



**Tecnológico  
de Monterrey**

---

*Campus Querétaro*

*Diseño de sistemas en chip  
TE2003B.1*

*Profesor:  
Agustín Domínguez Oviedo*

***Reproductor de  
Audio***

*Por:  
Marcela Ibarra Mora  
A01231973  
Mariana Castro Payns  
A01706038*

*Viernes 11 de junio de 2021*

## **Índice**

- I.    Introducción**
- II.   Descripción del reto.**
- III.   Estado del arte.**
- IV.   Diagrama eléctrico.**
- V.    Código fuente.**
- VI.   Explicación del programa y funcionamiento.**
- VII.   Fotos del circuito eléctrico.**
- VIII.  Video de funcionalidad.**
- IX.   Conclusiones.**

## **I. Introducción.**

Dentro del valor de producción de un automóvil, los sistemas eléctricos incluidos abarcan alrededor del 40%.

Estos sistemas contienen ya sea un microcontrolador o microprocesador lo que permite realizar funciones específicas dentro del sistema, como en sistemas de infoentretenimiento, control de bolsas de aire, control de frenos, control de inyección de combustible, control de emisiones, interfaz de instrumentos, entre muchos otros.

Este tipo de tecnología se desarrolla en compañías proveedoras de la industria, tales como Harman, Visteon, Continental, Delphi, Bosch, entre otras. Con el paso de los años, todos los autos o en su gran mayoría se manejan de forma autónoma, contando con cada vez más sistemas inteligentes, que permitirán incrementar la calidad de vida mediante la reducción de accidentes.

Los sistemas de infoentretenimiento representan una parte crucial de los automóviles modernos. Normalmente son dispositivos en un vehículo que concentran en una sola interfaz la mayoría de las funciones del vehículo relacionadas con los sistemas que proporcionan información, navegador, recomendaciones de conducción, enlace con teléfonos inteligentes para hacer llamadas y tener conectividad a redes sociales, y entretenimiento por medio reproducción de audio, vídeo e imágenes.

## **II. Descripción del reto.**

Dentro del desarrollo de este reto aplicaremos cada uno de los conocimientos y competencias adquiridas a lo largo del bloque, para realizar el diseño e implementación de un reproductor de música con la capacidad de reproducir música mediante el microcontrolador, las canciones serán almacenadas en una tarjeta SD, el sistema tendrá un LCD que mostrará los datos de las canciones que se están reproduciendo, además tendrá tres botones para dar play y pausa, rewind y forward.

## **III. Estado del arte.**

- **Proceso de digitalización**

La digitalización es un proceso en el cual se convierten señales analógicas a digitales, así como asegurar que estas no contengan ningún tipo de ruido que puedan afectar nuestra aplicación. El proceso de digitalización tiene las siguientes etapas:

Muestreo: Se toman muestras periódicas de la amplitud de la señal analógica. La frecuencia de muestreo es determinada por el número de muestra que se toma en un intervalo de tiempo. De esta etapa se obtiene una señal discreta

Cuantización: De las muestras discreta obtenida anteriormente se mide el nivel de voltaje en cada una y se le asigna un valor finito aproximado de amplitud dependiendo de un margen de niveles preseleccionados.

Codificación: Se le asigna un valor binario a cada nivel de voltaje, dejando a la señal como un tren de pulsos digitales (sucesión de ceros y unos).

Al realizar este proceso hay que estar conscientes de que se tendrán pérdidas y hay un margen de error. Si se incrementa la frecuencia de muestreo se pierde espacio de memoria, ya que se necesita espacio extra de almacenamiento de muestras. Por otro lado, si se aumenta el número de niveles de voltaje dentro de la cuantización, afecta el número de bits necesarios para la codificación.

- **Audio monoaural y el estéreo**

Audio monoaural, consiste en una sola señal siendo reproducida en la bocina y se necesita un solo micrófono para poder grabar una señal de audio monoaural. Por otro lado, en el caso de conectarse a múltiples bocinas, el sonido sería el mismo en cada una de estas. Este formato de audio se considera el básico para salida de audio y es económico debido a que no se necesita equipo adicional además de un micrófono y una bocina. Este formato es usado principalmente en telecomunicaciones como los teléfonos y la radio. Así mismo, una de las ventajas de este formato es que el sonido llega de manera equitativa a todos los oyentes.

Por otro lado el audio estéreo, consiste en dos señales siendo recibidas por dos micrófonos, uno a la derecha y otro a la izquierda, estas dos señales se guardan en el mismo archivo que puede ser del tipo MP3 o WAV. Al ser reproducidas da la sensación que el audio proviene de dos lugares diferentes, creando profundidad.

- **Compresión de audio**

La compresión es un proceso que consiste básicamente en reducir el rango dinámico de una señal de audio, el rango dinámico es la diferencia entre las partes más ruidosas y más silenciosas de una señal de audio, dicho proceso generalmente se da con la ayuda de un aparato denominado compresor, el compresor se encarga de analizar la ganancia de la señal de entrada y de acuerdo a ciertos parámetros fijados, se atenúan aquellas partes que superan el nivel o umbral denominado según la configuración deseada, en pocas palabras, los compresores atenúan las partes más ruidosas de una señal y mejoran el resultado para que las partes más silenciosas sean más evidentes. Existen diferentes tipos de compresión, por lo que existen una gran variedad de compresores empleados para diferentes tareas, algunos de estos tipos de compresión

son compresión VCA, compresión Opto, compresión FET o compresión de válvula. Los controles que se le dan para configurar un compresor son generalmente:

Umbral: Indica que tan fuerte debe ser la señal antes de que se aplique la compresión.

Relación: Cuánta compresión se aplica. Por ejemplo, si se establece una relación de compresión en 6:1, la señal de entrada tendrá que cruzar el umbral de 6 dB para que el nivel de salida aumente en 1 dB.

Ataque: Se refiere a la rapidez con la que el compresor comienza a funcionar.

Release: Para establecer cuánto tiempo después de que la señal descienda por debajo del umbral se detiene el compresor.

Knee: Establece cómo reacciona el compresor a las señales una vez que se supera el umbral, existen ajusten Hard Knee y Soft Knee, dependiendo de si la señal se sujeta de inmediato o si la señal supera el umbral.

Make-Up gain: Permite aumentar la señal comprimida, ya que la compresión a menudo atenúa la señal de manera significativa.

Salida: Aumenta o atenúa el nivel de señal de salida del compresor.

De igual forma, “la compresión se utiliza en muchos formatos de audio para reducir el tamaño de los datos de audio. La compresión reduce el número de bits almacenados por muestra y, por lo tanto, la velocidad en bits. Algunos formatos no utilizan compresión, pero la mayoría utilizan uno de los tipos básicos de compresión.” IBM (2020).

- **Características de los diferentes formatos existentes de audio**

- A. Sin compresión:**

1. **PCM:** (Pulse-Code Modulation), es una representación digital de señales de audio analógicas sin procesar. Los sonidos analógicos existen como formas de onda, y para convertir una forma de onda en bits digitales, el sonido se debe muestrear y grabar ciertos pulsos. Este formato es el más común utilizado en CD y DVD.

Existe un subtipo de PCM llamado Modulación de código de pulso lineal o LPM, donde las muestras se toman a intervalos lineales.

2. **AIFF:** (Audio Interchange File Format), es un tipo de formato de audio que es similar a como Microsoft e IBM desarrollaron wav para windows, este fue un formato desarrollado por Apple en 1988. Los archivos de este tipo pueden contener varios tipos de formato de audio, sin embargo, la mayoría de los archivos AIFF contienen audio sin comprimir en formato PCM. Estos archivos solo son contenedores para la codificación PCM.

