



Universidad Autónoma Metropolitana

Unidad Azcapotzalco
Licenciatura en Ingeniería Electrónica

" Desarrollo de un fonocardiógrafo portátil inalámbrico conectado a dispositivos dentro de la red local y dispositivos móviles"

Proyecto tecnológico
Segunda Versión

Datos de la Alumno:

Nombre: Mario Alejandro Pavón Heredia

Matrícula: 2163002562

Correo electrónico: al2163002562@azc.uam.mx

Firma

Datos de la asesora:

Nombre: Grethell Georgina Pérez Sánchez

Categoría: Asociado

Nivel Académico: Doctorado

Departamento de Adscripción: Departamento de Electrónica

Correo Electrónico: ggps@azc.uam.mx

Firma

Datos del coasesor:

Nombre: Ernesto Rodrigo Vázquez Cerón

Categoría: Titular

Nivel Académico: Doctorado

Departamento de Adscripción: Departamento de Electrónica

Correo Electrónico: ervc@azc.uam.mx

Firma

Trimestre 22-I.

29 de junio del 2023

1 Declaratoria

En caso de que el Comité de Estudios de la Licenciatura en Electrónica apruebe la realización de la presente propuesta, otorgamos nuestra autorización para su publicación en la página de la División de Ciencias Básicas e Ingeniería.



Alumno
Mario Alejandro Pavón Heredia



Asesora
Dra. Grethell Georgina Pérez Sánchez
Departamento de Electrónica



Co-asesor
Dr. Ernesto Rodrigo Vázquez Cerón
Departamento de Electrónica

2 Dedicatoria

En primer lugar, quiero agradecerme a mí mismo. Cada paso, cada esfuerzo y cada sacrificio me ha llevado a crecer como persona y a descubrir mi verdadero potencial. He enfrentado obstáculos, pero también he encontrado inspiración y determinación en el proceso. Este proyecto es reflejo de mi dedicación y pasión por el conocimiento.

A mi familia: Mario, Concepción, Ángel, Dulce, Flavio, Rafael, Carlos, Guadalupe, Pascual, Néstor, Sofía, Luz, María, Héctor y Carmen; quiero expresar mi profundo amor y gratitud. Su apoyo incondicional y su creencia en mí han sido el combustible que ha impulsado mi éxito. Desde el comienzo, han estado ahí para celebrar mis triunfos y para apoyarme cuando las cosas se tornaban difíciles.

A mis amigos, quiero agradecerles por estar a mi lado en todo momento. Han sido mi sostén emocional, mi motivación y mi fuerza cuando más lo necesitaba. Sus palabras de aliento, su compañía y su confianza en mí han sido fundamentales para mantenerme enfocado y optimista a lo largo de esta travesía académica.

A mis profesores, “rcc DARK” y “frr CNP”, sin sus conocimientos y dedicación al enseñar, no estaría donde estoy ahora. Fomentaron mi curiosidad e interés en el mundo de la programación y la electrónica.

3 Resumen

En la época que vivimos actualmente, el tener un simple sensor conectado a un dispositivo no basta, es necesario poder guardar la información para ejecutar análisis más riguroso, tener estadísticas o comparar la información a futuro. De igual forma el hecho de tener un único dispositivo conectado al sensor es insuficiente, se requiere tener compatibilidad en el entorno de trabajo para poder mover la información de un lado a otro de manera sencilla y eficiente.

La siguiente propuesta, muestra el desarrollo de tres placas para un fonocardiograma, la caracterización de un hule espuma para amortiguar golpes externos, una aplicación móvil para visualizar la información en el celular y una implementación para visualizar la información en el navegador de cualquier dispositivo dentro de la red local.

Nota: En el siguiente repositorio se puede encontrar toda la documentación del proyecto, la cual incluye el código fuente, así como los diagramas eléctricos:

<https://github.com/CloserPilot/Fonocardiograma>

1	DECLARATORIA.....	2
2	DEDICATORIA.....	3
3	RESUMEN	4
4	INTRODUCCIÓN	10
5	ANTECEDENTES	13
6	JUSTIFICACIÓN	15
7	OBJETIVOS.....	16
8	MARCO TEÓRICO	17
8.1	REGULADOR DE VOLTAJE POSITIVO L78L05	17
8.2	REGULADOR DE VOLTAJE POSITIVO LP2985	18
8.3	AMPLIFICADOR DE INSTRUMENTACIÓN AD623.....	19
8.4	MICRÓFONO ELECTRET CMC-60158-42P	20
8.5	MÓDULO MAX9814 DE ADAFRUIT	20
8.6	PRE-AMPLIFICADOR DE MICRÓFONO MAX9814.....	21
8.7	AMPLIFICADOR OPERACIONAL LM324.....	22
8.8	DIODO SCHOTTKY 1N5819	23
8.9	MICROCONTROLADOR ATTINY 85	23
8.10	ARDUINO NANO	24
8.11	MÓDULO BLUETOOTH HC-05 (ZS-040)	25
8.12	MÓDULO EGBT-045MS.....	25
9	DESARROLLO DEL PROYECTO.....	27
9.1	METODOLOGÍA GENERAL.....	27
9.2	Aclaraciones del proyecto.....	30
9.3	CIRCUITO ELECTRÓNICO.....	31
10	MÓDULO A: AMPLIFICADOR DE INSTRUMENTACIÓN AD623.....	32
10.1	FUENTE DE ALIMENTACIÓN	35
10.1.1	<i>Regulador de voltaje L78L05</i>	35
10.1.2	<i>Interruptor e indicador led de encendido</i>	36
10.2	INDICADOR DE BATERÍA BAJA	37
10.2.1	<i>Voltaje en la terminal negativa (Divisor de voltaje 9V).....</i>	37
10.2.2	<i>Voltaje en la terminal positiva (Divisor de voltaje 5V)</i>	39
10.3	PRE-AMPLIFICADOR DEL MICRÓFONO AD623	41
10.3.1	<i>Resistencia de polarización y capacitor de acople del micrófono</i>	42
10.3.2	<i>Etapa de ganancia (Amplificador de Transimpedancia)</i>	43
10.4	VOLTAJE DE REFERENCIA OFFSET	45
10.5	FILTRO PASA BANDA 20Hz-1KHz	47
10.5.1	<i>Filtro pasa altas 20Hz</i>	48
10.5.2	<i>Filtro pasa bajas 1KHz</i>	48
10.6	ETAPA DE GANANCIA	49
10.7	MICROCONTROLADOR ATTINY 85.....	51
10.7.1	<i>Voltaje de referencia ADC</i>	51
10.8	CONEXIONES DEL ATTINY 85.....	53
10.8.1	<i>Conexión al programador.....</i>	53

