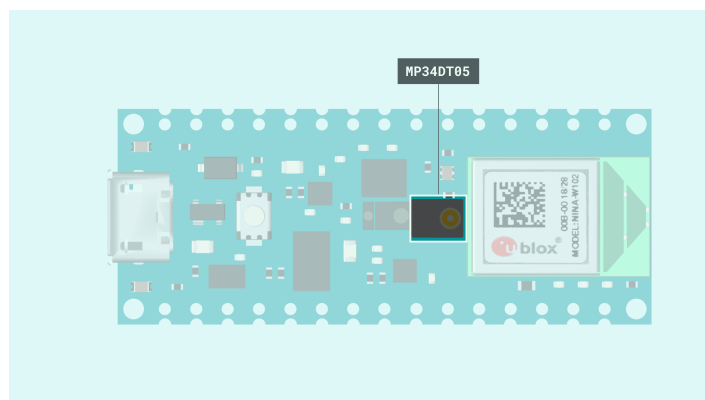


Figura 1: Ubicación del micrófono. Tomado de [1].

Características generales

Las características más importantes de este mcu se mencionan a continuación [2]

- CPU: ARM Cortex-M4 a 64MHz con FPU, 32-bit, 1MB Flash, 256kB SRAM.
- Bluetooth 5, IEEE 802.15.4-2006, 2,4 GHz.
- ARM TrustZone Cryptocell 310 security subsystem, secure boot.
- USB 2.0, QSPI, SPI.
- 48 GPIOs.
- 12-bit, ADC con 8 canales.
- 64 comparadores de nivel, 15 del tipo low-power.
- Sensor de temperatura.
- 4 × 4-canales PWM.
- Periféricos de audio: I2S, PDM
- 5 × 32-bit timers.
- 4× SPI maestros 3× SPI esclavos.
- 2× I2C.
- 2× UART.
- decodificador de cuadratura (QDEC).
- 3× RTC.

Diagrama de bloques

La figura 2 representa el diagrama de bloques de la placa.

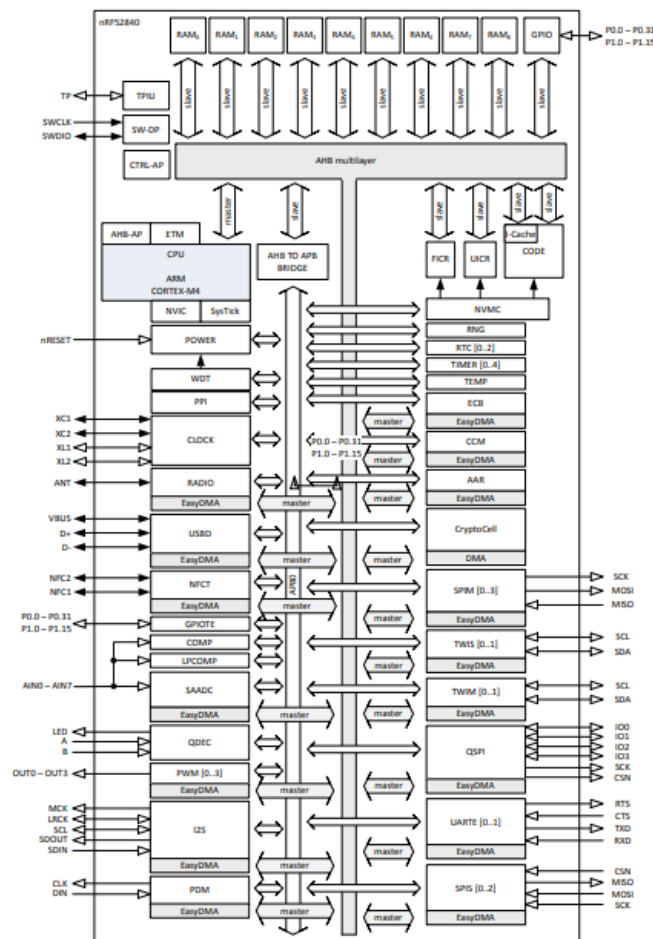


Diagrama de pines

El diagrama de la figura 3 brinda de manera más detallada la distribución de los pines.

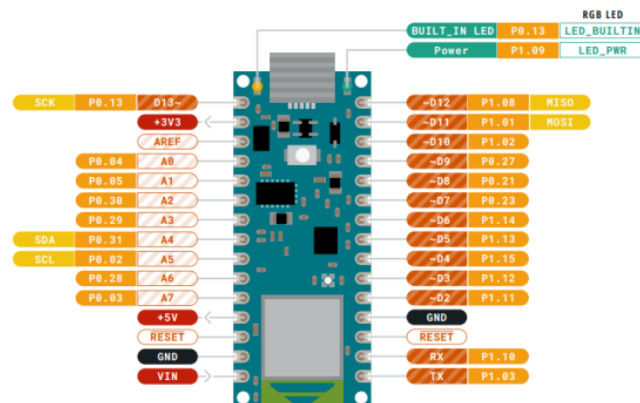


Figura 3: Diagrama de pines del Nano BLE 33 Sense . Tomado de [3].

Características eléctricas

Aquí se tomaron dos referencias para tener más claro este detalle, primero se muestran los valores máximos del mcu nRF52480 y los de la placa respectivamente.

Note	Min.	Max.	Unit
Supply voltages			
VDD	-0.3	+3.9	V
VDDH	-0.3	+5.8	V
VBUS	-0.3	+5.8	V
VSS		0	V
I/O pin voltage			
V_{IO} , VDD ≤ 3.6 V	-0.3	VDD + 0.3	V
V_{IO} , VDD > 3.6 V	-0.3	3.9	V
NFC antenna pin current			
$I_{HRC1/2}$		80	mA
Radio			
RF input level		10	dBm
Environmental aQFN package			
Storage temperature	-40	+125	°C
MSL	Moisture Sensitivity Level	2	
ESD HBM	Human Body Model	2	kV
ESD HBM Class	Human Body Model Class	2	
ESD CDM	Charged Device Model	750	V
Environmental WLCSP 3.544 x 3.607 mm package			
Storage temperature	-40	+125	°C
MSL	Moisture Sensitivity Level	1	
ESD HBM	Human Body Model	1	kV
ESD HBM Class	Human Body Model Class	1C	
ESD CDM	Charged Device Model	500	V
Flash memory			
Endurance	10 000		Write/erase cycles
Retention	10 years at 40°C		

Figura 4: Características eléctricas de nRF52480. Tomado de [2].

1.1.1 Recommended Operating Conditions

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

1.2 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

Figura 5: Características eléctricas de la placa. Tomado de [3].

Arduino Mega 2560 Rev3

El verdadero MCU utilizado es el elegoo MEGA 2560 R3, sin embargo comparte las mismas características que el arduino, la documentación de esta última está más depurada es por esta razón que fue la que se tomó como base.

4.0.1. Características generales

Las características generales de este MCU se describe a continuación [4]:

- Rendimiento de hasta 16 MIPS a 16 MHz
- 256k bytes (de los cuales 8k se utilizan para el cargador de arranque)
- 4k bytes EEPROM
- 8k bytes de SRAM interna
- 32×8 registros de trabajo de propósito general
- Contador en tiempo real con oscilador separado
- Cuatro canales PWM de 8 bits
- Cuatro USART serie programables
- Interfaz serie SPI controladora/periférica

4.0.2. Diagrama de bloques

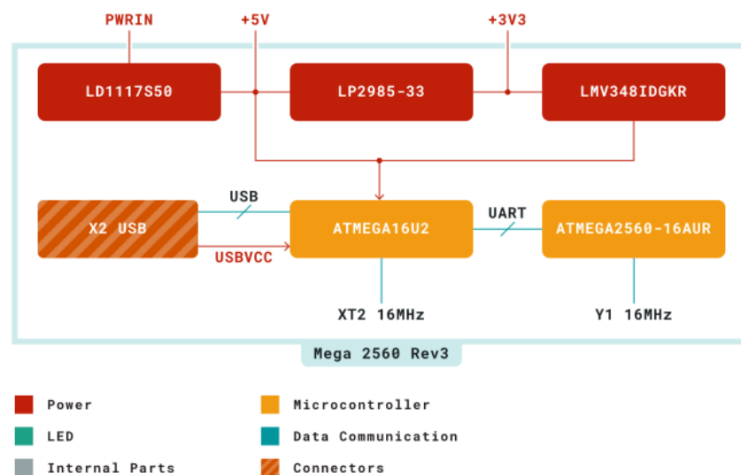


Figura 6: Diagrama de bloques del arduino mega 2560 REV3 [4].