3. **FLAC:** (Free Lossless Audio Codec), se convirtió en uno de los formatos sin pérdida más utilizados desde su introducción en el año 2001. La ventaja más grande de FLAC es que se puede comprimir un archivo fuente original hasta un 60 por ciento sin perder un solo bit de datos, también es un formato de audio libre de código abierto, por lo que no impone restricciones de propiedad intelectual. FLAC es

compatible con la mayoría de los programas y dispositivos principales, además es la principal alternativa a MP3, ya que con este se obtiene la calidad completa de audio sin comprimir ni procesar a la mitad del tamaño del archivo.

4. **ALAC:** (Apple Lossless Audio Codec), fue desarrollado y lanzado como un tipo de formato propietario por la compañía Apple en 2004, pero en 2011, se convirtió en código abierto y libre de regalías. Este formato resulta ser menos eficiente en comparación con otros tipos de formatos de audio como FLAC cuando se trata de compresión, pero los usuarios de Apple no tienen realmente una opción porque los sistemas operativos de iTunes y iOS brindan soporte para ALAC pero, no soporte para FLAC.

#### **B. Con compresión:**

1. **MP3:** (MPEG-1 Audio Layer 3), es el tipo de formato de audio comercial más empleado en el mundo. Fue lanzado en 1993 y explotó su popularidad, convirtiéndose así en el formato de audio más popular. Este formato tiene tres objetivos principales, eliminar todos los datos de sonido que existen más allá del rango auditivo de las personas normales, reducir la calidad de los sonidos que no son fáciles de escuchar y comprimir todos los demás datos de audio de la manera más eficiente posible. La mayoría de los dispositivos digitales con reproducción de audio pueden leer y reproducir archivos de tipo MP3.
2. **ACC:** (Advanced Audio Coding), se desarrolló en el año 1997 como el sucesor de MP3, se popularizó rápidamente como formato de uso, pero nunca superó al MP3 como el más popular.  
El algoritmo de compresión utilizado por ACC es mucho más avanzado y técnico que MP3, por lo que cuando compara la misma grabación en formatos MP3 y ACC con las mismas velocidades de bits, el ACC generalmente tendrá una mejor calidad de sonido. Este es el método de compresión de audio estándar utilizado por plataformas como YouTube, Android, iOS, iTunes así como portátiles y consolas como Nintendo y PlayStation.
3. **Ogg:** Es un formato de transmisión muy popular, pero a pesar de que es de código abierto, este es un formato contenedor libre y abierto, no es tan utilizado como MP3 o AAC. La plataforma Spotify es famosa por usar OGG como su formato de archivos de audio predeterminados. Este por lo general es el archivo de audio más pequeño, su problema principal está en su falta de compatibilidad. Este tipo de formatos cuentan con un tamaño que permite lograr la transmisión vía WiFi.
4. **Vorbis:** Este formato se lanzó por primera vez en el año 2000, se conoce como OGG, sin embargo OGG, no representa nada, incluso, ni siquiera en un formato de compresión. Es un contenedor multimedia que puede contener todo tipo de formatos de compresión, pero se usa más comúnmente para almacenar archivos Vorbis, su popularidad

creció debido a que se adhiere a los principios de código abierto y se desempeña de manera significativamente mejor que la mayoría de otros formatos de compresión con pérdida, produce un tamaño de archivo más pequeño para equivalente calidad de audio.

- **Conectores y cables utilizados para el sonido**

- C. USB**

- El formato USB puede ser utilizado para audio digital, requiere de procesamiento, sea por software o hardware. USB3.2 utilizando una velocidad de transmisión de 2.5 GB/s

- D. TS/TRS (Jack)**

- Implementa audio analógico por lo que no necesita conversiones, la señal que entrega es exacta la que capta. La forma con la que cuenta distribuye los polos de conexión en un solo cuerpo. Es de las más utilizadas.

- E. MIDI**

- Musical Instrument Digital Interface (MIDI), es principalmente utilizado para instrumentos musicales y está diseñada para que los instrumentos se puedan comunicar y controlar entre ellos.

- **Circuitos especializados para conversión Digital a Analógico para audio**

Los convertidores que se encargan de transformar las señales digitales en señales analógicas para poder escucharlas a través de altavoces o auriculares, son llamados DAC por sus siglas Digital Analog Converter, en español Convertidor Digital Analógico. El DAC se encarga de proporcionar la información necesaria a los dispositivos de salida de audio transformando/codificando la señal digital binaria a ondas.

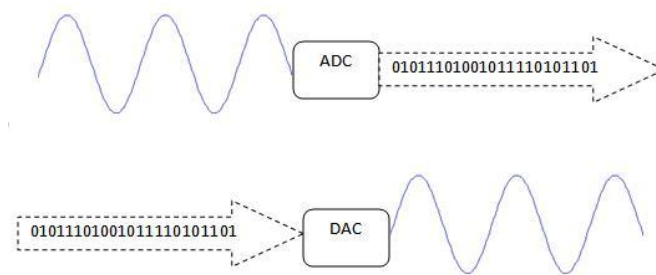


Figura 1. Conversores ADC y DAC

Las principales aplicaciones de DAC son para el procesamiento de señales digitales, ya que es mucho más fácil trabajar con señales una vez que son convertidas a formato binario, su segunda aplicación es como fuente de alimentación digital, esto se debe a que la mayoría de los microcontroladores son demasiado lentos para formar parte de un circuito de control de la fuente de alimentación, por lo que se utiliza el DAC para cambiar el voltaje y al mismo tiempo la referencia.

Existen 3 tipos de DAC especializados en la conversión digital a analógica para audio:

Amplificador sumador: Un amplificador sumador está diseñado para sumar las señales de entrada aplicadas para así generar una salida única, mientras que un amplificador sumador inversor combina varias entradas y produce una salida que es la suma de estas entradas y las señales de voltaje de entrada se aplican al terminal del inversor. El circuito diseñado para un amplificador sumador (Figura 2) consiste en varias entradas a aplicar que están conectadas al terminal inversor, este se beneficia de la configuración del inversor puede manejar muchas entradas al mismo tiempo.

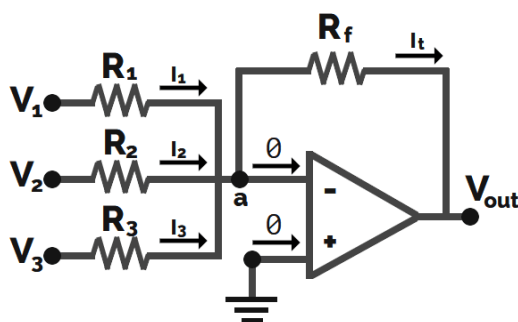


Figura 2. Amplificador sumador inversor

Sus dos aplicaciones principales son como mezclador de audio y como DAC, estos convertidores son utilizados en sistemas de monitoreo de tiempo real para monitorear los dispositivos a través de relevadores, sensores o motores en circuitos donde se utilizan microprocesadores o microcontroladores. Los valores de las resistencias se eligen de tal manera que el valor de resistencia de una rama debe ser el doble en comparación con el valor anterior de la resistencia.

Escalera R-2R: Este es un tipo de DAC muy simple ya que solo necesita dos valores de resistencia dispuestos en una escalera. La entrada binaria entre las resistencias  $2R$  y la salida se obtienen en la parte inferior de la escalera. El circuito de este DAC (Figura 3) tiene ciertas características, la primera es que su configuración es de tipo escalera, además es posible observar que se utilizan solamente dos valores de resistencias, en donde el segundo es el doble del primero, este DAC brinda un voltaje de salida que va entre 0 a 5 V.

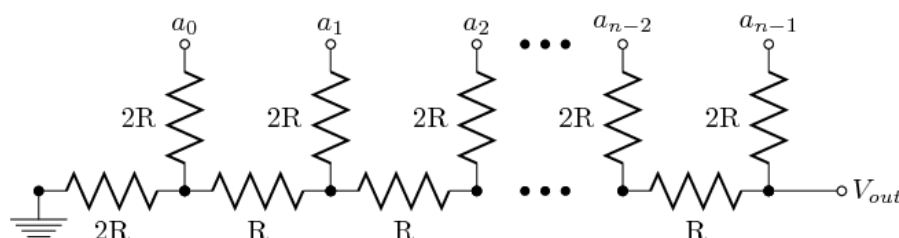


Figura 3. DAC Escalera R-2R

El funcionamiento de este circuito se puede entender mejor al hacer un análisis llamado, análisis de Thévenin.