10.8.2	<i>Conexión al circuito de acondicionamiento.....</i>	54
10.8.3	<i>Conexión al módulo Bluetooth</i>	55
10.9	<i>CONEXIÓN DE TIERRA AL PACIENTE</i>	57
11	MÓDULO B: CONTROL DE GANANCIA AUTOMÁTICO MAX9814	58
11.1	<i>FUENTE DE ALIMENTACIÓN</i>	61
11.1.1	<i>Regulador de voltaje L78L05</i>	61
11.1.2	<i>Interruptor e indicador led de encendido</i>	62
11.2	<i>INDICADOR DE BATERÍA BAJA</i>	63
11.2.1	<i>Voltaje en la terminal negativa (Divisor de voltaje 9V).....</i>	63
11.2.2	<i>Voltaje en la terminal positiva (Divisor de voltaje 5V)</i>	65
11.3	<i>PRE-AMPLIFICADOR DEL MICRÓFONO MAX9814</i>	67
11.3.1	<i>Microphone Bias Resistor.....</i>	67
11.3.2	<i>Bias Capacitor.....</i>	68
11.3.3	<i>Attack Time and Release Time</i>	68
11.3.4	<i>Ajuste de offset.....</i>	69
11.3.5	<i>Power-Supply Bypassing.....</i>	69
11.3.6	<i>Threshold Voltaje</i>	69
11.3.7	<i>Capacitor de acople a la salida.....</i>	71
11.3.8	<i>Ganancia del circuito.....</i>	71
11.3.9	<i>Modificaciones físicas al módulo de Adafruit.....</i>	72
11.4	<i>VOLTAJE DE REFERENCIA OFFSET</i>	73
11.5	<i>FILTRO PASA BANDA 20Hz-1KHz</i>	76
11.5.1	<i>Filtro pasa altas 20Hz</i>	77
11.5.2	<i>Filtro pasa bajas 1KHz</i>	77
11.6	<i>ETAPA DE GANANCIA</i>	78
11.7	<i>MICROCONTROLADOR ATTINY 85.....</i>	80
11.7.1	<i>Voltaje de referencia ADC</i>	80
11.8	<i>CONEXIONES DEL ATTINY 85.....</i>	82
11.8.1	<i>Conexión al programador.....</i>	82
11.8.2	<i>Conexión al circuito de acondicionamiento.....</i>	83
11.8.3	<i>Conexión al módulo Bluetooth</i>	84
11.9	<i>CONEXIÓN DE TIERRA AL PACIENTE</i>	86
12	MÓDULO C: CONTROL DE GANANCIA AUTOMÁTICO MAX9814 Y MÓDULO BLUETOOTH EGBT-045MS	87
12.1	<i>FUENTE DE ALIMENTACIÓN</i>	90
12.1.1	<i>Regulador de voltaje LP2985-33.....</i>	90
12.1.2	<i>Regulador de voltaje LP2985-5.....</i>	91
12.1.3	<i>Interruptor, diodo de protección e indicador led de encendido</i>	91
12.2	<i>INDICADOR DE BATERÍA BAJA</i>	92
12.2.1	<i>Voltaje en la terminal negativa (Divisor de voltaje a 9V).....</i>	92
12.2.2	<i>Voltaje en la terminal positiva (Divisor de voltaje 5V)</i>	94
12.3	<i>PRE-AMPLIFICADOR DEL MICRÓFONO MAX9814</i>	96
12.3.1	<i>Microphone Bias Resistor.....</i>	96
12.3.2	<i>Bias Capacitor.....</i>	97
12.3.3	<i>Attack Time and Release Time</i>	97
12.3.4	<i>Ajuste de offset.....</i>	98
12.3.5	<i>Power-Supply Bypassing.....</i>	98

12.3.6	<i>Threshold Voltaje</i>	98
12.3.7	<i>Capacitor de acople a la salida</i>	100
12.3.8	<i>Ganancia del circuito</i>	100
12.3.9	<i>Modificaciones físicas al módulo de Adafruit</i>	101
12.4	VOLTAJE DE REFERENCIA OFFSET	103
12.5	FILTRO PASA BANDA 20Hz-1KHz	105
12.5.1	<i>Filtro pasa altas 20Hz</i>	106
12.5.2	<i>Filtro pasa bajas 1KHz</i>	106
12.6	ETAPA DE GANANCIA	107
12.7	MÓDULO BLUETOOTH EGBT-045MS.....	109
12.7.1	<i>Pin de transmisión de datos TXD</i>	109
12.7.2	<i>Pin de recepción de datos RXD</i>	109
12.7.3	<i>Pin indicador led de estatus</i>	110
12.7.4	<i>Pulsador de modos AT</i>	110
12.7.5	<i>Conecciones del módulo EGBT-045MS</i>	111
12.8	MICROCONTROLADOR ATTINY 85.....	112
12.8.1	<i>Voltaje de referencia ADC</i>	112
12.9	CONEXIONES DEL ATTINY 85.....	113
12.9.1	<i>Conexión al programador</i>	113
12.9.2	<i>Conexión al circuito de acondicionamiento</i>	114
12.9.3	<i>Conexión al módulo Bluetooth</i>	114
12.10	CONEXIÓN DE TIERRA AL PACIENTE	115
13	ARDUINO COMO ISP	116
14	PROGRAMACIÓN DEL ATTINY85	117
14.1	CONEXIONES DEL ATTINY85	118
14.2	CONEXIONES ENTRE EL ATTINY 85 Y EL ARDUINO NANO.....	119
14.3	CARGA DEL PROGRAMA (ATTINY85)	120
15	PROGRAMACIÓN DEL MÓDULO BLUETOOTH HC-05 / EGBT-045MS	122
15.1	CONEXIONES DEL MÓDULO BLUETOOTH	122
15.2	CONEXIONES ENTRE EL MÓDULO BLUETOOTH Y EL ARDUINO NANO	124
15.3	CARGA DEL PROGRAMA (HC-05/EGBT-045MS)	124
15.3.1	<i>Test de comunicación</i>	126
15.3.2	<i>Cambiar el nombre del módulo HC-05</i>	126
15.3.3	<i>Código de vinculación</i>	126
15.3.4	<i>Velocidad de comunicación</i>	126
16	CONEXIONES ENTRE EL MÓDULO BLUETOOTH EL ATTINY85	127
17	CARACTERIZACIÓN DEL HULE ESPUMA E IMPLEMENTACIÓN DEL ARNÉS.	127
18	APLICACIÓN ANDROID (ESTRUCTURA)	129
18.1	HILOS DE PROGRAMACIÓN	130
18.1.1	<i>Main Activity</i>	130
18.1.2	<i>Thread Data</i>	132
18.1.3	<i>Thread Data Process</i>	133
18.1.4	<i>ThreadHttpServer</i>	134
18.2	WEB_SOCKETHTTP	134
18.3	<u>CONSTANTS</u>	135