4.0.3. Diagrama de pines

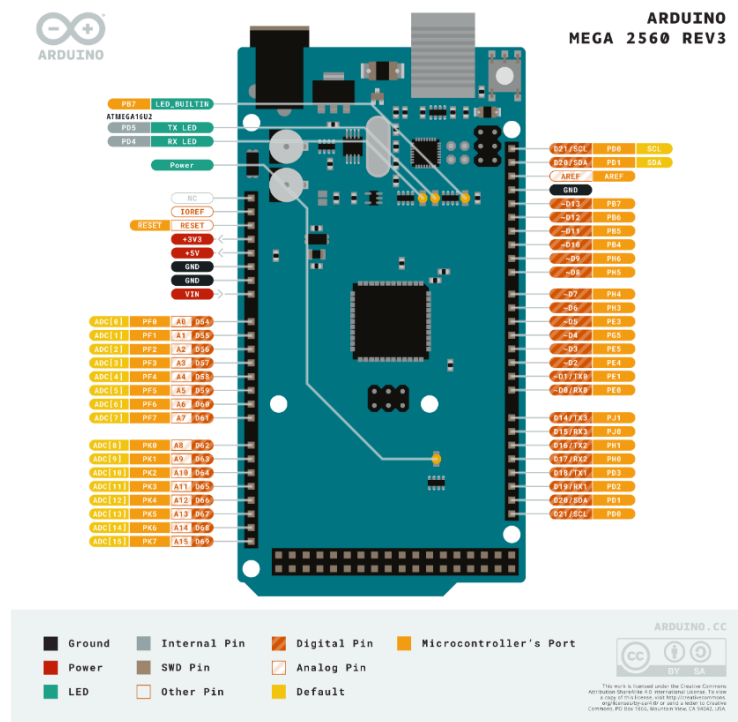


Figura 7: Diagrama de pines del arduino mega 2560 REV3 [4].

4.0.4. Características eléctricas

Symbol	Description	Min	Typ	Max	Unit
V_{IN}	Input voltage from VIN pad / DC Jack	7	7.0	12	V
V_{USB}	Input voltage from USB connector	4.8	5.0	5.5	V
T_{OP}	Operating Temperature	-40	25	85	°C

Figura 8: Condiciones de operación del arduino mega 2560 REV3 [4].

Periféricos utilizados

Los principales periféricos para este proyecto fueron el módulo I2C y el sensor MP34DT05.

La idea de usar este módulo es incorporarlo con la pantalla lcd porque facilita la conexión con el arduino Mega2560 por lo que solo basta con conectar los pines GND, VCC, SDA y SCL de la figura 10, donde los dos últimos tienen que ver los datos seriales y el reloj. Donde una condición de parada: alto-bajo corresponde a la transición SDA (I/O) mientras que SCL está en alto y es enviada por el master. Otro detalle es que estos pines necesitan resistencias de pull-up apropiadas y tomar en consideración la capacitancia total de todos los esclavos del bus I²C. Los dos primeros pines se trata de una conexión básica, fuente a tierra y la alimentación del módulo que esto va conectado con los 5 V del arduino Mega2560 [6].

Por otro lado, el sensor MP34DT05 es un micrófono ultra-compacto que usa PDM (modulación densidad de pulso) para representar una señal analógica como una señal binaria. El rango del sensor posee diferentes valores:

- Radio señal de ruido: 64 dB.
- Sensibilidad: $-26\text{dBFS} \pm 3\text{ dB}$

- Rango de temperatura: -40 a 85 °C

Componentes electrónicos complementarios

En función de cumplir con los objetivos del proyecto los siguientes componentes fueron de gran ayuda para cumplirlos.

Keypad

La función del Keypad es para escribir la clave que se mostrará en la pantalla lcd.



Figura 9: Keypad

Módulo I2C

Este componente es sumamente importante porque se incorpora a la pantalla lcd para establecer la comunicación con el microcontrolador y luego con el keypad.

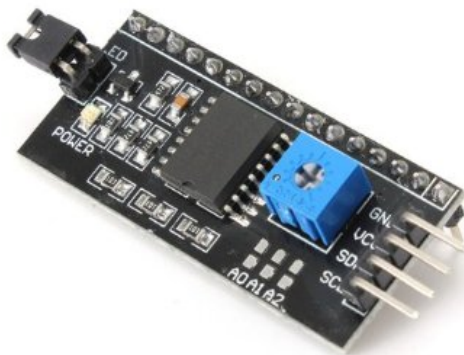


Figura 10: Módulo I2C.

Pantalla lcd

Su tarea es mostrar la entrada del usuario, si escribe la contraseña correcta o incorrecta se mostrará un mensaje en la pantalla.



Figura 11: Pantalla lcd.

Servomotor

Este componente se encarga de abrir o cerrar la puerta dependiendo de dos aspectos. El primero es con la clave escrita en el keypad y lo segundo con base a las palabras claves: *abrir* o *cierra*.



Figura 12: Servomotor.

Ahora, por medio de muchas pruebas realizadas se determinó que no fue posible trabajar solo con el Arduino Nano 33 Ble Sense ya que se determinó en el laboratorio que el pin de 5 V no funcionaba porque mostraba una salida de 5 V. El otro problema que se mostró fue a la hora de hacer la conexión con el keypad porque a la hora de presionar cualquier botón por medio del monitor serial se estaban reconociendo caracteres completamente desconocidos, ante esto se diseñó un filtro RC en la salida, esto para las pulsaciones, sin embargo, el problema persistía. El otro problema es con la pantalla lcd, ella necesita una alimentación de al menos 5 V, en ese sentido debe estar conectada a un pin que posee esa salida, de lo contrario muestra un bajo contraste y no se puede apreciar correctamente el contenido, para solucionar este problema pudo haberse hecho un par Darlington o realizar lo mostrado en la figura 13.

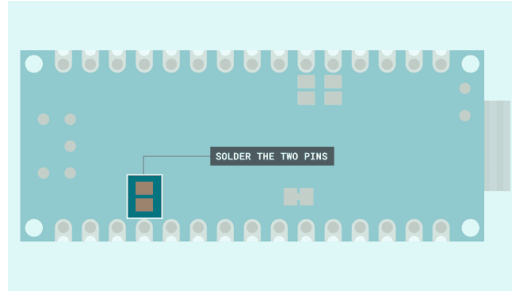


Figura 13: Habilitación de 5 V.

No obstante, por cuestiones de tiempo y complejidad en ese tipo de soldadura especial/fina no se tomaron esos caminos. Ante todos estas situaciones lo más favorable fue hacer uso de otra placa para realizar estas conexiones. Entonces, el proyecto tiene dos etapas:

- La placa Arduino MEGA2560, se encarga de abrir y cerrar la puerta por medio del keypad cuando se le escribe la contraseña correcta y controlar el display.
- La placa Arduino Nano 33 Ble Sense fue entrenada con 4 clases; dos keywords (abrir/cerrar) y otras dos con ruido y sonidos desconocidos. Así, cuando reconozca alguna de las palabras claves se abrirá o cerrará la puerta y encenderá un LED dependiendo del estado.

Tabla 1: Lista de equipos

Componente	Cantidad	Precio
Arduino Nano 33 Ble Sense	1	60\$
Arduino Mega2560	1	18\$
Keypad	1	9\$
Pantalla lcd	1	6.95\$
Servomotor	2	11\$
Batería 9 V	1	5.5\$
Kit de resistencias	1	0.2\$
Kit de jumpers	1	2 \$
Módulo I2C	1	4\$
Total		117\$

Diseño del circuito

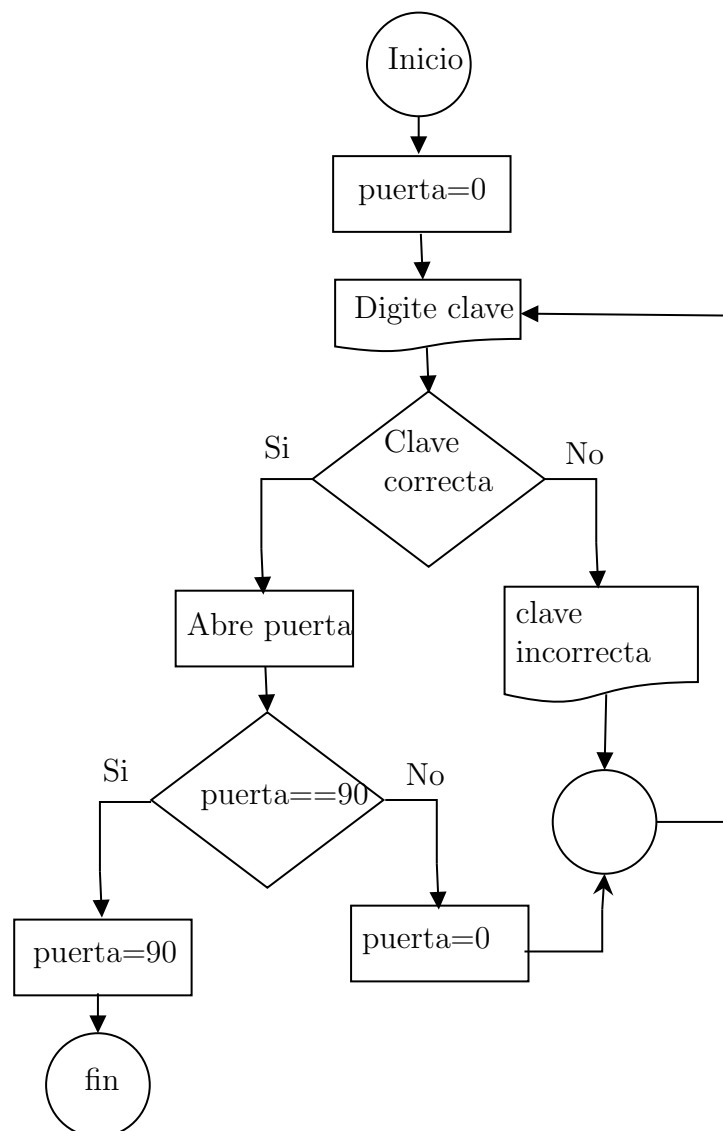


Figura 14: Diagrama de bloques smart door lock.