Este tipo de circuito suele ser económico.



PWM DAC: La señal PWM al principio, tiene una forma de onda binaria con solo picos altos y bajos con un ciclo de trabajo variable, es decir con relación entre el tiempo y el período de tiempo. Está diseñado principalmente para usarse como filtro RC y convertir la señal PWM en un valor de voltaje filtrado.

Este es un método utilizado en microcontroladores como Arduino debido a su fácil programación para utilizar su función PWM para generar salidas analógicas. La onda PWM se filtra mediante un filtro de paso bajo para eliminar fluctuaciones y proporciona un voltaje analógico suave.

Otro tipo de DAC es el Delta-Sigma, es también muy utilizado.

- **Circuito VS1053B** utilizado para decodificar audio por compresión

Contiene un alto rendimiento y un núcleo de procesador DSP de bajo consumo, data memory, 16 KiB de instrucción RAM y 0.5+ KiB RAM de datos para aplicaciones de usuario que se ejecutan simultáneamente con cualquier decodificador incorporado, control serie e interfaces de datos de entrada, hasta 8 pines de E/S de uso general, un UART, así como un ADC estéreo de alta calidad variable y estéreo DAC, seguido de un amplificador de auriculares y un búfer de voltaje común.

El circuito VS1053B (Figura 4.) recibe su entrada en bitstream a través de un bus de entrada serie, al que escucha como esclavo del sistema. La secuencia de entrada se decodifica y pasa por un control de volumen digital a un DAC sigma delta de 18 bits con sobremuestreo y varios bits. La decodificación se controla mediante un bus de control serie. Además de la descodificación básica, es posible añadir al usuario características específicas de la aplicación, como los efectos DSP.

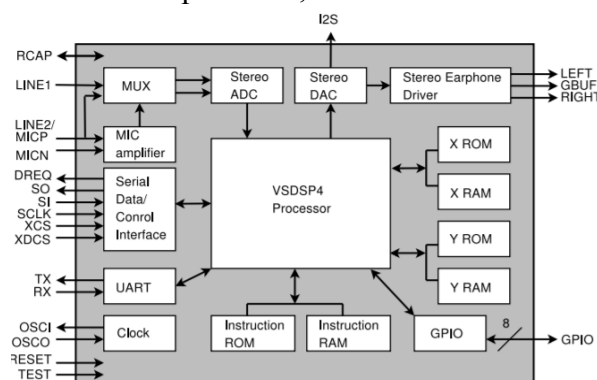


Figura 4. Circuito VS1053B

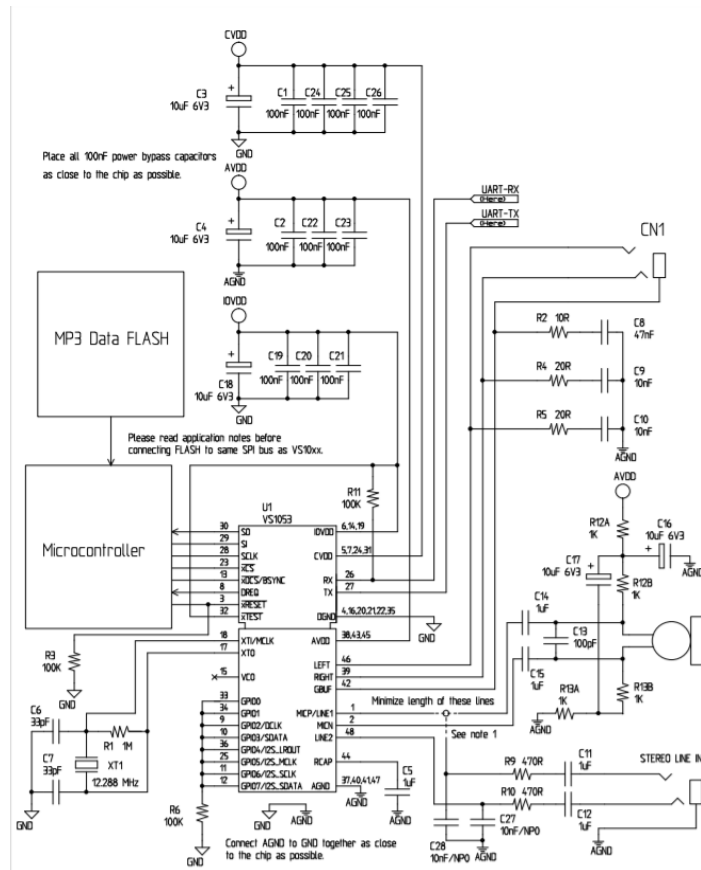


Figura 5. Diagrama de conexión típica

La figura 5 muestra el diagrama de conexión típica para un VD1053B utilizando un LQFP-48, esta conexión asume que SM\_SDINew está activo. Si SM\_SDISHARE también es utilizado, xDCS deberá ser atado a low o high.

Aquí el buffer compun GBUF puede ser utilizado para voltaje común para audífonos. Esto eliminar la necesidad de una larga isolación de capacitores en salidas de línea, por lo que el pin de salida de audio puede ser conectado directamente al conector del audífono.

Este dispositivo cuenta con interfaz de datos de serie e interfaz de comandos de serie, cada una de estas interfaces tiene sus respectivos protocolos.

El protocolo serial para la Interfaz de datos de serie, o por sus siglas SPI/SDI establece que la interfaz de datos seriales funcionan de modo esclavo, por lo que la señal DCLK debe ser generada por un circuito externo, mientras que los datos se pueden registrar en el borde ascendente o descendente del DCLK. Así mismo el VS1053B asume que la entrada de datos está sincronizada en bytes, dichos bytes SDI se pueden transmitir MSb o LSb primero, dependiendo del registro SCI\_MODE. El firmware puede aceptar la tasa de bits máxima que admite el SDI.

Por otro lado, el protocolo serial de bus para el SCI (Serial Command Interface) consta de un byte de instrucción, byte de dirección y una palabra de datos de 16 bits. Cada operación de lectura o escritura puede leer o escribir un registro único. Los bits de datos se leen al borde ascendente, por lo que el usuario debe actualizar los datos en el borde que cae. La operación se especifica mediante un código de operación de

instrucción de 8 bits. Las instrucciones admitidas son de lectura y escritura (read, write).

- **Alternativas para el control de volumen en un reproductor de audio**

Existen varias alternativas para controlar el volumen del reproductor, de manera analógica se tiene como opción un potenciómetro, por otro lado de manera digital se puede utilizar un potenciómetro digital o piezas tipo chip como por ejemplo PT2258 IC o incluso PGA2311.

- **Circuitos amplificadores de audio.**

Amplificador es el término general utilizado para describir un circuito que produce una versión aumentada de su señal de entrada. Sin embargo, no todos los circuitos amplificadores son iguales, ya que se clasifican de acuerdo con sus configuraciones de circuitos y modos de operación.

En electrónica, los amplificadores de señal se consideran dispositivos de uso común, ya que tienen la capacidad de amplificar señales de entrada pequeñas, por ejemplo la de un altavoz.

Hay muchas formas de circuitos que son clasificados como amplificadores, estos van desde amplificadores operacionales y amplificadores de señal hasta grandes amplificadores de señal y potencia. Los amplificadores se clasifican dependiendo del tamaño de la señal, ya sea grande o pequeña, así como conforme a su tipo de configuración física y frecuencia de operación.