18.4	<i>__PLUGINCONTROL</i>	135
18.5	PLUGINS.....	135
18.5.1	<i>myChronometer</i>	135
18.5.2	<i>myPlotter</i>	136
18.5.3	<i>myTimmer</i>	136
18.5.4	<i>myToast</i>	137
18.5.5	<i>myVibrator</i>	137
18.6	INDEX.HTML.....	138
18.7	LAYOUT.....	138
18.7.1	<i>activity_main</i>	138
18.7.2	<i>activity_main (Land)</i>	139
18.7.3	<i>configuration</i>	139
18.7.4	<i>found_devices</i>	140
18.7.5	<i>activity_linked_devices</i>	140
18.7.6	<i>overflow</i>	141
18.8	DRAWABLE.....	142
18.9	VALUES.....	142
18.9.1	<i>Strings</i>	142
18.9.2	<i>Colors</i>	142
18.10	GRADLE SCRIPTS.....	143
18.10.1	<i>GraphView</i>	143
18.10.2	<i>Java-WebSocket:1.5.3</i>	144
18.11	ANDROID MANIFEST	144
19	APLICACIÓN ANDROID.....	145
19.1	FUNCIONAMIENTO DE LA APLICACIÓN MÓVIL	145
19.1.1	<i>Sincronización del circuito del fonocardiograma</i>	145
19.1.2	<i>Creación del archivo</i>	147
19.1.3	<i>Muestreo de la señal</i>	148
19.1.4	<i>Configuración</i>	149
19.1.5	<i>Red local</i>	150
19.2	FUNCIONAMIENTO DE LA APLICACIÓN EN RED LOCAL	151
19.2.1	<i>Muestreo de la señal</i>	151
19.2.2	<i>Configuración</i>	151
19.2.3	<i>Creación del archivo</i>	152
19.3	PARTES DE LA APLICACIÓN MÓVIL	153
19.3.1	<i>Barra de menú</i>	154
19.3.2	<i>Estado de la aplicación</i>	156
19.3.3	<i>Gráfica</i>	157
19.4	PARTES DE LA APLICACIÓN EN RED LOCAL	157
20	RESULTADOS	158
20.1	MÓDULO A: AMPLIFICADOR DE INSTRUMENTACIÓN AD623	159
20.2	MÓDULO B: CONTROL DE GANANCIA AUTOMÁTICO MAX9814	161
20.3	MÓDULO C: CONTROL DE GANANCIA AUTOMÁTICO MAX9814 Y MÓDULO BLUETOOTH EGBT-045MS	163
21	ANÁLISIS DE RESULTADOS	165
21.1	MÓDULO A: AMPLIFICADOR DE INSTRUMENTACIÓN AD623	165
21.2	MÓDULO B: CONTROL DE GANANCIA AUTOMÁTICO MAX9814	165

21.3	MÓDULO C: CONTROL DE GANANCIA AUTOMÁTICO MAX9814 Y MÓDULO BLUETOOTH EGBT-045MS	165
22	CONCLUSIONES	166
22.1	CIRCUITOS DE ACONDICIONAMIENTO	166
22.2	APLICACIÓN MÓVIL.....	167
22.3	APLICACIÓN EN RED LOCAL.....	167
23	BIBLIOGRAFÍA.....	168

4 Introducción

Un estetoscopio es un dispositivo diseñado para la auscultación de sonidos en el pecho u otras partes del cuerpo, por su parte, la auscultación es el procedimiento básico de un análisis clínico para evaluar el estado del paciente, así como sus signos vitales, lo cual constituye el primer indicador para un diagnóstico general. Un médico experimentado es capaz de emitir un diagnóstico preciso con base en la auscultación y auxiliándose de equipo clínico.

El estetoscopio es indispensable en la auscultación ya que amplifica los sonidos del pecho u otras partes del cuerpo. La función del estetoscopio es la de transportar mecánicamente las ondas sonoras hacia el oído del especialista. Este dispositivo no contiene memoria, por lo cual resulta imposible replicar los sonidos del corazón o hacer un análisis de la señal para identificar padecimientos.

Actualmente, se tienen estetoscopios digitales los cuales permiten almacenar y reproducir los sonidos cardiacos, así como identificar padecimientos observando su forma de onda a través de un fonocardiograma. Sin embargo, estos dispositivos están limitados a las especificaciones dadas por el fabricante tanto de hardware como de software ya que no brindan la información técnica necesaria para modificarlos. El hecho de conocer los diagramas eléctricos y el código fuente, permitiría ampliar las funcionalidades del dispositivo, agregando nuevos sensores o maneras de comunicarse como lo es la conectividad bluetooth o wifi; así como el uso de microcontroladores de software libre, propiciando un desarrollo independiente de parte de los usuarios.

Un ejemplo de los parámetros de un estetoscopio digital es el visto en la [Figura 1](#). A pesar de tener un rango de frecuencias de 20Hz a 2KHz situado en los límites inferiores de la audición humana, resulta ser un gran auxiliar en la auscultación cardiaca, ya que se requiere de un oído entrenado para distinguir frecuencias situadas en los límites audibles.

Parameter	Specifications
Frequency range	20 Hz to 2KHz
Smartphone minimum internal storage for recording	500 kB
Smartphone minimum internal storage for app installation	3 MB
Battery Duration in continuous streaming	12 h
Battery Cycle life	> 300 cycles

Figura 1.- Parámetros del estetoscopio digital "eKuore one wireless" [1]

En la [Figura 2](#) se muestran los principales focos de auscultación (pulmonar, aórtica, tricúspide y mitral), en los cuales se pueden distinguir con mayor precisión los distintos tonos de las válvulas cardíacas.

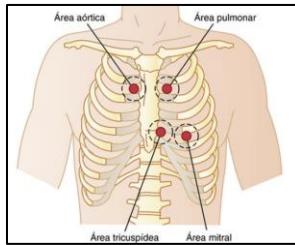


Figura 2.- Puntos focales de la auscultación cardiaca [2]

Los sonidos cardiacos son asociados con el cierre de las válvulas cardíacas. El primer tono cardíaco se debe al cierre de las válvulas auriculoventriculares (mitral y tricúspide) indicando el comienzo de la sístole. El segundo tono corresponde al cierre de las válvulas sigmoideas (pulmonar y aórtica) indicando el comienzo de la diástole. Pueden existir otros tipos de tonos tal como el tercero y cuarto; este tipo de ruidos pueden ser producidos por patologías o fisiologías, por ejemplo, un tercer ruido puede ser visto en la [Figura 3 \(C\)](#) en la insuficiencia mitral. Un cuarto tono cardíaco puede ser visto en [Figura 3 \(B\)](#) en la estenosis aórtica.

Gracias a los datos obtenidos de los ruidos del corazón, es posible identificar anomalías en los sonidos tal como los soplos cardíacos, turbulencias u otros padecimientos ([Figura 3](#)).

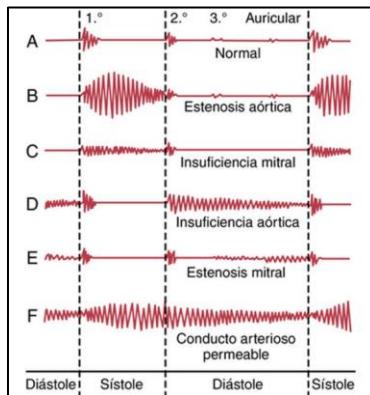


Figura 3.- Fonocardiograma de un corazón normal (A) y otros con padecimientos (B, C, D, E, F,) [2]

Los tonos cardíacos presentan frecuencias entre 1Hz y 1KHz ([Figura 4](#)). A pesar de que el espectro audible del ser humano comprende frecuencias entre 20Hz a 20KH, el rango de audición más sensible se encuentra entre 2KHz a 5KHz [3]. Algunos tonos relacionados con padecimientos, como los soplos cardíacos, ocurren en frecuencias donde el oído humano no es capaz de identificarlos claramente. Es debido a esto, que se hace uso del fonocardiograma para registrar, graficar, filtrar y reproducir las ondas sonoras emitidas de los latidos cardíacos.