El diagrama mostrado anteriormente es la lógica que posee el proyecto en general. Gracias a este esquema, desde un inicio se tuvo claro los objetivos a cumplir.

5. Desarrollo/Análisis

Inicialmente se hicieron todas las configuraciones necesarios con las referencias[5] dadas en clase. Hecho esto se logró sincronizar el microcontrolador con la página Edge Impulse. Esto permitió hacer uso del micrófono del Arduino Nano Ble 33 Sense y así comenzar con las grabaciones.

Dataset


Data explorer

Data sources

CSV Wizard


DATA COLLECTED


4m 0s






TRAIN / TEST SPLIT

90% / 10%









Dataset



Training (216)

Test (24)













SAMPLE NAME	LABEL	ADDED	LENGTH	
noise.orig_test.AirConditioner_10....	noise	Feb 12 2024, 10:07:...	1s	
noise.orig_test.AirConditioner_10....	noise	Feb 12 2024, 10:07:...	1s	
noise.orig_test.AirConditioner_9.w...	noise	Feb 12 2024, 10:07:...	1s	
noise.orig_test.AirConditioner_9.w...	noise	Feb 12 2024, 10:07:...	1s	
noise.orig_test.AirConditioner_9.w...	noise	Feb 12 2024, 10:07:...	1s	
noise.orig_test.AirConditioner_9.w...	noise	Feb 12 2024, 10:07:...	1s	
noise.orig_test.AirConditioner_8.w...	noise	Feb 12 2024, 10:07:...	1s	
noise.orig_test.AirConditioner_8.w...	noise	Feb 12 2024, 10:07:...	1s	
noise.orig_test.AirConditioner_8.w...	noise	Feb 12 2024, 10:07:...	1s	
noise.orig_test.AirConditioner_8.w...	noise	Feb 12 2024, 10:07:...	1s	

Figura 15: Conjunto de datos.

En total de 240 muestras clasificadas por etiquetas.

- OPEN_TAG.

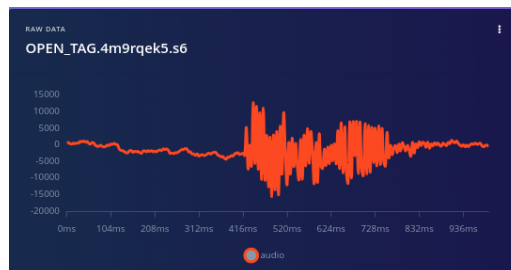


Figura 16: Espectro señal abrir.

- CLOSE_TAG.

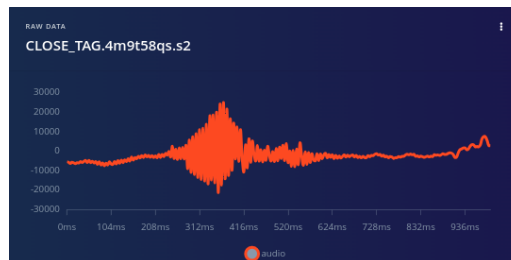


Figura 17: Espectro señal cierra.

- NOISE.

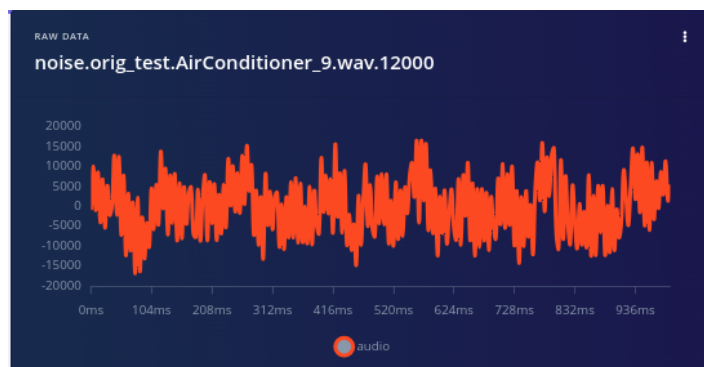


Figura 18: Espectro del ruido.

■ UNKNOWN.

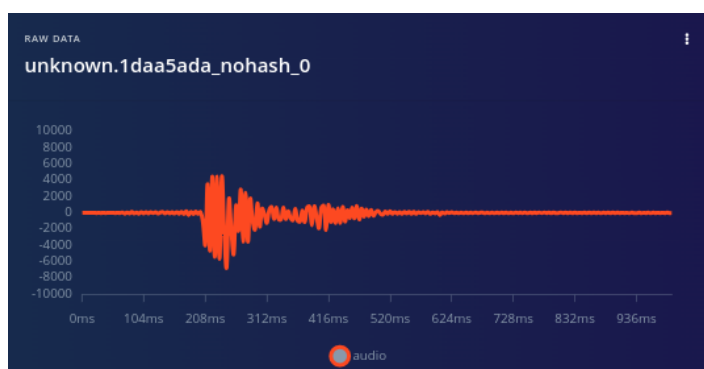


Figura 19: Espectro de sonido desconocido.

La misma plataforma ayudó a separar las grabaciones de 20 s en 12 muestras. Lo que hace más rápido y sencillo el procesamiento de datos en las 2 palabras clave. Ahora, las dos etiquetas restantes fueron proporcionadas por Edge-impulse.

Luego, con todos los datos listos en [7] es posible realizar la clasificación de los datos.

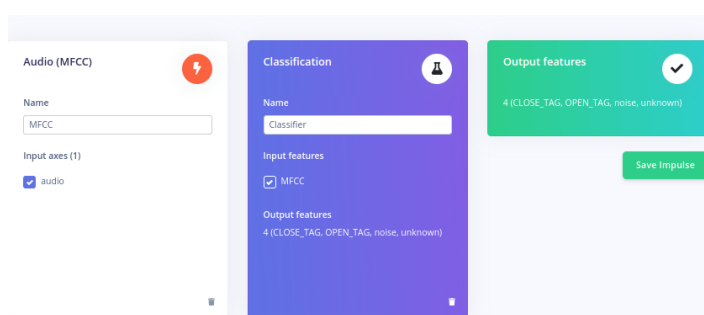


Figura 20: Clasificación de los datos.

De donde se usa una técnica de extracción de parámetros: MFCC (Mel-Frequency Cepstral Coefficients), se definen como coeficientes cepstrales¹ que se aplican sobre una ventana de tiempo de la señal de voz. Desde el punto de vista matemático es un operador que transforma una convolución en el tiempo en una suma espectral y de esa manera, se logran extraer los dos componentes en una señal de voz: la excitación y el tracto vocal [8].

Una vez clasificados los datos, los resultados que se obtuvieron fueron los siguientes:

¹se definen como la transformada inversa de Fourier del logaritmo del espectro de la señal de voz.

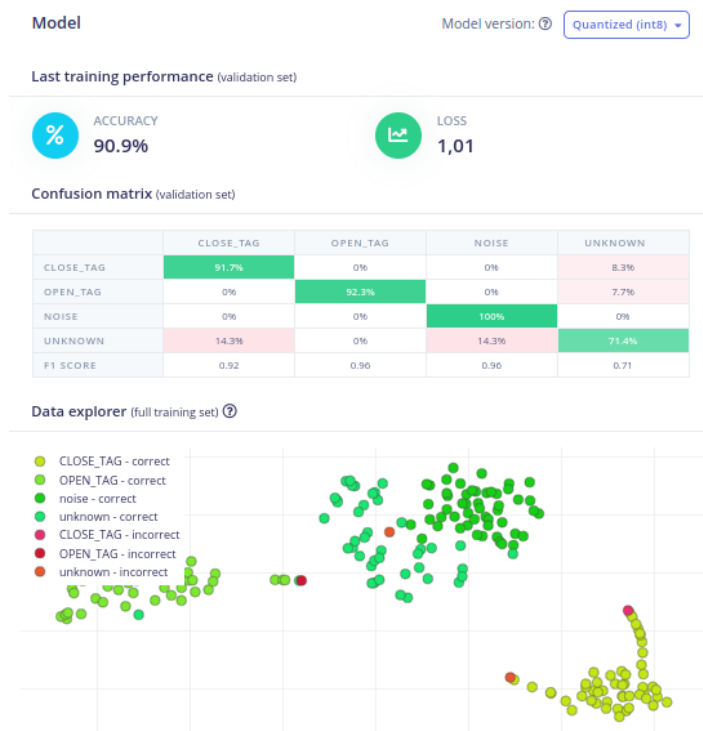


Figura 21: Resultado fina del entrenamiento.

Por último, lo que se hizo fue configurar el **deployment** de todo lo anterior para importarlo a una librería de Arduino. Ya con esto es posible instalarla en el IDE y así editar el código conforme a las necesidades requeridas de acorde a los objetivos plantados. Es decir, que logre reconocer la palabra: **abrir** y **cierra** para activar el servomotor y así abrir o cerrar el pica-portal de una puerta, donde habrán unos LEDs que mostrarán el estado del diseño implementado.