Tipo de señal	Tipo de configuración	Clasificación	Frecuencia de operación
Pequeña señal	Emisor común	Amplificador de clase A	Corriente continua (DC)
Gran señal	Base común	Amplificador de clase B	Frecuencias de audio (AF)
	Colector común	Amplificador de clase AB	Frecuencias de radio (RF)
		Amplificador de clase C	Frecuencias VHF, UHF y SHF

Tabla 1. Clasificación de amplificadores de señal

Un amplificador de señal ideal, tiene tres propiedades principales, la resistencia de entrada, la resistencia de salida y por supuesto la amplificación, mejor conocida como ganancia.

El modelo de amplificador ideal (Figura 6), tiene como característica que la diferencia amplificada entre las señales de entrada y salida se conoce como ganancia del amplificador. La ganancia es básicamente una medida de cuánto se amplifica la señal de entrada.

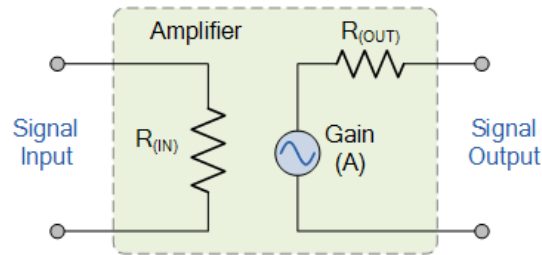


Figura 6. Amplificador ideal.

Por otro lado, los amplificadores de audio de potencia modesta para controlar altavoces pequeños o incluso otras cargas ligeras se pueden construir de varias maneras. La primera opción suele ser un circuito integrado diseñado para ese fin específico, como es el caso del LM386, este amplificador muestra una configuración de alta ganancia (Figura 7).

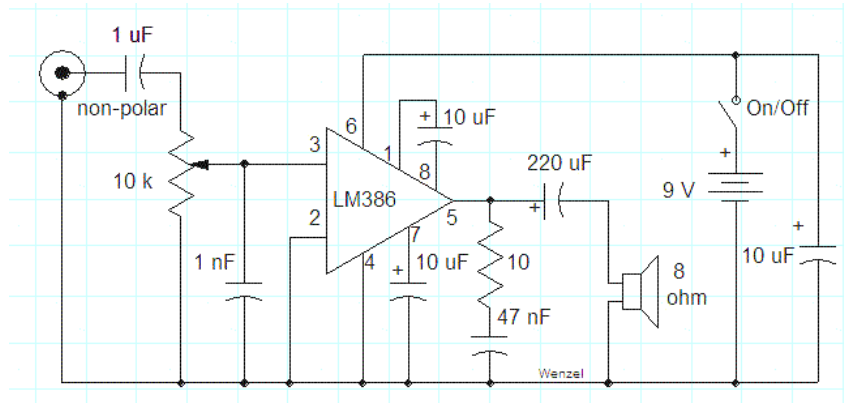


Figura 7. Amplificador de audio simple LM386

Otro tipo de amplificador es el de audio de computadora, este amplificador aumenta el nivel de audio de tarjetas de sonido de baja potencia u otras fuentes de audio que controlan pequeños altavoces como juguetes o pequeñas radios de transistores. El circuito (Figura 8). Entrega aproximadamente 2 vatios.

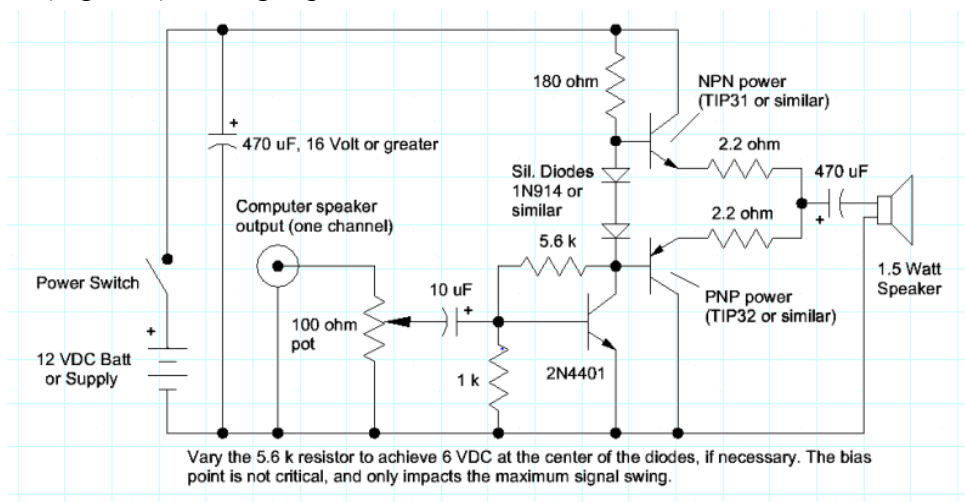


Figura 8. Amplificador de audio de computadora.

Otro tipo de amplificador es de 4 transistores para aplicaciones de altavoces pequeños. Este tipo de circuito (Figura 9.) muestra un amplificador de utilidad de 4 transistores adecuado para diferentes proyectos que incluyen receptores, intercomunicadores, micrófonos e incluso monitoreo de audio en general, este amplificador tiene un circuito de aislamiento de potencia y limitación de ancho de banda para reducir las oscilaciones. Se puede modificar para que funcione con una batería de 9 voltios.

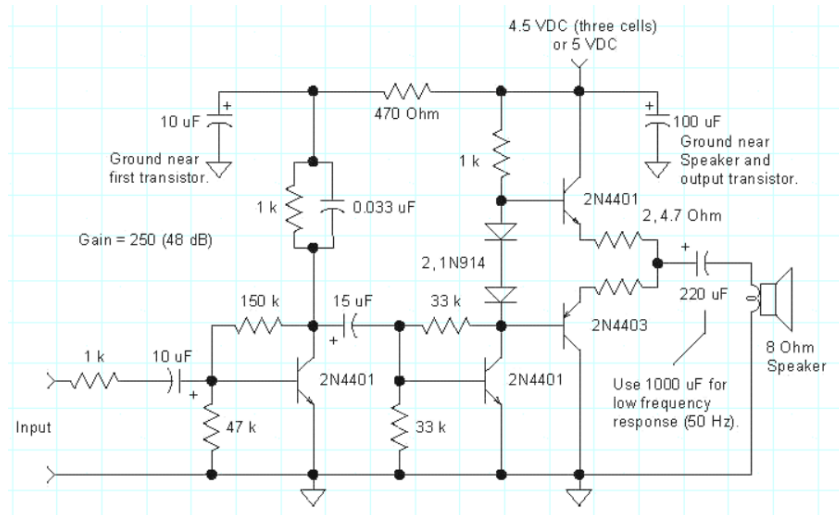


Figura 9. Amplificador para altavoces pequeños.

Por otro lado también se tiene el Amplificador de audio Op-Amp (Figura 10.), este amplificador es un amplificador de audio versátil que emplea un amplificador operacional LM358 de bajo costo. Las entradas diferenciales dan al amplificador una excelente inmunidad a las señales en modo común que son causa de inestabilidad. Estos amplificadores son de los más comunes debido principalmente a su bajo costo.