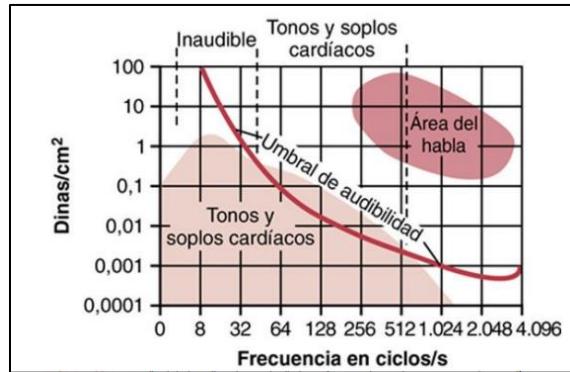


Figura 4.- Amplitud de las vibraciones de distintas frecuencias en los tonos y soplos cardíacos en relación con el umbral de audibilidad [2]

El proyecto propuesto se enfoca en construir un fonocardiógrafo portátil inalámbrico, mediante el acondicionamiento de la señal obtenida a través de un micrófono Electret, la cual a su vez tendrá una etapa de amplificación y una etapa de filtrado en un rango de frecuencias de 20Hz-1KHz. La señal analógica será conectada a un microcontrolador el cual enviará la señal digitalizada vía bluetooth. La visualización de la señal será a través de una aplicación en celular programada en Android Studio, teniendo un dispositivo portátil. Además, se tendrá la opción de vincular la información a la red local permitiendo visualizar los datos en el navegador de cualquier dispositivo.

5 Antecedentes

Existen varios proyectos que hacen uso de estetoscopios digitales para procesar y graficar los datos de la auscultación cardiaca (fonocardiógrafo). Algunos proyectos, crean sus propios estetoscopios digitales desarrollando el circuito de acondicionamiento y adquisición de señales, mientras que otros usan estetoscopios ya fabricados. En ambos casos, se adapta el hardware a una PC o Raspberry para almacenar, procesar y desplegar la información al usuario.

Algunos se describen a continuación:

"Dispositivo para la auscultación de sonidos corporales"

"... el sistema analógico está integrado por un subsistema de adquisición acústica que permite realizar la adquisición de los sonidos corporales para obtener una señal de tipo acústico, que posteriormente es sometida a un subsistema de transducción acústico-electrónico, en el cual se implementa un micrófono y un sistema de acondicionamiento de señal que procesa la señal para poder ser filtrada y amplificada. A continuación, se emplea una tarjeta de desarrollo Arduino UNO para convertir la señal analógica a digital. Esta señal es transmitida a una tarjeta de desarrollo Raspberry Pi 3B donde se implementan los filtros digitales para su visualización de forma gráfica en una interfaz visual que muestre los detalles de la auscultación" [4]

A diferencia del proyecto anterior, el dispositivo será portátil, es decir, el circuito será alimentado con una batería sin necesidad de fuentes externas, así como la parte de procesamiento, almacenaje y despliegue de información será en dispositivos móviles (celular) vía Bluetooth o a través del navegador en red local.

"Diseño y construcción de un fonocardiógrafo digital con visualización en LabVIEW"

"Se presenta el diseño de un fonocardiógrafo con la capacidad de amplificar y filtrar las señales acústicas derivadas del corazón. Posteriormente se realiza una conversión análogo-digital de las señales para ser enviadas por puerto serial a un entorno gráfico en LabVIEW para su registro y análisis en tiempo real." [5]

A diferencia del proyecto anterior, este proyecto utilizará Bluetooth para mandar las señales a un dispositivo móvil o una conectividad en red local. Además, el circuito tendrá integrada toda la circuitería necesaria para tener un dispositivo portátil.

"Diseño e implementación de un sistema para el monitoreo de los sonidos cardiacos"

"En este trabajo se describe el diseño e implementación de un sistema de monitoreo fonocardiográfico capaz de realizar el procesamiento y extracción de las características temporales y espectrales de los sonidos cardiacos en personas sanas por medio de dos técnicas de análisis: la transformada de Fourier en tiempo corto y la transformada wavelet." [6]

El proyecto anterior hace uso de micrófonos capacitivos conectados directamente a una tarjeta de desarrollo DSP y ésta a su vez conectada a la PC. A diferencia del proyecto anterior, este proyecto implementará un circuito de acondicionamiento de señales usando micrófonos Electret, el cual estará conectado a un microcontrolador y las señales serán transmitidas de forma inalámbrica (Bluetooth o red local).

"Diseño e implementación de un fonocardiógrafo digital para el laboratorio de bioelectrónica"

"Este proyecto tiene por objetivo diseñar e implementar un prototipo de un fonocardiógrafo digital para el laboratorio de bioelectrónica, con el propósito de escuchar y visualizar mediante un PC los latidos cardiacos captados mediante una campana de un estetoscopio [...]"

Para visualizar las señales de audio captadas, se implementa un código en Matlab que permite observar las variaciones en tiempo real mostrando amplitud, tiempo y su respectivo espectro de frecuencias normalizado." [7]

El enfoque del proyecto propuesto es enviar los datos a través de conexiones inalámbricas, permitiendo que el paciente tenga la menor cantidad de cables posibles, siendo un dispositivo móvil. De igual forma el software usado para la recepción de información será Open Source a diferencia del trabajo mencionado que utiliza Matlab.

" Diseño y desarrollo de un prototipo de estetoscopio digital"

"El objetivo de este proyecto es el desarrollo de un prototipo de estetoscopio digital priorizando la parte analógica sobre la digital. Se pretende obtener un instrumento con el menor presupuesto." [8]

En el proyecto propuesto se desarrollará un prototipo de fonocardiógrafo, además de implementar una aplicación para su visualización de dispositivos móviles a través de medios inalámbricos a diferencia del proyecto antes citado.

"Estetoscopio electrónico inalámbrico para el registro de los sonidos cardiacos mediante un sensor piezoelectrónico"

" [...] se implementó un circuito basado en un sensor generador (piezoelectrónico), un amplificador de carga y filtros que permitieran detectar [...]. Se programó el Arduino y se configuraron los Xbee's, para realizar transmisiones inalámbricas, gestionando el consumo y, mediante una interfaz gráfica desarrollada en LabView, permitía adquirir y registrar en tiempo real la señal de FCG" [9]

El proyecto anterior crea su propio micrófono haciendo uso de un sensor piezoelectrónico acoplado a una membrana de resina. El uso del piezoelectrónico y de la membrana, permiten un mayor control en la forma de onda que se recibe, ya que se crea el micrófono y se caracterizan los materiales a usar, caso contrario, en un micrófono Electret su membrana se encuentra previamente fabricada, teniendo que trabajar sobre estas especificaciones. Estos micrófonos al contar con su propia membrana y un circuito preamplificador, resultan más sencillos de manipular que un piezoelectrónico. De igual forma la conectividad del circuito usado es Xbee's y LabView mientras que en este proyecto se usará Bluetooth, Android Studio y una conectividad en red local.

6 Justificación

El desarrollo de un fonocardiógrafo permite guardar la información de los sonidos cardiacos, así como su visualización en tiempo real, facilitando las tareas de los especialistas. Ciertas pruebas físicas de esfuerzo requieren que el fonocardiógrafo sea inalámbrico, portátil y no invasivo, debido a los posibles movimientos que la persona bajo prueba pueda realizar.