Una de las primeras pruebas fue editar un poco el código generado y ver como se comportaba cuando escuchaba una de las palabras claves.



Figura 22: Reconocimiento de la palabra abrir.



Figura 23: Reconocimiento de la palabra cierra.

Al decir la palabra abrir y cierra el LED del mcu muestra un color verde y azul respectivamente. También muestra un color rojo al inicio pero esto tiene que ver con la lógica que se tenía implementada.

Con esta comprobación fue posible hacer uso del servo motor, implementarle cierta lógica para que éste gire de 0 a 90 grados y viceversa con base al estado en el que se encuentre.

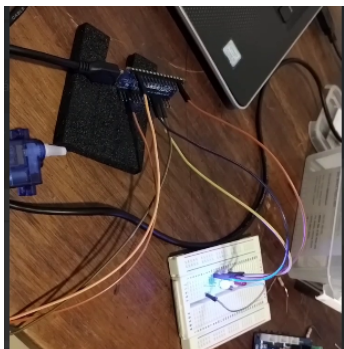


Figura 24: Accionamiento del servo con la palabra abrir.

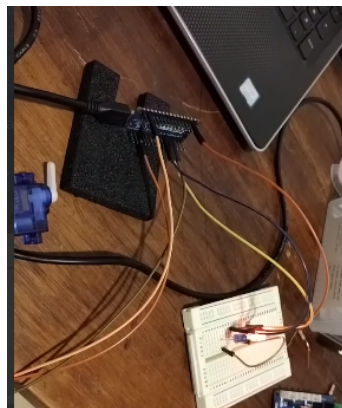


Figura 25: Accionamiento del servo con la palabra cierra.

Note que:

- 0 grados: puerta cerrada.
- 90 grados: puerta abierta.

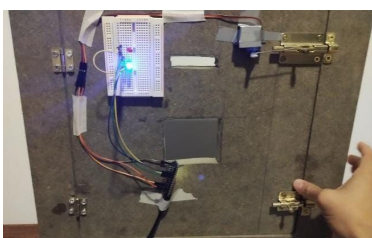


Figura 26: Funcionamiento completo con la palabra abrir.



Figura 27: Funcionamiento completo con la palabra cierra.

Las imágenes 26 y 27 muestran el comportamiento esperado.

Una vez realizado el reconocimiento de voz, se empezó probando todos los componentes electrónicos para verificar su funcionalidad y hacer ejemplos sencillos para entender como trabajar con ellos. La primera prueba realizada fue las del servo, en el que se utilizó la librería de "Servo.h" para poder manejar el mismo, es importante saber que no todos los servos tienen un mismo punto inicial, por lo que para este proyecto se tuvieron que tantear valores para poder trabajarlo de forma correcta.

Para el keypad se utiliza la librería de "Keypad.h" ya que, simplifica mucho el trabajo, simplemente es de crear una matriz que contenga los caracteres del keypad y hacer un mapeo correcto de los cables para que se detecte el carácter correcto dependiendo del botón que se pulse. Se tuvieron que hacer varias pruebas para determinar como estaba mapeado el keypad.

Para el lcd se decidió trabajar con un módulo de I2C, de no ser así la cantidad de cables pasaron de 8 a 4 y gracias a esto solo se tuvo que utilizar la librería de "LiquidCrystal_I2C.h" y se imprimen los mensajes deseados utilizando una comunicación serial mediante el puerto USB. Todo esto se puede observar en el siguiente enlace. <https://drive.google.com/file/d/1F8vAwWyKLxurYiivwDr2JWspp3p14MrM/view?usp=sharing> Una vez comprobado el funcionamiento de todo lo utilizado se procedió a ensamblar todo junto con la puerta, el resultado final se observa en las siguientes imagenes:



Figura 28: Cerradura final ensamblado parte frontal.



Figura 29: Cerradura final ensamblado parte trasera.

6. Conclusiones y recomendaciones

A modo de conclusión, es de gran satisfacción haber realizado un entrenamiento adecuado y así obtener los resultados esperados, es decir, que el microcontrolador logre procesar todas las señales para abrir o cerrar una puerta. Aunque algunas veces el sistema se acciona por si solo, y esto puede ser por ciertas razones, por ejemplo, la cantidad de muestras entrenadas, se esperaba hacer un procesamiento de datos más grandes pero Edge Impulse le costaba cada vez más incluso en repetidas ocasiones el algoritmo detrás de esto excedía el tiempo esperado por lo que terminaba desechar toda la grabación. Por tanto, hay que tener una paciencia (muy pequeña) y poder ver su funcionamiento. Este tipo de proyectos es de carácter muy enriquecedor porque implica mucha investigación, pruebas y errores, los cuales son enseñanzas muy buenas que ayudaron a avanzar en el desarrollo del proyecto. Es increíble la cantidad de proyectos de TinyML que se pueden hacer a partir de lo visto en el curso y lo bueno todo esto que hay muchos recursos para lograrlo.

Como recomendaciones:

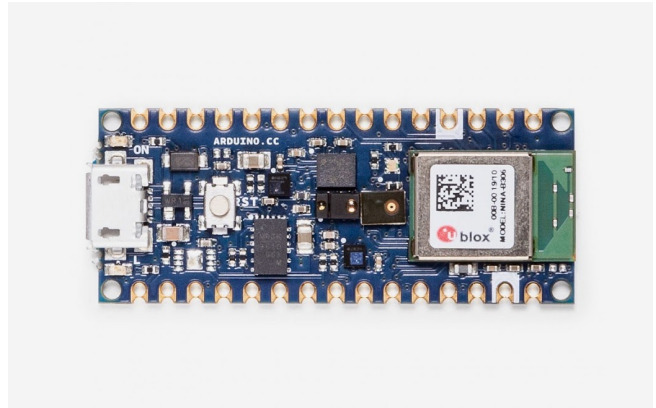
- Comprobar que el funcionamiento de los pines es el correcto, no asumir que todos están buenos y que te lleve a un cambio de planes en el diseño.
- Elegir un proyecto que sea realizable y que este de acuerdo a sus conocimientos. Esto por el poco tiempo que se dispone en el tercer ciclo.
- Tener mucho cuidado con el manejo de los componentes en todo momento porque causar algún daño en media planificación es muy doloroso por cuestiones de tiempo de más a invertir en la búsqueda del mismo.
- Disfrutar del proceso porque es una etapa de mucho aprendizaje cuando se emprenden nuevos conocimientos sin saber nada.

Referencias

- [1] Arduino Docs. Nano 33 ble sense cheat sheet. Arduino Docs, <https://docs.arduino.cc/tutorials/nano-33-ble-sense/cheat-sheet/>, febrero 2024. Accedido en febrero de 2024.
- [2] Nordic Infocenter. nrf52840 datasheet. Nordic Infocenter, https://infocenter.nordicsemi.com/pdf/nRF52840_PS_v1.1.pdf, Mayo 2019. Accedido en febrero de 2024.
- [3] Arduino Docs. Arduino nano 33 ble. Arduino Docs, <https://docs.arduino.cc/resources/datasheets/ABX00030-datasheet.pdf>, Mayo 2019. Accedido en febrero de 2024.
- [4] Arduino Docs. Mega 2560 rev3. Arduino Docs, <https://docs.arduino.cc/hardware/mega-2560/#tech-specs>, Febrero 2024. Accedido en febrero de 2024.
- [5] Edge Impulse. Responding to your voice. Edge impulse, <https://docs.edgeimpulse.com/docs/tutorials/end-to-end-tutorials/responding-to-your-voice>, noviembre 2023. Accedido en febrero de 2024.
- [6] Texas Instruments. Pcf8574 remote 8-bit. Texas Instruments, https://www.ti.com/lit/ds/symlink/pcf8574.pdf?ts=1638973422078&ref_url=https%253A%252F%252Fwww.ti.com%252Fproduct%252FPF8574%253Futm_source%253Dgoogle%2526utm_medium%253Dcpc%2526utm_campaign%253Dasc-int-null-prod%2526dynamic-cpc-pf-google-wwe%2526utm_content%253Dprod%2526dynamic%2526ds_k%253DDYNAMIC%2BSEARCH%2BAD%2526DCM%253Dyes%2526gclid%253D%2526gclid%253D, abril 2015. Accedido en febrero de 2024.
- [7] JSCC07183. Iparte-proyectoie0624. , <https://studio.edgeimpulse.com/studio/348181>, febrero 2024. Accedido en febrero de 2024.
- [8] Universidad de Sevilla. 3 extraccion de caracteristicas. Universidad de Sevilla, <https://biblus.us.es/bibing/proyectos/abreproy/12054/fichero/MEMORIA%252F8.Cap%C3%ADtulo+3.pdf>, junio 2017. Accedido en febrero de 2024.

7. Anexos

En esta sección se añade el link del repositorio: <https://github.com/JosueC07183/Proyecto-IE0624> y se muestran las hojas del fabricante de los componentes usados para este proyecto.