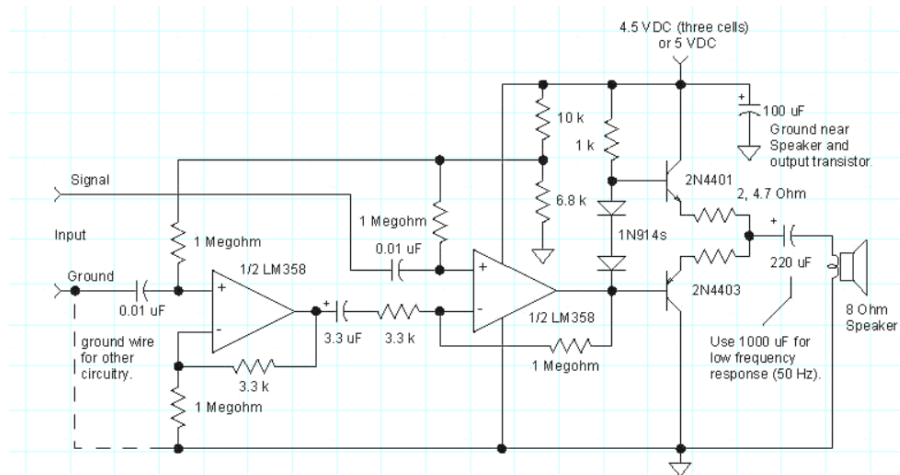


Figura 10. Amplificador Op-Amp

- **Tipos de LCDs para conexión con un microcontrolador**  
**LCD alfanuméricos**

Se considera uno de los display más utilizados debido a que son fáciles y rápidos de configurar, si el usuario necesita desplegar un carácter basta con mandar el carácter y no es necesario tomar en cuenta cada punto que forma ese carácter, así como que el

usuario es capaz de crear sus propios caracteres. Hay múltiples fabricantes y distribuidores lo cual hace que sean compatibles entre ellos y que el precio no sea tan variado. Estos displays cuentan con un diseño “Chip on Board” o COB que cuenta con un circuito impreso junto a la pantalla de cristal líquido. Módulos como estos no son dañados por vibraciones.

### **Gráficos Monocromáticos**

Este tipo de LCD es capaz de desplegar imágenes, letras y números. Dependiendo del tamaño del LCD, la pantalla cuenta con puntos, a los cuales se les conoce como píxeles, que pueden ser apagados o prendidos interdependientemente a cómo el usuario los programe.

### **OLED**

Displays de diodos emisores de luz orgánica, son más brillantes, consumen menor energía, ofrecen ángulos de visión más amplios. Se componen por material orgánico emisor de luz que se encuentra entre dos placas conductoras una de material tipo N y otra de tipo P. Este tipo de LED es utilizado para la producción de televisores, sistemas portátiles, consolas de videojuegos, entre otros.

- **Manera de conectar una memoria SD a un microcontrolador: interfaz que se utiliza (con descripción de cada señal) y hardware extra necesario si el voltaje del microcontrolador es de 5 volts.**

Una memoria microSD cuenta con la arquitectura que se muestra en la Figura 11 la cual se describe en la Tabla 2, en el cual se puede observar que se tienen 4 líneas de datos, una línea de comando/respuesta, una línea de alimentación, línea de tierra y una línea de reloj. Este tipo de memoria utiliza el método de comunicación “Serial Peripheral Interface” (SPI) el cual usa la señal de reloj para definir la velocidad de transmisión de datos, así mismo se muestra en la tabla 2 los diferentes usos que se le da a las salidas y entradas con este protocolo. El voltaje recomendado para una memoria SD es de 2.7 to 3.6 volts cualquier voltaje mayor dañara la memoria. En el caso de un microcontrolador que esté alimentado a 5 volts, la alternativa sería tener un circuito interfaz el cual actúa como filtro y poder atenuar la señal a un voltaje que la memoria pueda obtener. Otra opción sería reducir el voltaje de alimentación que se le da al microcontrolador para que tenga salidas que estén dentro de los parámetros de la SD

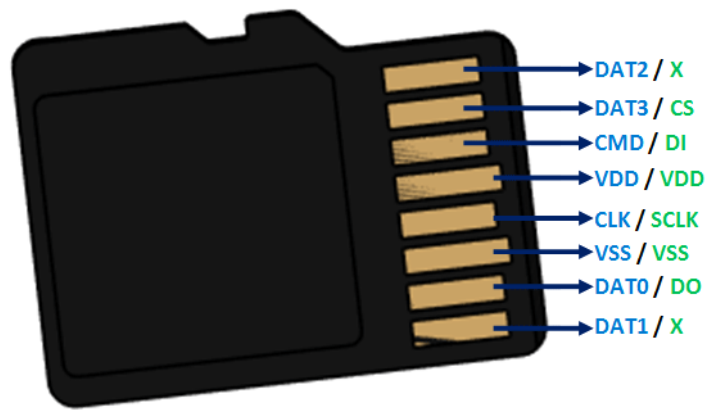


Figura 11. Pinout de una memoria microSD

Pin Number	Pin Name	In SD Mode	In SPI Mode
1	DAT2/X	Connector Data line 2	No use
2	DAT3/CS	Connector Data line 3	Chip Select
3	CMD/DI	Command / Response Line	Data Input
4	VDD/VDD	Power supply (+3.3V)	Power supply (+3.3V)
5	CLK/SCLK	Clock	Serial Clock
6	VSS/VSS	Ground	Ground
7	DAT0/DO	Connector Data line 0	Data Out
8	DAT1/X	Connector Data line 1	No use

Tabla 2. Desglose de los pines de una memoria microSD

- **Revisar el siguiente proyecto:** <http://elm-chan.org/works/sd8p/report.html>

El proyecto mostrado en el sitio web, consiste en un reproductor de audio SD simple con un IC de 8 pines, el microcontrolador AVR de 8 pines utilizado fue de la serie ATtinyX5, este tiene dos salidas PWM rápidas en una frecuencia portadora de 250kHz. Lo que permite emitir señales de audio de amplios rangos de frecuencia. La construcción de este reproductor de audio fue por medio del AVR de 8 pines y una tarjeta de memoria SD, esta tarjeta de memoria se controla con tan sólo 6 líneas, 2 para alimentación y 4 de control.

En cuanto al Hardware se tienen cuatro esquemas para el reproductor en diferentes configuraciones de salida. El medio de grabación que se utilizó fue una microSD. Los altavoces funcionan directamente con la salida PWM del AV, a pesar de que la carga puede ser demasiado pesada para el puerto de salida, la corriente nunca excede las clasificaciones máximas absolutas a un voltaje de suministro de 3 volts, por lo tanto el

sonido no es muy alto, pero se crea suficiente sonido para escuchar con altavoces de alta frecuencia.

Por otro lado, en la parte de Software, el uso de tarjetas de memoria en proyectos integrados, significa que el firmware debe admitir el sistema de archivos FAT. Hay algunos archivos de firmware diferentes en el archivo para cada configuración de hardware. Los valores de fusible se combinan en los archivos hexadecimales como lo hace PIC, sin embargo algunos programadores AVR no admiten este tipo de archivos hexadecimales, en caso de una situación de este tipo, es necesario quitar la sección de fusible del archivo hexadecimal y escribirla por separado.

La configuración Mono usa 5 pines entrada y salida y no es necesario cambiar el fusible, además se puede programar en el método ISP. Las configuraciones de salida dual, estéreo, alta resolución y OCL, utilizan todos los pines de entrada y salida, incluido el pin de Reset.

Este reproductor de audio SD solo admite archivos de sonido en formato RIFF-WAVE conocidos como archivos de onda de Microsoft en LPCM, 8/16 bits, mono/estéreo y una frecuencia de muestreo de hasta 48 kHz. Los archivos MP3 se deberán convertir a archivo de tipo .wav, es reproductor inicial al encenderlo o al insertar la tarjeta, si se tiene un directorio de archivos de sonido de tipo wav en el directorio de raíz, los archivos de sonido en el directorio se reproducirá en el orden la lista de directorios, en cambio, si no existe dicho directorio, en su lugar se reproducen los archivos wave. Cuenta con un interruptor de botón para saltar al siguiente archivo.

- **Costo de un reproductor de música realizado con diferentes tecnologías:**

- **Microcontrolador de 8 bits, SD y un LCD Alfanumérico**

Usando el microcontrolador de 8 bits ATmega328P que tiene un costo aproximado de 2 dólares, que serían equivalentes a 40 pesos, tomando una memoria micro SD de 32 GB con un costo de aproximadamente 180 pesos y un display LCD alfanumérico de 2x16 con un costo de aproximadamente 80 pesos. Esta configuración para un reproductor de música tendría un costo aproximado de 300 pesos.