Los estetoscopios digitales comerciales permiten llevar a cabo una auscultación cardiaca, sin embargo, se tienen limitadas las aplicaciones para el registro de las señales, ya sea por parte del hardware o del software. Estos dispositivos únicamente son compatibles con el software que manejan, además de que algunas licencias requieren comprar el producto primero.

El desarrollar un fonocardiógrafo desde el diseño hasta su implementación en dispositivos móviles o red local, permite una mejora constante y escalabilidad tanto de hardware como de software. Cabe mencionar que el uso de recursos Open Source, proporciona flexibilidad para realizar modificaciones para futuras investigaciones.

El circuito del fonocardiógrafo tendrá toda la electrónica integrada permitiendo que el dispositivo sea portátil, de esta forma, resulta práctico para el paciente usar el dispositivo, ya que no cuenta con cables externos hacia fuentes de alimentación o transmisión de datos.

Con base al tipo de situación de monitoreo, se puede elegir una conectividad vía bluetooth o red local. Al tener un celular conectado vía Bluetooth, es posible registrar la señal en cualquier lugar, ya que se tiene un dispositivo móvil. Por otro lado, al usar la opción red local, se deja la limitante de usar un celular Android para leer la información y es posible visualizar los datos del paciente en cualquier dispositivo conectado en la red local.

7 Objetivos

4.1) General

- Desarrollar un fonocardiógrafo portátil usando una campana de estetoscopio, un micrófono Electret y circuito de acondicionamiento, cuya información pueda ser desplegada en un celular vía bluetooth o en la red local vía wifi.

4.2) Objetivo específico

- Implementar el dispositivo para el acondicionamiento acústico de una señal cardiaca registrada con una campana de estetoscopio.
- Diseñar un circuito electrónico de acondicionamiento para el filtrado y transmisión de la señal eléctrica proveniente de un micrófono Electret.
- Desarrollar una aplicación en Android Studio, para el procesamiento, visualización y almacenamiento del fonocardiograma en un dispositivo Android.
- Integrar un programa complementario a la aplicación del celular que permita visualizar los datos en cualquier navegador dentro de la red local.
- Evaluar el correcto funcionamiento del fonocardiógrafo mediante pruebas de registro de una señal cardiaca tomando como referencia los puntos focales de un sujeto bajo prueba.

8 Marco Teórico

8.1 Regulador de Voltaje positivo L78L05

El circuito integrado L78L05 es un regulador de voltaje positivo de tres terminales el cual cuenta con un limitador de corriente interno de 100mA ([Figura 5](#)). Además, cuenta con una protección contra sobrecalentamientos y no requiere de componentes externos para montarlo.

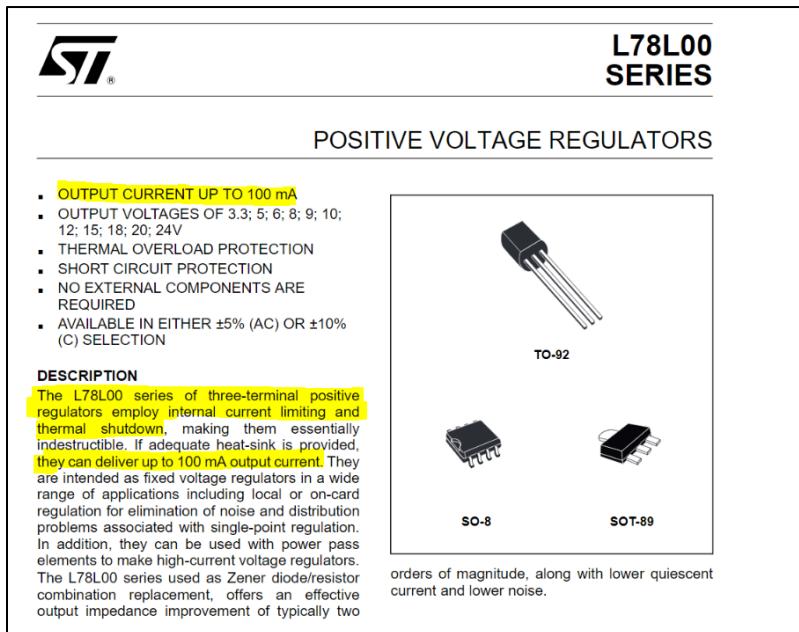


Figura 5.- Corriente máxima, L78L05 [10].

Según su hoja de datos [10], el voltaje máximo admitido en la entrada es de 30V ([Figura 6](#)), mientras que el voltaje mínimo para tener una regulación de 5V es de 1.7V por encima del voltaje visto en su salida ([Figura 7](#)), es decir, 6.7V en la entrada de voltaje.

Conociendo que el voltaje en la entrada debe ser como mínimo 6.7V, se propone dar un margen de voltaje para evitar problemas de desempeño en el regulador, teniendo como voltaje mínimo 7.5V.

L78L00 SERIES				
ABSOLUTE MAXIMUM RATINGS				
Symbol	Parameter ^a		Value	Unit
V _I	DC Input Voltage	V _O = 3.3 to 9 V	30	V
		V _O = 12 to 15 V	35	
		V _O = 18 to 24 V	40	
I _O	Output Current		100	mA
P _{tot}	Power Dissipation		Internally Limited (*)	
T _{stg}	Storage Temperature Range		-40 to 150	°C
T _{op}	Operating Junction Temperature Range	for L78L00C, L78L00AC	0 to 125	°C
		for L78L00AB	-40 to 125	

(*) Our SO-8 package used for Voltage Regulators is modified internally to have pins 2, 3, 6 and 7 electrically connected to the die attach flag. This particular frame decreases the total thermal resistance of the package and increases its ability to dissipate power when an appropriate area of copper on the printed circuit board is available for heat-sinking. The external dimensions are the same as for the standard SO-8.

Figura 6.- Voltaje máximo, L78L05 [10].

L78L00 SERIES						
ELECTRICAL CHARACTERISTICS OF L78L05AB AND L78L05AC						
(refer to the test circuits, $V_I = 10V$, $I_O = 40 mA$, $C_I = 0.33 \mu F$, $C_O = 0.1 \mu F$, $T_J = 0$ to $125^\circ C$ for L78L05AC, $T_J = -40$ to $125^\circ C$ for L78L05AB, unless otherwise specified)						
Symbol	Parameter	Test Conditions	Min.	Typ.	Max.	Unit
V_O	Output Voltage	$T_J = 25^\circ C$	4.8	5	5.2	V
V_O	Output Voltage	$I_O = 1$ to 40 mA $V_I = 7$ to 20 V	4.75		5.25	V
V_O		$I_O = 1$ to 70 mA $V_I = 10$ V	4.75		5.25	
ΔV_O	Line Regulation	$V_I = 7$ to 20 V $T_J = 25^\circ C$			150	mV
ΔV_O		$V_I = 8$ to 20 V $T_J = 25^\circ C$			100	
ΔV_O	Load Regulation	$I_O = 1$ to 100 mA $T_J = 25^\circ C$			60	mV
ΔV_O		$I_O = 1$ to 40 mA $T_J = 25^\circ C$			30	
I_d	Quiescent Current	$T_J = 25^\circ C$			6	mA
I_d		$T_J = 125^\circ C$			5.5	mA
ΔI_d	Quiescent Current Change	$I_O = 1$ to 40 mA			0.1	mA
ΔI_d		$V_I = 8$ to 20 V			1.5	
eN	Output Noise Voltage	$B = 10Hz$ to $100KHz$ $T_J = 25^\circ C$		40		μV
SVR	Supply Voltage Rejection	$V_I = 8$ to 18 V $f = 120Hz$ $I_O = 40$ mA $T_J = 25^\circ C$	41	49		dB
V_d	Dropout Voltage			1.7		V

Figura 7.- Caída de voltaje, L78L05 [10].