Description

Nano 33 BLE Sense is a miniature sized module containing a NINA B306 module, based on Nordic nRF52480 and containing a Cortex M4F, a crypto chip which can securely store certificates and pre shared keys 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, IoT application



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
- **LPS22HB** (Barometer and temperature sensor)
 - 260 to 1260 hPa absolute pressure range with 24 bit precision
 - High overpressure capability: 20x full-scale
 - Embedded temperature compensation
 - 16-bit temperature data output
 - 1 Hz to 75 Hz output data rate
 - Interrupt functions: Data Ready, FIFO flags, pressure thresholds
- **HTS221** (relative humidity sensor)
 - 0-100% relative humidity range
 - High rH sensitivity: 0.004% rH/LSB
 - Humidity accuracy: $\pm 3.5\%$ rH, 20 to +80% rH
 - Temperature accuracy: ± 0.5 °C, 15 to +40 °C
 - 16-bit humidity and temperature output data



- **APDS-9960** (Digital proximity, Ambient light, RGB and Gesture Sensor)
 - Ambient Light and RGB Color Sensing with UV and IR blocking filters
 - Very high sensitivity – Ideally suited for operation behind dark glass
 - Proximity Sensing with Ambient light rejection
 - Complex Gesture Sensing
- **MP34DT05** (Digital Microphone)
 - AOP = 122.5 dB SPL
 - 64 dB signal-to-noise ratio
 - Omnidirectional sensitivity
 - -26 dBFS \pm 3 dB sensitivity
- **ATECC608A** (Crypto Chip)
 - Cryptographic co-processor with secure hardware based key storage
 - Protected storage for up to 16 keys, certificates or data
 - ECDH: FIPS SP800-56A Elliptic Curve Diffie-Hellman
 - NIST standard P256 elliptic curve support
 - SHA-256 & HMAC hash including off-chip context save/restore
 - AES-128 encrypt/decrypt, galois field multiply for GCM
- **MPM3610** DC-DC
 - Regulates input voltage from up to 21V with a minimum of 65% efficiency @minimum load
 - More than 85% efficiency @12V



2.7.3 Color and ALS Detection

The Color and ALS detection feature provides red, green, blue and clear light intensity data. Each of the R, G, B, C channels have a UV and IR blocking filter and a dedicated data converter producing 16-bit data simultaneously. This architecture allows applications to accurately measure ambient light and sense color which enables devices to calculate color temperature and control display backlight.

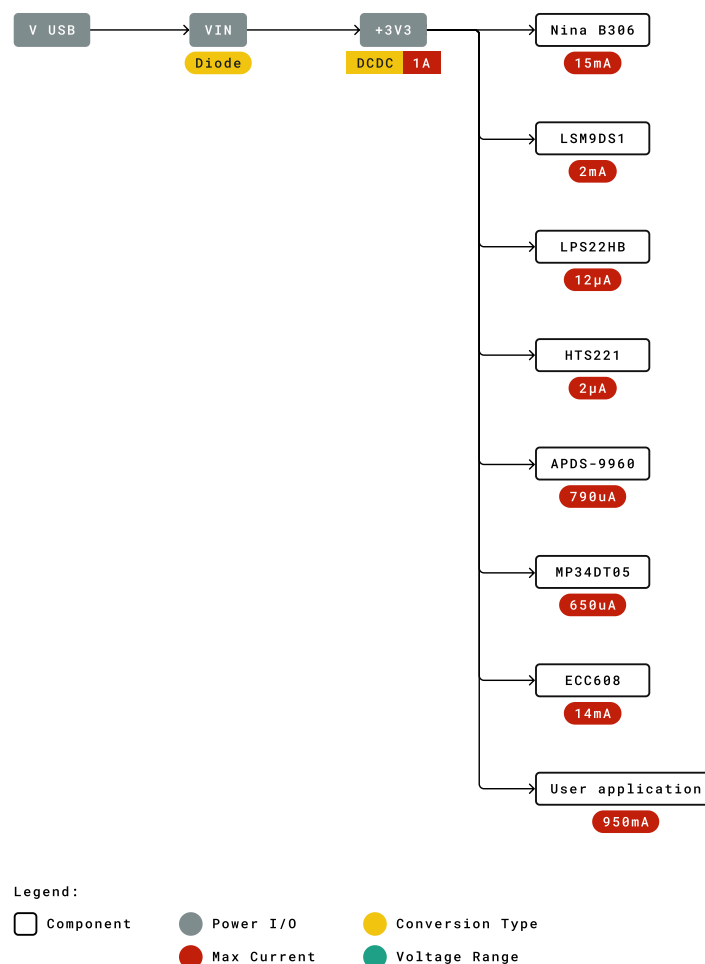
2.8 Digital Microphone

The MP34DT05 is an ultra-compact, low-power, omnidirectional, digital MEMS microphone built with a capacitive sensing element and an IC interface.

The sensing element, capable of detecting acoustic waves, is manufactured using a specialized silicon micromachining process dedicated to produce audio sensors

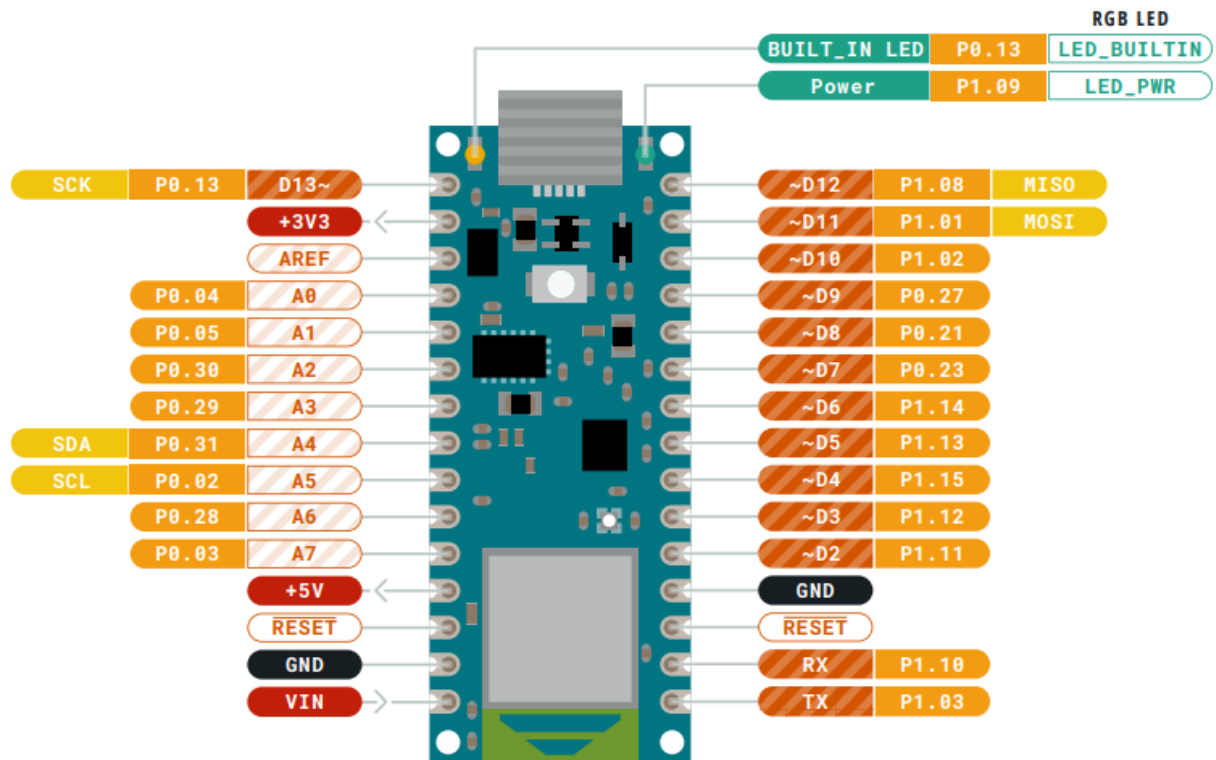
2.9 Power Tree

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



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.



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
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
13	RST	Digital In	Active low reset input (duplicate of pin 18)
14	GND	Power	Power Ground



Pin	Function	Type	Description
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

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

Features

- **ATmega2560 Processor**
 - Up to 16 MIPS Throughput at 16MHz
 - 256k bytes (of which 8k is used for the bootloader)
 - 4k bytes EEPROM
 - 8k bytes Internal SRAM
 - 32 × 8 General Purpose Working Registers
 - Real Time Counter with Separate Oscillator
 - Four 8-bit PWM Channels
 - Four Programmable Serial USART
 - Controller/Peripheral SPI Serial Interface
- **ATmega16U2**
 - Up to 16 MIPS Throughput at 16 MHz
 - 16k bytes ISP Flash Memory
 - 512 bytes EEPROM
 - 512 bytes SRAM
 - USART with SPI master only mode and hardware flow control (RTS/CTS)
 - Master/Slave SPI Serial Interface
- **Sleep Modes**
 - Idle
 - ADC Noise Reduction
 - Power-save
 - Power-down
 - Standby
 - Extended Standby
- **Power**
 - USB Connection
 - External AC/DC Adapter
- **I/O**
 - 54 Digital
 - 16 Analog
 - 15 PWM Output