- **Microcontrolador de 8 bits, SD y un LCD Gráfico**

Usando el microcontrolador de 8 bits ATmega328P que tiene un costo aproximado de 2 dólares, que serían equivalentes a 40 pesos, tomando una memoria micro SD de 32 GB con un costo de aproximadamente 180 pesos y un display LCD gráfico de 128x64 con un costo de aproximadamente 200 pesos. Esta configuración para un reproductor de música tendría un costo aproximado de 420 pesos.

- **Microcontrolador de 8 bits, SD, decodificador de MP3 y un LCD Gráfico**

Usando el microcontrolador de 8 bits ATmega328P que tiene un costo aproximado de 2 dólares, que serían equivalentes a 40 pesos, tomando una memoria micro SD de 32 GB con un costo de aproximadamente 180 pesos, un display LCD gráfico de 128x64 con un costo de aproximadamente 200 pesos y un decodificador de MP3 VS1053B de costo aproximado de 450 pesos. Esta



configuración para un reproductor de música tendría un costo aproximado de 870 pesos.

- **Raspberry y un LCD gráfico**

Tomando en cuenta el modelo de Raspberry Pi 3 B +, que tiene un costo aproximado de 1,100 pesos y un display LCD gráfico de 128x64 con un costo de aproximadamente 200 pesos, esta configuración tiene un costo aproximado de 1,300 pesos.

#### **IV. Propuesta del sistema que se desarrollará en esta unidad de formación**

Para la solución del reto, se planea desarrollar un sistema de entretenimiento para un automóvil, haciendo uso del microcontrolador ATmega2560 que está incluido en el Arduino Mega. Nuestra propuesta consiste principalmente en realizar un programa donde se pueda llevar el control de las canciones que se están reproduciendo mediante botones, que datos de estas puedan ser desplegados en un display LCD alfanumérico, las canciones serán almacenadas en una memoria microSD.

#### **V. Diagrama eléctrico.**

##### **figura 13. Diagrama eléctrico**

#### **VI. Código fuente.**

```
#asm

.equ __lcd_port=0x02
.equ __lcd_EN=1
.equ __lcd_RS=0
.equ __lcd_D4=2
.equ __lcd_D5=3
.equ __lcd_D6=4
.equ __lcd_D7=5

#endasm

#include <io.h>
#include <delay.h>
#include <stdio.h>
#include <ff.h>
#include <display.h>

//Código base que reproduce A001.WAV que es un WAV, Mono, 8-bit, y
frec de muestreo de 22050HZ

char bufferL[256];
char bufferH[256];
char NombreArchivo[] = "0:A001.wav";
```

```

unsigned int i=0;
unsigned int j=0;
unsigned int z=0;
bit LeerBufferH,LeerBufferL;
unsigned long muestras;
bit play;
char song=1;
bit stereo;

interrupt [TIM1_COMPA] void timer1_compa_isr(void)
{
disk_timerproc();
/* MMC/SD/SD HC card access low level timing function */
}

//Interrupción que se ejecuta cada T=1/Fmuestreo_Wav
interrupt [TIM2_COMPA] void timer2_compa_isr(void)
{
    if (stereo==0)
    {
        if (i<256)
        {
            OCR0A=bufferL[i];
            OCR0B=bufferL[i++];
        }
        else
        {
            OCR0A=bufferH[i-256];
            OCR0B=bufferH[i-256];
            i++;
        }
    }
    if (i==256)
        LeerBufferL=1;
    if (i==512)
    {
        LeerBufferH=1;
        i=0;
    }
}
else

```

```

{
    if (i<256)
    {
        OCR0A=bufferL[i++];
        OCR0B=bufferL[i++];
    }
    else
    {
        OCR0A=bufferH[i-256];
        i++;
        OCR0B=bufferH[i-256];
        i++;
    }
    if (i==256)
        LeerBufferL=1;
    if (i==512)
    {
        LeerBufferH=1;
        i=0;
    }
}

}

void main()
{
    unsigned int  br;

    /* FAT function result */
    FRESULT res;

    /* will hold the information for logical drive 0: */
    FATFS drive;
    FIL archivo; // file objects

    CLKPR=0x80;
    CLKPR=0x01;          //Cambiar a 8MHz la frecuencia de operación
del micro

    // Código para hacer una interrupción periódica cada 10ms
    // Timer/Counter 1 initialization

```

```

// Clock source: System Clock
// Clock value: 1000.000 kHz
// Mode: CTC top=OCR1A
// Compare A Match Interrupt: On
TCCR1B=0x0A;      //CK/8 10ms con oscilador de 8MHz
OCR1AH=0x27;
OCR1AL=0x10;
TIMSK1=0x02;

//PWM para conversión Digital Analógica WAV->Sonido
// Timer/Counter 0 initialization
// Clock source: System Clock
// Clock value: 8000.000 kHz
// Mode: Fast PWM top=0xFF
// OC0A output: Non-Inverted PWM
// TCCR0A=0x83;
TCCR0A=0xA3;

DDRB.7=1; //Salida bocina (OC0A)
DDRG.5=1; //Salida bocina (OC0B)

// Timer/Counter 2 initialization
// Clock source: System Clock
// Clock value: 1000.000 kHz
// Mode: CTC top=OCR2A
ASSR=0x00;
TCCR2A=0x02;
TCCR2B=0x02;
OCR2A=0x2C;

// Timer/Counter 2 Interrupt(s) initialization
TIMSK2=0x02;

DDRD.7=1;
#asm("sei")
disk_initialize(0); /* Inicia el puerto SPI para la SD */
delay_ms(500);

SetupLCD();
PORTC.7=1;
PORTC.6=1;

```

```

PORTC.5=1;

/* mount logical drive 0: */
if ((res=f_mount(0,&drive))==FR_OK){    ;
    while(1)
    {
        NombreArchivo[5]=song+'0';
        /*Lectura de Archivo*/
        res = f_open(&archivo, NombreArchivo, FA_OPEN_EXISTING |
FA_READ);
        if (res==FR_OK){
            PORTD.7=1;
            f_read(&archivo, bufferL, 44,&br); //leer encabezado

muestras=(long)bufferL[43]*16773216+(long)bufferL[42]*65536+(long)buff
erL[41]*256+bufferL[40];
            f_lseek(&archivo,muestras+44+20);
            f_read(&archivo, bufferH, 100, &br);
            EraseLCD();
            for (j=0;j<100;j++){
                if(bufferH[j]==0){
                    z=j;
                    break;
                }
                CharLCD(bufferH[j]);
            }
            MoveCursor(0,1);
            for (j=z;j<100;j++){
                if (bufferH[j]==0x54){
                    z=j+5;
                    break;
                }
            }

            for (j=z;j<100;j++){
                if(bufferH[j]==0)
                    break;
                CharLCD(bufferH[j]);
            }

```

```

f_lseek(&archivo,44);

//leer la frecuencia de muestreo del encabezado
if(bufferL[24]==0x22)
    OCR2A=0x2C;
if(bufferL[24]==0xC0)
    OCR2A=0x29;
if(bufferL[24]==0x80)
    OCR2A=0x3E;

if (bufferL[22]==1)
    stereo=0;
else
    stereo=1;
i=0;

f_read(&archivo, bufferL, 256,&br); //leer los primeros
512 bytes del WAV
f_read(&archivo, bufferH, 256,&br);
LeerBufferL=0;           //banderas
LeerBufferH=0;
TCCR0B=0x01;    //Prende sonido
do{
    while((LeerBufferH==0)&&(LeerBufferL==0));    // si
cualquiera de los dos sea diferente de 0 se sale del while
    if (LeerBufferL)
    {
        f_read(&archivo, bufferL, 256,&br); //leer
encabezado
        LeerBufferL=0;
    }
    else
    {
        f_read(&archivo, bufferH, 256,&br); //leer
encabezado
        LeerBufferH=0;
    }

    if (PINC.7==0){    //play y pausa