8.2 Regulador de voltaje positivo LP2985

Los LP2985 son una familia de reguladores de voltaje positivos de bajo ruido, con una salida máxima de 150mA y baja caída de voltaje. Se tienen diferentes tipos de reguladores que van desde los 2.5V hasta los 16V. Este tipo de reguladores se encuentran en un empaquetado SOT-23 (Figura 8).

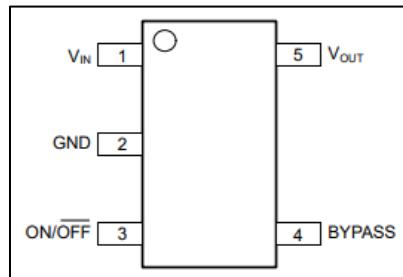


Figura 8.- Empaquetado del LP2985 [11].

Aparte de las típicas terminales de Vin, Vout y GND, cuenta con una terminal de bypass en la cual se puede conectar un capacitor para reducir el nivel de ruido. De igual forma, cuenta con una terminal de ON/OFF para habilitar el LDO.

El voltaje máximo de caída de voltaje en el regulador es de 600mV, teniendo como voltaje de entrada un voltaje igual al voltaje de salida +1V. (Figura 9).

6.5 Electrical Characteristics (continued)								
specified at $T_J = 25^\circ\text{C}$, $V_{IN} = V_{OUT(\text{nom})} + 1.0 \text{ V}$ or $VIN = 2.5 \text{ V}$ (whichever is greater), $I_{OUT} = 1 \text{ mA}$, $V_{ON/OFF} = 2 \text{ V}$, $C_{IN} = 1.0 \mu\text{F}$, and $C_{OUT} = 2.2 \mu\text{F}$ (unless otherwise noted)								
$V_{IN} - V_{OUT}$	Dropout voltage ⁽¹⁾	TEST CONDITIONS			MIN	TYP	MAX	UNIT
		$I_{OUT} = 0 \text{ mA}$			Legacy chip	1	3	mV
		$I_{OUT} = 0 \text{ mA}, -40^\circ\text{C} \leq T_J \leq 125^\circ\text{C}$			New chip	1	2.75	
		$I_{OUT} = 1 \text{ mA}$			Legacy chip	5		
		$I_{OUT} = 1 \text{ mA}, -40^\circ\text{C} \leq T_J \leq 125^\circ\text{C}$			New chip	3		
		$I_{OUT} = 10 \text{ mA}$			Legacy chip	7	10	
		$I_{OUT} = 10 \text{ mA}, -40^\circ\text{C} \leq T_J \leq 125^\circ\text{C}$			New chip	11.5	14	
		$I_{OUT} = 50 \text{ mA}$			Legacy chip	15		
		$I_{OUT} = 50 \text{ mA}, -40^\circ\text{C} \leq T_J \leq 125^\circ\text{C}$			New chip	40	60	
		$I_{OUT} = 150 \text{ mA}$			Legacy chip	98	115	
		$I_{OUT} = 150 \text{ mA}, -40^\circ\text{C} \leq T_J \leq 125^\circ\text{C}$			New chip	90		
		$I_{OUT} = 150 \text{ mA}, -40^\circ\text{C} \leq T_J \leq 125^\circ\text{C}$			Legacy chip	120	150	
		$I_{OUT} = 150 \text{ mA}, -40^\circ\text{C} \leq T_J \leq 125^\circ\text{C}$			New chip	120	145	
		$I_{OUT} = 250 \text{ mA}$			Legacy chip	225		
		$I_{OUT} = 250 \text{ mA}, -40^\circ\text{C} \leq T_J \leq 125^\circ\text{C}$			New chip	184		
		$I_{OUT} = 350 \text{ mA}$			Legacy chip	280	350	
		$I_{OUT} = 350 \text{ mA}, -40^\circ\text{C} \leq T_J \leq 125^\circ\text{C}$			New chip	180	198	
		$I_{OUT} = 575 \text{ mA}$			Legacy chip	575		
		$I_{OUT} = 575 \text{ mA}, -40^\circ\text{C} \leq T_J \leq 125^\circ\text{C}$			New chip	254		

Figura 9.- Caída de voltaje del LP2985 [11].

8.3 Amplificador de instrumentación AD623

El AD623 es un amplificador de instrumentación que puede ser alimentado con voltaje unipolar o bipolar, logrando entregar voltajes rail-to-rail en su salida. Este circuito integrado cuenta con ganancia al conectar una resistencia entre dos de sus terminales, ganancia que va desde 1 hasta 1000 unidades. Además, posee un alto CRMM, es un componente de bajo consumo energético y ofrece estabilidad ante la temperatura. El tipo de empaquetado con el que se encuentra este circuito integrado es PDIP, SOIC_N o MSOP.

Según su hoja de datos [12], los rangos de voltaje de entrada (Common-mode input voltage range), son desde la fuente de alimentación negativa “ V^- ” +0.15V, hasta la fuente de alimentación positiva “ V^+ ” -1.5V (Figura 10).

Data Sheet AD623											
Parameter	Test Conditions/Comments	AD623ANZ, AD623ARZ			AD623ARM			AD623BNZ, AD623BRZ			Unit
INPUT											
Input Impedance		2 2		2 2	2 2		2 2	2 2		2 2	
Differential Common-Mode		2 2		2 2	2 2		2 2	2 2		2 2	
Input Voltage Range ⁽¹⁾	$V_s = 3 \text{ V}$ to 12 V	$(-V_s) = 0.15$	$(+V_s) = 1.5$	$(-V_s) = 0.15$	$(+V_s) = 1.5$	$(-V_s) = 0.15$	$(+V_s) = 1.5$	$(-V_s) = 0.15$	$(+V_s) = 1.5$	$(-V_s) = 0.15$	$(+V_s) = 1.5$
Common-Mode Rejection at 60 Hz with $1 \text{ k}\Omega$ Source Imbalance											
$G = 1$	Common-mode voltage (V_{CM}) = 0 V to 3 V	70	80	70	80	77	86	77	86	77	86
$G = 10$	$V_{CM} = 0 \text{ V}$ to 3 V	90	100	90	100	94	100	94	100	94	100
$G = 100$	$V_{CM} = 0 \text{ V}$ to 3 V	105	110	105	110	105	110	105	110	105	110
$G = 1000$	$V_{CM} = 0 \text{ V}$ to 3 V	105	110	105	110	105	110	105	110	105	110
OUTPUT											
Output Swing	$R_L = 10 \text{ k}\Omega$	0.2	$(+V_s) = 0.2$	0.2	$(+V_s) = 0.2$	0.2	$(+V_s) = 0.2$	0.2	$(+V_s) = 0.2$	0.2	$(+V_s) = 0.2$
	$R_L = 100 \text{ k}\Omega$	0.05	$(+V_s) = 0.05$	0.05	$(+V_s) = 0.05$	0.05	$(+V_s) = 0.05$	0.05	$(+V_s) = 0.05$	0.05	$(+V_s) = 0.05$

Figura 10.- Common-mode voltage range y Voltage output swing, AD623 [12].