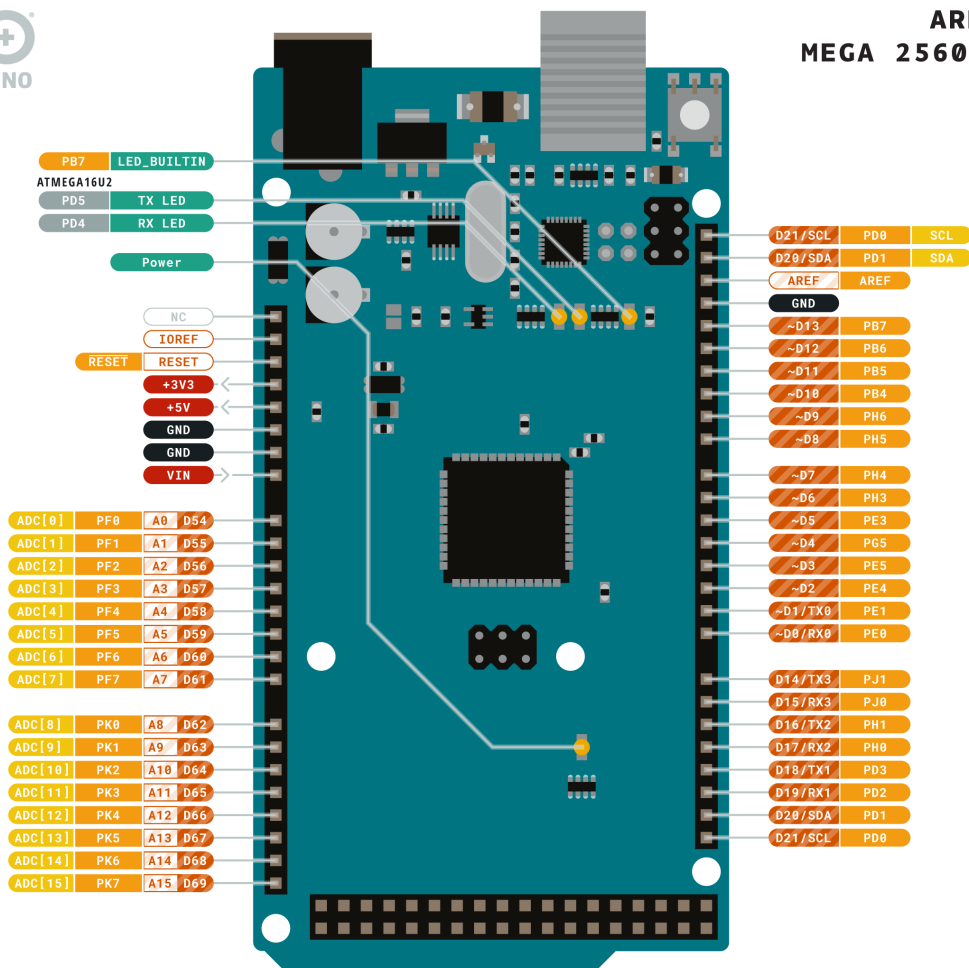
Contents

1 The Board	4
1.1 Application Examples	4
1.2 Accessories	4
1.3 Related Products	4
2 Ratings	5
2.1 Recommended Operating Conditions	5
3 Functional Overview	5
3.1 Block Diagram	5
3.2 Board Topology	6
3.3 Processor	7
3.4 Power Tree	7
4 Board Operation	8
4.1 Getting Started - IDE	8
4.2 Getting Started - Arduino Web Editor	8
4.3 Sample Sketches	8
4.4 Online Resources	8
5 Connector Pinouts	9
5.1 Analog	10
5.2 Digital	10
5.3 ATMEGA16U2 JP5	12
5.4 ATMEGA16U2 ICSP1	12
5.5 Digital Pins D22 - D53 LHS	12
5.6 Digital Pins D22 - D53 RHS	13
6 Mechanical Information	13
6.1 Board Outline	13
6.2 Board Mount Holes	14
7 Declaration of Conformity CE DoC (EU)	14
8 Declaration of Conformity to EU RoHS & REACH 211 01/19/2021	
9 Conflict Minerals Declaration	16
10 FCC Caution	16
11 Company Information	17
12 Reference Documentation	17
13 Revision History	17

5 Connector Pinouts



ARDUINO MEGA 2560 REV3



Ground	Internal Pin	Digital Pin	Microcontroller's Port
Power	SWD Pin	Analog Pin	
LED	Other Pin	Default	

ARDUINO.CC



This work is licensed under the Creative Commons Attribution-ShareAlike 4.0 International License. To view a copy of this license, visit <http://creativecommons.org/licenses/by-sa/4.0/> or send a letter to Creative Commons, PO Box 1866, Mountain View, CA 94042, USA.

Arduino Mega Pinout

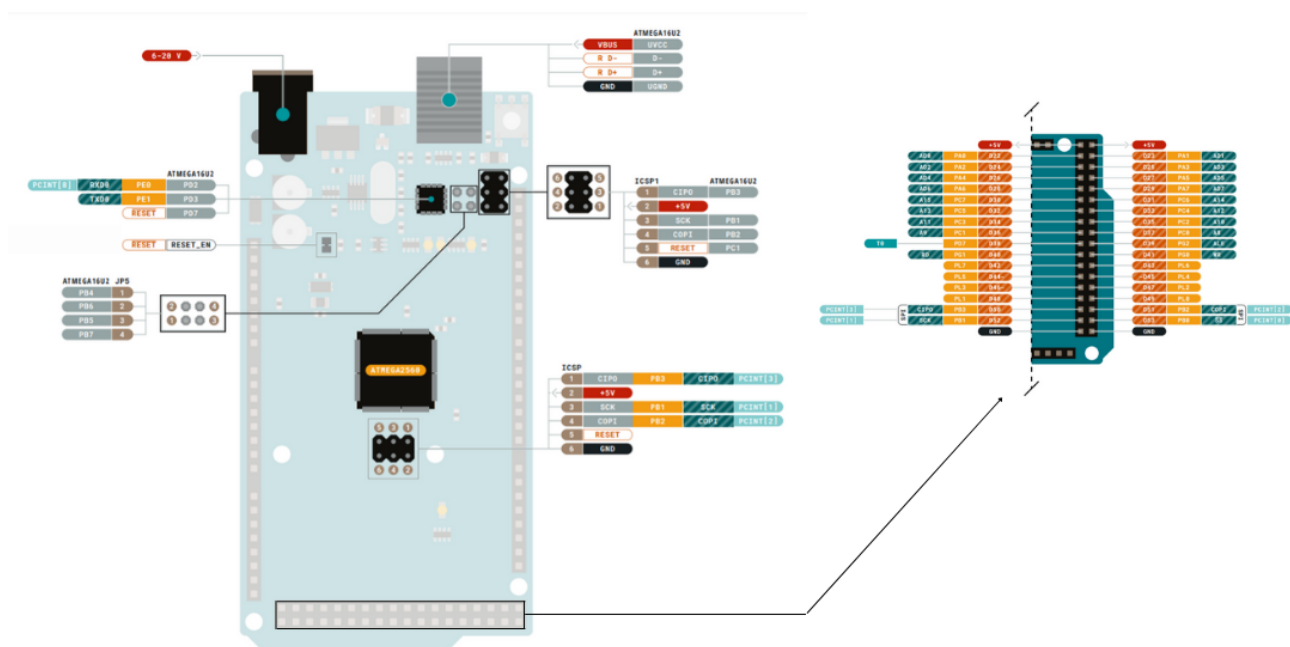
5.1 Analog

Pin	Function	Type	Description
1	NC	NC	Not Connected
2	IOREF	IOREF	Reference for digital logic V - connected to 5V
3	Reset	Reset	Reset
4	+3V3	Power	+3V3 Power Rail
5	+5V	Power	+5V Power Rail
6	GND	Power	Ground
7	GND	Power	Ground
8	VIN	Power	Voltage Input
9	A0	Analog	Analog input 0 /GPIO
10	A1	Analog	Analog input 1 /GPIO
11	A2	Analog	Analog input 2 /GPIO
12	A3	Analog	Analog input 3 /GPIO
13	A4	Analog	Analog input 4 /GPIO
14	A5	Analog	Analog input 5 /GPIO
15	A6	Analog	Analog input 6 /GPIO
16	A7	Analog	Analog input 7 /GPIO
17	A8	Analog	Analog input 8 /GPIO
18	A9	Analog	Analog input 9 /GPIO
19	A10	Analog	Analog input 10 /GPIO
20	A11	Analog	Analog input 11 /GPIO
21	A12	Analog	Analog input 12 /GPIO
22	A13	Analog	Analog input 13 /GPIO
23	A14	Analog	Analog input 14 /GPIO
24	A15	Analog	Analog input 15 /GPIO

5.2 Digital

Pin	Function	Type	Description
1	D21/SCL	Digital Input/I2C	Digital input 21/I2C Dataline
2	D20/SDA	Digital Input/I2C	Digital input 20/I2C Dataline
3	AREF	Digital	Analog Reference Voltage
4	GND	Power	Ground
5	D13	Digital/GPIO	Digital input 13/GPIO
6	D12	Digital/GPIO	Digital input 12/GPIO
7	D11	Digital/GPIO	Digital input 11/GPIO
8	D10	Digital/GPIO	Digital input 10/GPIO
9	D9	Digital/GPIO	Digital input 9/GPIO
10	D8	Digital/GPIO	Digital input 8/GPIO
11	D7	Digital/GPIO	Digital input 7/GPIO
12	D6	Digital/GPIO	Digital input 6/GPIO
13	D5	Digital/GPIO	Digital input 5/GPIO
14	D4	Digital/GPIO	Digital input 4/GPIO