```

```

        play=~play;
        if (play==1){
            TCCR0B=0x00;
            while (PINC.7==0);
        }
        else{
            TCCR0B=0x01;
            while (PINC.7==0);
        }
    }

    if (PINC.5==0){        //reinicia cancion
        TCCR0B=0x00;
        delay_ms(30); //rebote al presionar
        f_lseek(&archivo,44);
        while (PINC.7==0);
        delay_ms(10); //rebote al soltar
        TCCR0B=0x01;      //Prende sonido
    }

    if (PINC.6==0){        // cambia de cancion
        TCCR0B=0x00;
        break;
    }

}while(br==256);
TCCR0B=0x00; //Apaga sonido
f_close(&archivo);
song++;
if (song==7)
    song=1;

}
}

}

f_mount(0, 0); //Cerrar drive de SD
while(1);
}

```

## **VII. Explicación del programa y funcionamiento.**

El programa hace uso de los diferentes Times integrados dentro del microcontrolador ATmega2560 para la lectura y reproducción de música. Las tareas específicas fueron las siguientes:

Timer 0: En modo de Fast PWM se utiliza para la conversión digital-analógica, con una salida en el registro OC0A en PD6 para la primera bocina y PG5 para la segunda bocina.

Timer 1: Se utiliza para la interrupción que se necesita para la lectura de la memoria SD.

Timer 2: En modo de CTC usando un pre escalador de 8, realiza interrupciones periódicas cada periodo de reproducción.

Leyendo el encabezado del archivo WAV en el cual indica la frecuencia de muestreo de la canción, se calcula el inverso de esta frecuencia y se obtiene el periodo de reproducción, nuestro programa es capaz de leer la frecuencia de muestreo y en base a esta, selecciona el valor correspondiente para el registro de comparación del Timer 2, OCR2A. Las frecuencias de muestreo las cuales fueron utilizadas dentro del proyecto fueron de 16 KHz, 24 KHz, 22.050KHz.

Respecto a la funcionalidad de los botones, para el botón de pausa y play, se cuenta con una bandera llamada “play” la cual indica si se está reproduciendo la canción o no. Al momento de presionar el botón de play/pausa (pin PC7), se evalúa el estado de la bandera y dependiendo de esta, se activa o se desactiva el Timer 0. Para el botón que reinicia la canción (pin PC5), se interrumpe el Timer 0 para volver a leer el archivo desde el inicio, una vez que el archivo está en el inicio nuevamente se vuelve a activar el Timer 0 para el sonido, en este botón se manejo el rebote al presionar por medio de delays. Por otro lado, para el cambio de canción al presionar el botón (pin PC6), se sale del ciclo de lectura y procesamiento del archivo, se interrumpe el Timer 1 para apagar el sonido, y cierra el archivo que estaba abierto, debido a que los archivos tienen un nombre en donde solo cambia el último número, sólo es necesario aumentar en uno el número del nombre del archivo, en este caso, por medio de la variable de tipo char “song”.

## **VIII. Fotos del circuito eléctrico.**



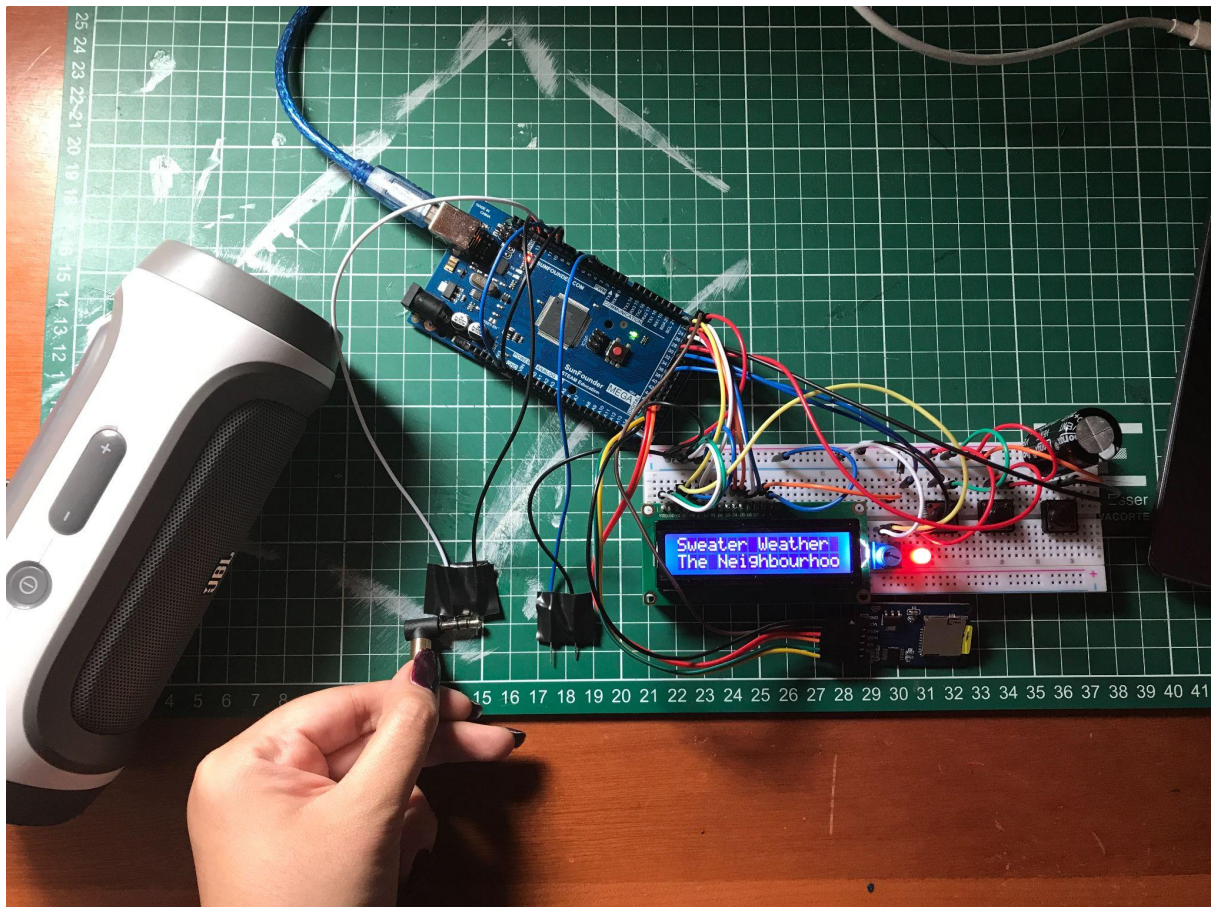


Figura 14. Fotografía del circuito.

#### IX. Video de funcionalidad.

Youtube: <https://youtu.be/IwuqqIHCQA0>

Drive:

<https://drive.google.com/file/d/1dMFhsHZObUUhUqbkYeOw9fj7Q6e8ZU6v/view?usp=sharing>

#### X. Conclusiones.

Marcela Ibarra: La realización de este reto, nos ayudó a desarrollar habilidades que nos ayudarán en nuestra carrera, como el desarrollo de interfaces inteligentes, sistemas embebidos, entre otros. Así mismo, nos ayudó a hacer análisis de los requerimientos del proyecto y en base a eso elegir el hardware necesario para resolverlo de manera eficiente teniendo en cuenta los gastos.

Mariana Castro: El desarrollo de este reto contribuyó altamente al desarrollo de nuevas competencias que serán de gran ayuda para nuestra vida profesional, ya que nos permitió lograr la implementación de sistemas embebidos con características específicas, los cuales con cada vez más importantes y usados en la industria, además adquirimos una gran cantidad de nuevos aprendizajes de valor.