8.4 Micrófono Electret CMC-60158-42P

El micrófono Electret usado en el circuito es el modelo CMC-60158-42P, este micrófono fue elegido debido a sus dimensiones siendo lo suficientemente pequeño para ingresar dentro de la campana del estetoscopio (*Figura 11*).

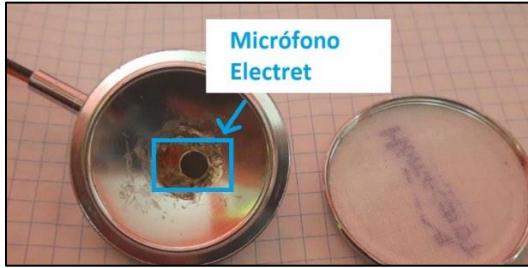


Figura 11.- Campana del estetoscopio y micrófono Electret.

Tomando los parámetros de su hoja de datos [13], se tiene la siguiente tabla:

Tabla 1.-Parámetros del micrófono Electret CMC-6012-42P [13]

Parámetro	Valor	Unidades
Sensibilidad	-42+-3	dBV
Voltaje de operación	2	Vdc
Consumo de corriente (máximo)	0.4	mA
Impedancia	2.2	KΩ
SNR (min)	58	dB

8.5 Módulo MAX9814 de Adafruit

El módulo de Adafruit cuenta con una MAX9814 el cual viene configurado dentro de la placa para trabajar con su propio micrófono Electret de 20-20KHz.

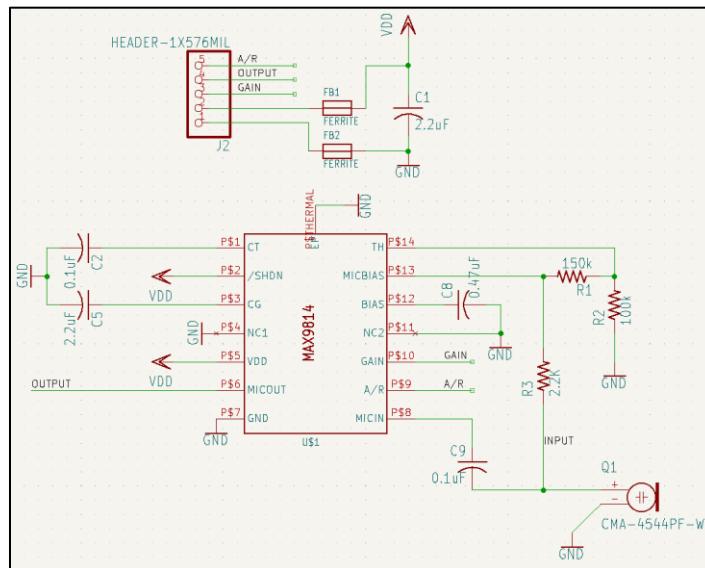


Figura 12.- Esquemático del módulo MAX9814 de Adafruit [14].

8.6 Pre-amplificador de micrófono MAX9814

El MAX9814 es un amplificador de micrófono de bajo costo con un control de ganancia automático. Cuenta con una ganancia seleccionable de 40dB, 50dB o 60dB en un uso de funcionamiento típico. A través de un divisor de voltaje externo es posible ajustar el control de ganancia automática y un solo capacitor es el encargado de ajustar los tiempos del Attack/Release time. El generador de voltaje para micrófono de bajo ruido puede polarizar casi cualquier micrófono Electret. El voltaje de trabajo del MAX9814 es de 2.7V hasta 5.5V.

Un diagrama eléctrico simplificado es presentado a continuación (*Figura 13*).

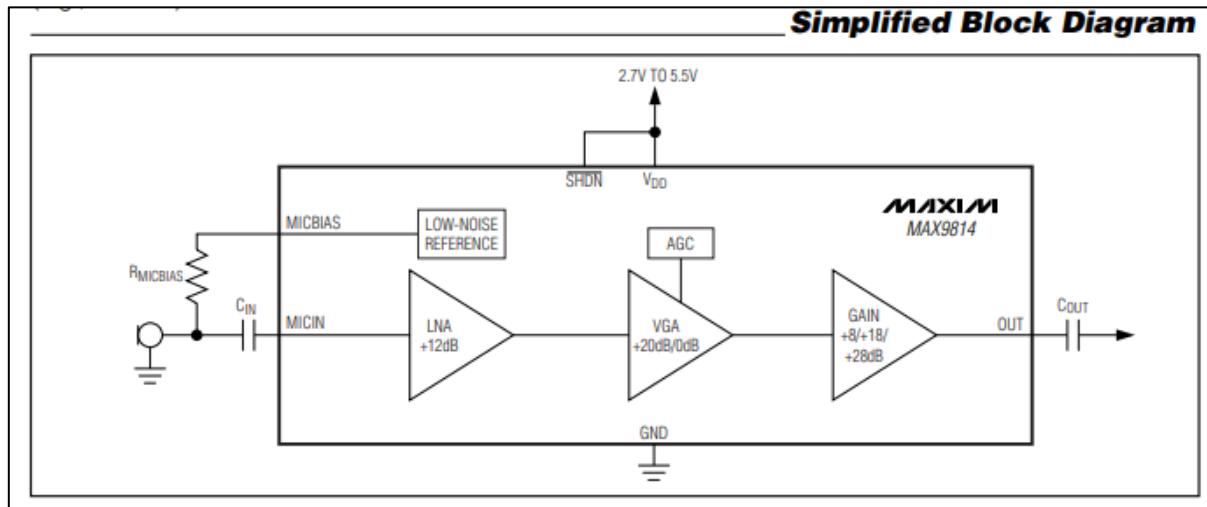


Figura 13.- Diagrama simplificado MAX9814 [15].

El empaquetado en el que se presenta este chip es el TDFN

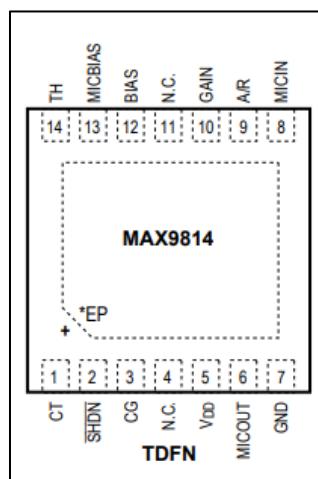


Figura 14.- Empaquetado TDFN del MAX9814 [15].

8.7 Amplificador operacional LM324

Según su hoja de datos [16], los rangos de voltaje de entrada (Common-mode input voltage range) son desde la fuente de alimentación negativa “V-”, hasta la fuente de alimentación positiva “V+”-2V (*Figura 15*). Sin embargo, los rangos de voltaje en la salida vienen dados por “V+ -1.75V” y por “V- +1V” (*Figura 16*).