Pin	Function	Type	Description
15	D3	Digital/GPIO	Digital input 3 /GPIO
16	D2	Digital/GPIO	Digital input 2 /GPIO
17	D1/TX0	Digital/GPIO	Digital input 1 /GPIO
18	D0/Tx1	Digital/GPIO	Digital input 0 /GPIO
19	D14	Digital/GPIO	Digital input 14 /GPIO
20	D15	Digital/GPIO	Digital input 15 /GPIO
21	D16	Digital/GPIO	Digital input 16 /GPIO
22	D17	Digital/GPIO	Digital input 17 /GPIO
23	D18	Digital/GPIO	Digital input 18 /GPIO
24	D19	Digital/GPIO	Digital input 19 /GPIO
25	D20	Digital/GPIO	Digital input 20 /GPIO
26	D21	Digital/GPIO	Digital input 21 /GPIO





User Guide

I2C Serial Interface 1602 LCD Module

This is I2C interface 16x2 LCD display module, a high-quality 2 line 16 character LCD module with on-board contrast control adjustment, backlight and I2C communication interface. For Arduino beginners, no more cumbersome and complex LCD driver circuit connection. The real significance advantages of this I2C Serial LCD module will simplify the circuit connection, save some I/O pins on Arduino board, simplified firmware development with widely available Arduino library.



SKU: [DSP-1182](#)

Brief Data:

- Compatible with Arduino Board or other controller board with I2C bus.
- Display Type: Negative white on Blue backlight.
- I2C Address: 0x38-0x3F (0x3F default)
- Supply voltage: 5V
- Interface: I2C to 4bits LCD data and control lines.
- Contrast Adjustment: built-in Potentiometer.
- Backlight Control: Firmware or jumper wire.
- Board Size: 80x36 mm.

Setting Up:

Hitachi's HD44780 based character LCD are very cheap and widely available, and is an essential part for any project that displays information. Using the LCD piggy-back board, desired data can be displayed on the LCD through the I2C bus. In principle, such backpacks are built around PCF8574 (from NXP) which is a general purpose bidirectional 8 bit I/O port expander that uses the I2C protocol. The PCF8574 is a silicon CMOS circuit provides general purpose remote I/O expansion (an 8-bit quasi-bidirectional) for most microcontroller families via the two-line bidirectional bus (I2C-bus). Note that most piggy-back modules are centered around PCF8574T (SO16 package of PCF8574 in DIP16 package) with a default slave address of 0x27. If your piggy-back board holds a PCF8574AT chip, then the default slave address will change to 0x3F. In short, if the piggy-back board is based on PCF8574T and the address connections (A0-A1-A2) are not bridged with solder it will have the slave address 0x27.



Address selection pads in the I2C-to-LCD piggy-back board.

Table 5. PCF8574A address map

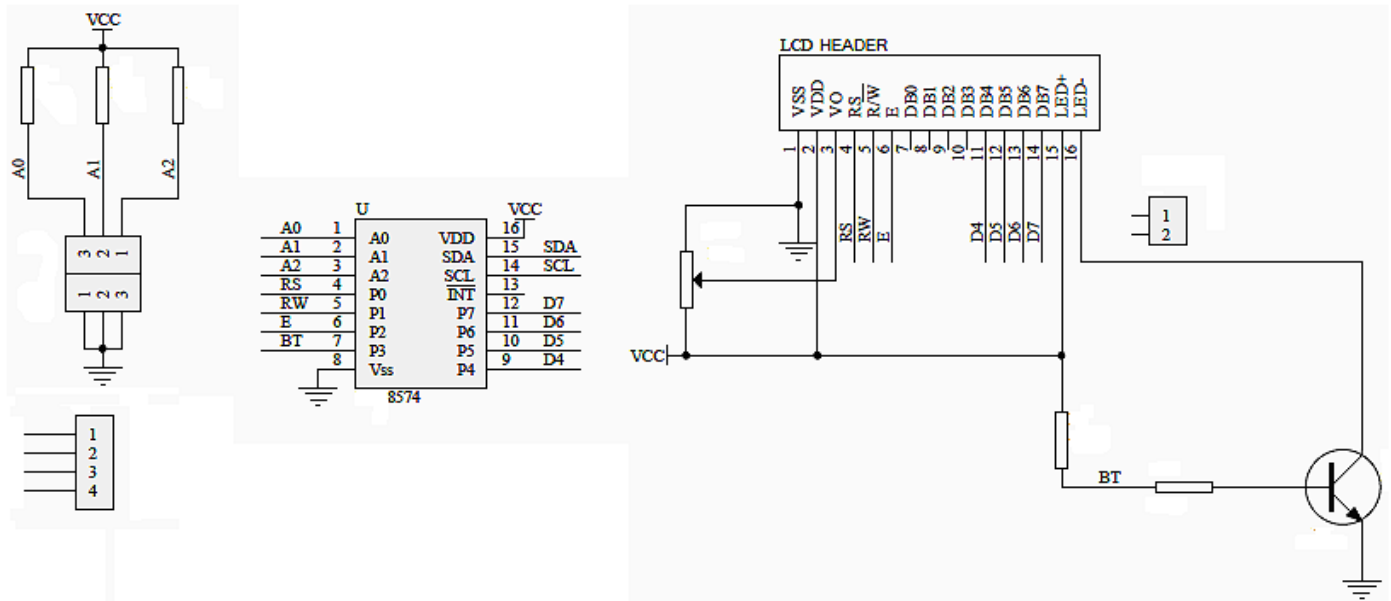
Pin connectivity			Address of PCF8574A								Address byte value		7-bit hexadecimal address without R/W
A2	A1	A0	A6	A5	A4	A3	A2	A1	A0	R/W	Write	Read	
V _{SS}	V _{SS}	V _{SS}	0	1	1	1	0	0	0	-	70h	71h	38h
V _{SS}	V _{SS}	V _{DD}	0	1	1	1	0	0	1	-	72h	73h	39h
V _{SS}	V _{DD}	V _{SS}	0	1	1	1	0	1	0	-	74h	75h	3Ah
V _{SS}	V _{DD}	V _{DD}	0	1	1	1	0	1	1	-	76h	77h	3Bh
V _{DD}	V _{SS}	V _{SS}	0	1	1	1	1	0	0	-	78h	79h	3Ch
V _{DD}	V _{SS}	V _{DD}	0	1	1	1	1	0	1	-	7Ah	7Bh	3Dh
V _{DD}	V _{DD}	V _{SS}	0	1	1	1	1	1	0	-	7Ch	7Dh	3Eh
V _{DD}	V _{DD}	V _{DD}	0	1	1	1	1	1	1	-	7Eh	7Fh	3Fh

Address Setting of PCD8574A (extract from PCF8574A data specs).

Note: When the pad A0~A2 is open, the pin is pull up to VDD. When the pin is solder shorted, it is pull down to VSS.

The default setting of this module is A0~A2 all open, so is pull up to VDD. The address is 3Fh in this case.

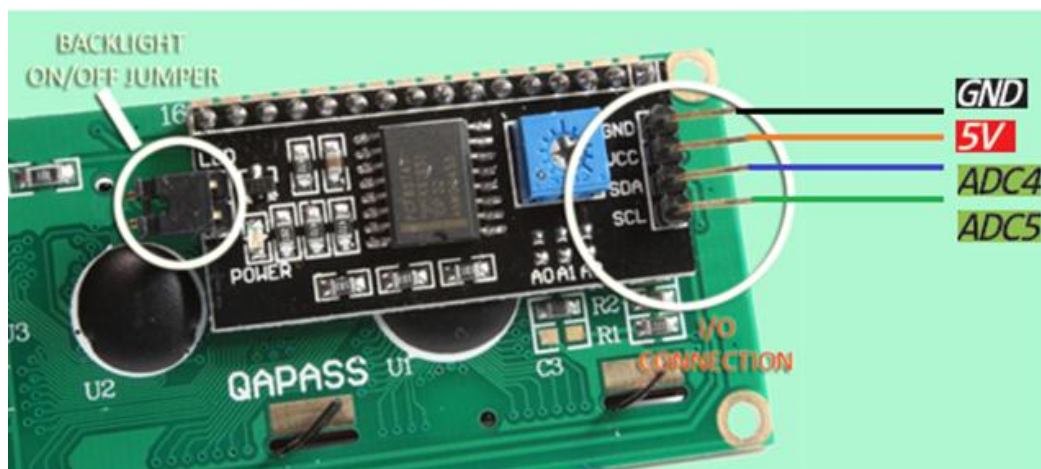
Reference circuit diagram of an Arduino-compatible LCD backpack is shown below. What follows next is information on how to use one of these inexpensive backpacks to interface with a microcontroller in ways it was exactly intended.