## **XI. Referencias.**

Alonso, R. (2021, 5 mayo). ¿Qué tipos de conexiones de audio hay y cuál es mejor utilizar? HardZone.

<https://hardzone.es/tutoriales/componentes/tipo-conexiones-audio/>

Chan, E. (s. f.). How to Use MMC/SDC. Elm Chan. Recuperado 6 de junio de 2021, de [http://elm-chan.org/docs/mmc/mmc\\_e.html#pinout](http://elm-chan.org/docs/mmc/mmc_e.html#pinout)

Componentes 101. (2019). Introduction to Digital to Analog Converters (DAC). Recuperado el 6 de junio de 2021 del sitio web:

<https://components101.com/articles/digital-to-analog-converters-dac>

Digitalización. (s. f.). Multimedia. Recuperado 4 de junio de 2021, de

<http://www2.udec.cl/%7Elsalazarv/digitalizacion.html>

Display Lcd Pantalla 16x2 + Interfaz I2 Compatible Arduino. (s. f.). Mercado Libre. Recuperado 5 de junio de 2021, de

[https://articulo.mercadolibre.com.mx/MLM-796794524-display-lcd-pantalla-16x2-interfaz-i2-compatible-arduino-\\_JM#position=1&search\\_layout=stack&type=pad&tracking\\_id=554738e9-486d-43d2-a48b-9590e41b23cb&is\\_advertising=true&ad\\_domain=VOCATCORE\\_LST&ad\\_position=1&ad\\_click\\_id=NzNmYWwNiM2YtMjFkYi00Y2ZmLWI2ZTEtZjYxMmIwNTk1MWZh](https://articulo.mercadolibre.com.mx/MLM-796794524-display-lcd-pantalla-16x2-interfaz-i2-compatible-arduino-_JM#position=1&search_layout=stack&type=pad&tracking_id=554738e9-486d-43d2-a48b-9590e41b23cb&is_advertising=true&ad_domain=VOCATCORE_LST&ad_position=1&ad_click_id=NzNmYWwNiM2YtMjFkYi00Y2ZmLWI2ZTEtZjYxMmIwNTk1MWZh)

Display Lcd Pantalla 128x64 Gráfico Fondo Azul Arduino Pic. (s. f.). Mercado Libre. Recuperado 4 de junio de 2021, de

[https://articulo.mercadolibre.com.mx/MLM-695654857-display-lcd-pantalla-128x64-grafico-fondo-azul-arduino-pic-\\_JM#position=5&search\\_layout=grid&type=item&tracking\\_id=38599a6f-52ff-4b95-b8d6-cd7d2db7d05c](https://articulo.mercadolibre.com.mx/MLM-695654857-display-lcd-pantalla-128x64-grafico-fondo-azul-arduino-pic-_JM#position=5&search_layout=grid&type=item&tracking_id=38599a6f-52ff-4b95-b8d6-cd7d2db7d05c)

Electrical Technology (s.f) Digital to Analog Converter (DAC) - Types, working & Applications. Recuperado el 6 de junio de 2021 del sitio web:

<https://www.electricaltechnology.org/2020/04/digital-to-analog-converter-dac.html>

Electrónica Fácil. (2018). Amplificador Sumador DAC. Recuperado el 6 de junio de 2021 del sitio web:

<https://mielectronicafacil.com/analogica/amplificador-sumador-inversor/>

Electronics Tutorials. (2020). Introduction to the amplifier. Recuperado el 6 de junio de 2021 del sitio web: [https://www.electronics-tutorials.ws/amplifier/amp\\_1.html](https://www.electronics-tutorials.ws/amplifier/amp_1.html)

González, Y (2020). 18 tipos de formato de audio y sus características. Recuperado el 6 de junio de 2021 del sitio web Grupo Ático 34:

<https://protecciondatos-l opd.com/empresas/tipos-formato-audio/>

Hahn, M (2018). Audio Compression 101: How to use a compressor for a better mix. Recuperado el 5 de junio de 2021 del sitio web:

<https://blog.landr.com/how-to-use-a-compressor/>

Hawkins, E. (2017, 21 noviembre). Alphanumeric LCD Displays. Focus LCDs.

<https://focuslcds.com/journals/alphanumeric-lcd-displays/>

Hawkins, E. (2017, 21 noviembre). An Introduction to Graphic LCD Displays. Focus LCDs. <https://focuslcds.com/journals/an-introduction-to-graphic-lcd-displays/>

Hoffman, C. (2020, 28 febrero). Mono vs. Stereo Sound: The Difference Explained (With Audio Examples) | Black Ghost Audio. Black Ghost Audio.

<https://www.blackghostaudio.com/blog/mono-vs-stereo-sound-the-difference-explained-with-audio-examples>

IBM (2020). Formatos de Audio. Recuperado el 6 de junio de 2021 del sitio web:

<https://cloud.ibm.com/docs/speech-to-text?topic=speech-to-text-audio-formats&locale=es>

Lee, J (2019). The 10 Most Common Audio Formats: Which One Should You Use?. Recuperado el 6 de junio de 2021 del sitio web Make Use Of:

<https://www.makeuseof.com/tag/audio-file-format-right-needs/>

Microchip. (s. f.). ATmega328P - 8-bit AVR Microcontrollers. Recuperado 4 de junio de 2021, de <https://www.microchip.com/wwwproducts/en/ATmega328P>

Mp3 Tablero De Música Vs1053b Grabación/amplificador De Pote. (s. f.). Mercado Libre. Recuperado 4 de junio de 2021, de

[https://articulo.mercadolibre.com.mx/MLM-919078953-mp3-tablero-de-musica-vs1053b-grabacionamplificador-de-pote-\\_JM#position=1&search\\_layout=grid&type=item&tracking\\_id=6adedfd2-ab09-4686-9725-3bb22c10bfbb](https://articulo.mercadolibre.com.mx/MLM-919078953-mp3-tablero-de-musica-vs1053b-grabacionamplificador-de-pote-_JM#position=1&search_layout=grid&type=item&tracking_id=6adedfd2-ab09-4686-9725-3bb22c10bfbb)

Musicaler. (2016). La compresión. Recuperado el 5 de junio de 2021 del sitio web:  
<https://musicalecer.com/apendices/la-dinamica/la-compresion/>

Navarro, K. (2014). R-2R Ladder en conversor digital analógico (CDA). Recuperado el 6 de junio de 2021 del sitio web:  
<http://panamahitek.com/r-2r-ladder-en-conversor-digital-analogico/>

Network México. (s. f.). OLED. Recuperado 6 de junio de 2021, de  
<https://mexico.newark.com/display-oled-technology?ICID=I-CT-TP-BROWSE-3>

Office Depot México. (2021, 4 junio). Memoria Micro SD con Adaptador SanDisk Ultra / 32gb / SDHC / UHS-I / A1 / Clase 10.  
<https://www.officedepot.com.mx/officedepot/en/Categor%C3%ADa/Todas/C%C3%B3mputo/Almacenamiento/SD-y-Micro-SD/Memoria-Micro-SD-con-Adaptador-SanDisk-Ultra-32gb-SDHC-UHS-I-A1-Clase-10/p/82052>

Tech Lib (2021). Audio Amplifiers. Recuperado el 6 de junio de 2021 del sitio web:  
<http://techlib.com/electronics/audioamps.html>

Techopedia. (2015, 5 enero). Monaural Sound. Techopedia.Com.  
<https://www.techopedia.com/definition/31085/monaural-sound>

Vincent, S (2012). The beginner's guide to compression. Recuperado el 5 de junio de 2021 del sitio web:  
<https://music.tutsplus.com/tutorials/the-beginners-guide-to-compression--audio-953>

300ofms. (s. f.). Raspberry Pi 3 Modelo B+ Plus. 330ohms. Recuperado 4 de junio de 2021, de <https://330ohms.com/products/raspberry-pi-3-modelo-b?src=raspberrypi>