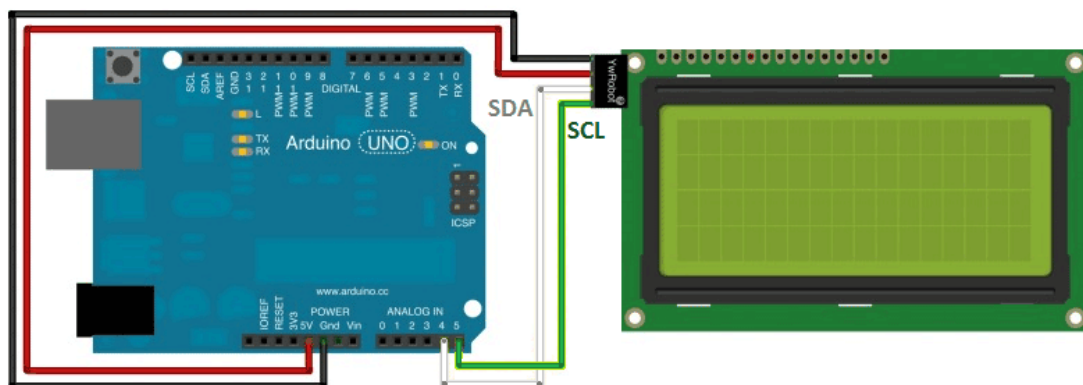
Reference circuit diagram of the I2C-to-LCD piggy-back board.

I2C LCD Display.

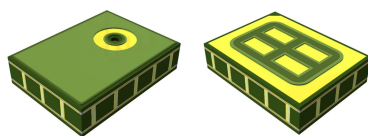
At first you need to solder the I2C-to-LCD piggy-back board to the 16-pins LCD module. Ensure that the I2C-to-LCD piggy-back board pins are straight and fit in the LCD module, then solder in the first pin while keeping the I2C-to-LCD piggy-back board in the same plane with the LCD module. Once you have finished the soldering work, get four jumper wires and connect the LCD module to your Arduino as per the instruction given below.



LCD display to Arduino wiring.



MEMS audio sensor omnidirectional digital microphone



HCLGA - 4LD (3 x 4 x 1 mm)

Features

- Single supply voltage
- Low power consumption
- AOP = 122.5 dB SPL
- 64 dB signal-to-noise ratio
- Omnidirectional sensitivity
- -26 dBFS ± 3 dB sensitivity
- PDM output
- HCLGA package
 - Top-port design
 - SMD-compliant
 - EMI-shielded
 - ECOPACK, RoHS, and “Green” compliant

Applications

- Mobile terminals
- Laptop and notebook computers
- Portable media players
- VoIP
- Speech recognition
- A/V eLearning devices
- Gaming and virtual reality input devices
- Digital still and video cameras
- Antitheft systems

Product status link	
MP34DT05-A	
Product summary	
Order code	MP34DT05TR-A
Temperature range [°C]	-40 to +85
Package	HCLGA (3 x 4 x 1 mm) 4LD
Packing	Tape and reel

Description

The **MP34DT05-A** is an ultra-compact, low-power, omnidirectional, digital MEMS microphone built with a capacitive sensing element and an IC interface.

The sensing element, capable of detecting acoustic waves, is manufactured using a specialized silicon micromachining process dedicated to producing audio sensors.

The IC interface is manufactured using a CMOS process that allows designing a dedicated circuit able to provide a digital signal externally in PDM format.

The MP34DT05-A is a low-distortion digital microphone with a 64 dB signal-to-noise ratio and -26 dBFS ± 3 dB sensitivity.

The MP34DT05-A is available in a top-port, SMD-compliant, EMI-shielded package and is guaranteed to operate over an extended temperature range from -40 °C to +85 °C.

2 Acoustic and electrical specifications

2.1 Acoustic and electrical characteristics

The values listed in the table below are specified for Vdd = 1.8 V, Clock = 2.4 MHz, T = 25 °C, unless otherwise noted.

Table 2. Acoustic and electrical characteristics

Symbol	Parameter	Test condition	Min.	Typ. ⁽¹⁾	Max.	Unit
Vdd	Supply voltage		1.6	1.8	3.6	V
Idd	Current consumption in normal mode	Mean value		650		μA
IddPdn	Current consumption in power-down mode ⁽²⁾				5	μA
Sc	Short-circuit current		1		10	mA
AOP	Acoustic overload point			122.5		dB SPL
So	Sensitivity		-29	-26	-23	dBFS
SNR	Signal-to-noise ratio	A-weighted @1 kHz, 94 dB SPL		64		dB(A)
PSR	Power supply rejection	100 mVpp sine 1 kHz			-90	dBFS
f _{CLK}	Input clock frequency ⁽³⁾		1.2	2.4	3.25	MHz
Ton	Turn-on time ⁽⁴⁾	Guaranteed by design			10	ms
Top	Operating temperature range		-40		+85	°C
V _{IOL}	Low-level logic input/output voltage	I _{out} = 1 mA	-0.3		0.35xVdd	V
V _{IOH}	High-level logic input/output voltage	I _{out} = 1 mA	0.65xVdd		Vdd+0.3	V
C _{LOAD}	Capacitive load				100	pF

1. Typical specifications are not guaranteed.
2. Input clock in static mode.
3. Duty cycle: min = 40% max = 60%.
4. Time from the first clock edge to valid output data.

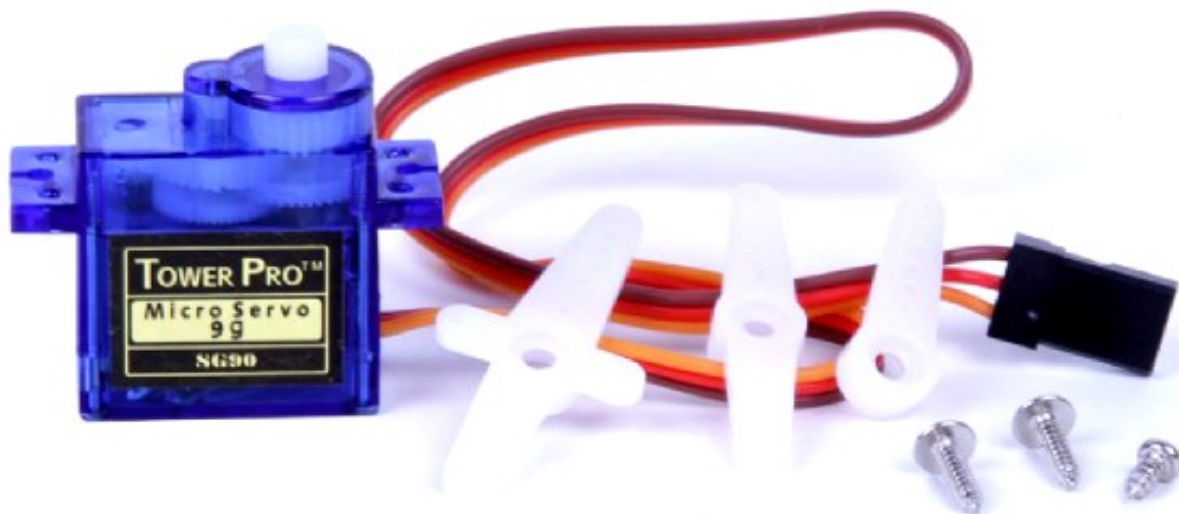
Table 3. Distortion specifications @ 1 kHz

Parameter	Test condition	Typical value ⁽¹⁾
Distortion	94 dB SPL	0.2% THD + N
Distortion	110 dB SPL	0.7% THD + N
Distortion	120 dBSP	6% THD + N

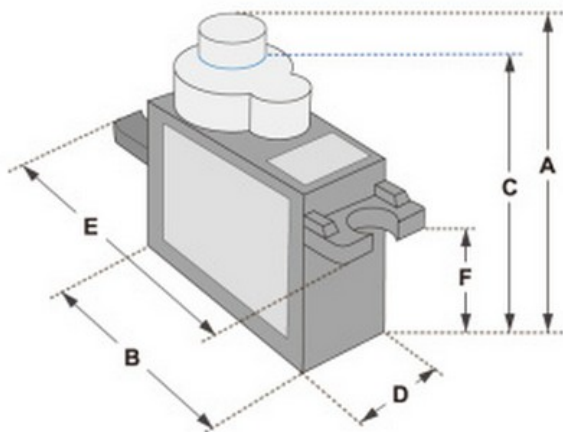
1. Typical specifications are not guaranteed.

SERVO MOTOR SG90

DATA SHEET



Tiny and lightweight with high output power. Servo can rotate approximately 180 degrees (90 in each direction), and works just like the standard kinds but smaller. You can use any servo code, hardware or library to control these servos. Good for beginners who want to make stuff move without building a motor controller with feedback & gear box, especially since it will fit in small places. It comes with a 3 horns (arms) and hardware.



Dimensions & Specifications
A (mm) : 32
B (mm) : 23
C (mm) : 28.5
D (mm) : 12
E (mm) : 32
F (mm) : 19.5
Speed (sec) : 0.1
Torque (kg-cm) : 2.5
Weight (g) : 14.7
Voltage : 4.8 - 6

Position "0" (1.5 ms pulse) is middle, "90" (~2ms pulse) is middle, is all the way to the right, "-90" (~1ms pulse) is all the way to the left.

PWM=Orange (⏏)
Vcc = Red (+)
Ground=Brown (-)