6.3 Recommended Operating Conditions							
over operating free-air temperature range (unless otherwise noted)							
	LM324B, LM324BA, LM2902B, LM2902BA		LM2902		LM324xx, LM224xx, LM2902xxx, LM124x		UNIT
	MIN	MAX	MIN	MAX	MIN	MAX	
V _{CC} Supply voltage	3	36	3	26	3	30	V
V _{CM} Common-mode voltage	0	V _{CC} - 2	0	V _{CC} - 2	0	V _{CC} - 2	V

Figura 15.- Common-mode voltaje range LM324 [11]

OUTPUT							
V _O			I _{OUT} = -50 µA		1.35	1.5	V
V _O		Positive Rail (V+)	I _{OUT} = -1 mA		1.4	1.6	V
V _O			I _{OUT} = -5 mA		1.5	1.75	V
V _O	Voltage output swing from rail		I _{OUT} = 50 µA		100	150	mV
V _O		Negative Rail (V-)	I _{OUT} = 1 mA		0.75	1	V
V _O			V _S = 5 V, RL ≤ 10 kΩ connected to (V-)	T _A = -40°C to 85°C	5	20	mV
I _O					-20 ⁽¹⁾	-30	mA
		V _S = 15 V; V _O = V ₋ ; V _{ID} = 1 V	Source	T _A = -40°C to 85°C	-10 ⁽¹⁾		mA
		V _S = 15 V; V _O = V ₊ ; V _{ID} = -1 V	Sink		10 ⁽¹⁾	20	mA
		V _{ID} = -1 V; V _O = (V ₋) + 200 mV		T _A = -40°C to 85°C	5 ⁽¹⁾		mA
I _{SC}	Short-circuit current	V _S = 20 V, (V ₊) = 10 V, (V ₋) = -10 V, V _O = 0 V			50	85	µA
C _{LOAD}	Capacitive load drive					100	pF

Figura 16.- Voltaje output swing from rail LM324 [11]

Teniendo los rangos de voltaje del LM324 y dado que el circuito utiliza una alimentación unipolar de 5V, se proponen los siguientes voltajes de operación en el circuito:

- Límite inferior = V- +1V = 1V
- Límite superior = +V - 2V = 3V

8.8 Diodo Schottky 1N5819

El diodo 1N5819 es un diodo de bajo consumo y alta eficiencia. Su uso principal es para aplicaciones donde se necesite un diodo con bajo consumo de voltaje, inversores a alta frecuencia y protección de polaridad.

Según su hoja de datos [17], el diodo 1N5819 puede soportar como máximo 28V reverse RMS con una corriente de 1A, además cuenta con un voltaje de encendido de 0.6V ([Figura 17](#)).

Maximum Ratings and Electrical Characteristics (@ $T_A = +25^\circ\text{C}$, unless otherwise specified.)					
Single phase, half wave, 60Hz, resistive or inductive load. For capacitive load, derate current by 20%.					
Characteristic	Symbol	1N5817	1N5818	1N5819	Unit
Peak Repetitive Reverse Voltage Working Peak Reverse Voltage DC Blocking Voltage	V _{RRM} V _{RWM} V _R	20	30	40	V
RMS Reverse Voltage	V _{R(RMS)}	14	21	28	V
Average Rectified Output Current (Note 4) @ $T_L = +90^\circ\text{C}$	I _O	1.0			A
Non-Repetitive Peak Forward Surge Current 8.3ms Single Half Sine-Wave Superimposed on Rated Load	I _{FSM}	25			A
Forward Voltage (Note 5) @ I _F = 1.0A @ I _F = 3.0A	V _{FM}	0.450 0.750	0.550 0.875	0.60 0.90	V
Peak Reverse Leakage Current @ $T_A = +25^\circ\text{C}$ at Rated DC Blocking Voltage (Note 5) @ $T_A = +100^\circ\text{C}$	I _{RM}	1.0 10			mA
Typical Total Capacitance (Note 6)	C _T	110			pF
Typical Thermal Resistance Junction to Lead (Note 7)	R _{θJL}	15			°C/W
Typical Thermal Resistance Junction to Ambient	R _{θJA}	50			
Operating and Storage Temperature Range	T _{J, T_{STG}}	-65 to +125			°C

Figura 17.- Voltajes del 1N5819 [17], (módulo C).

8.9 Microcontrolador Attiny 85

El Attiny 85 es un microcontrolador de 8 bits de bajo consumo energético, cuenta con 6 pines bidireccionales programables con resistencias pull-up internas, cada uno de estos pines cuenta con características especiales en caso de requerirlo.

Los voltajes de alimentación del microcontrolador son 4.5V a 5.5V. En la [Figura 18](#), se pueden observar los voltajes de trabajo para las terminales del attiny85, conociendo que la fuente de alimentación el proyecto es de 5V, se tienen los siguientes rangos de voltaje:

- Entrada baja -> [-0.5 , 1.5]V
- Entrada alta -> [3.0 , 5.5]V
- Salida baja -> [0.0 , 0.6]V
- Salida alta -> [4.3 , 5.0]V

21.2 DC Characteristics

Table 21-1. DC Characteristics. $T_A = -40^{\circ}\text{C}$ to $+85^{\circ}\text{C}$

Symbol	Parameter	Condition	Min.	Typ. ⁽¹⁾	Max.	Units
V_{IL}	Input Low-voltage, except XTAL1 and RESET pin	$V_{CC} = 1.8\text{V} - 2.4\text{V}$ $V_{CC} = 2.4\text{V} - 5.5\text{V}$	-0.5 -0.5		$0.2V_{CC}^{(3)}$ $0.3V_{CC}^{(3)}$	V
V_{IH}	Input High-voltage, except XTAL1 and RESET pin	$V_{CC} = 1.8\text{V} - 2.4\text{V}$ $V_{CC} = 2.4\text{V} - 5.5\text{V}$	$0.7V_{CC}^{(2)}$ $0.6V_{CC}^{(2)}$		$V_{CC} + 0.5$ $V_{CC} + 0.5$	V
V_{IL1}	Input Low-voltage, XTAL1 pin, External Clock Selected	$V_{CC} = 1.8\text{V} - 5.5\text{V}$	-0.5		$0.1V_{CC}^{(3)}$	V
V_{IH1}	Input High-voltage, XTAL1 pin, External Clock Selected	$V_{CC} = 1.8\text{V} - 2.4\text{V}$ $V_{CC} = 2.4\text{V} - 5.5\text{V}$	$0.8V_{CC}^{(2)}$ $0.7V_{CC}^{(2)}$		$V_{CC} + 0.5$ $V_{CC} + 0.5$	V
V_{IL2}	Input Low-voltage, RESET pin	$V_{CC} = 1.8\text{V} - 5.5\text{V}$	-0.5		$0.2V_{CC}^{(3)}$	V
V_{IH2}	Input High-voltage, RESET pin	$V_{CC} = 1.8\text{V} - 5.5\text{V}$	$0.9V_{CC}^{(2)}$		$V_{CC} + 0.5$	V
V_{IL3}	Input Low-voltage, RESET pin as I/O	$V_{CC} = 1.8\text{V} - 2.4\text{V}$ $V_{CC} = 2.4\text{V} - 5.5\text{V}$	-0.5 -0.5		$0.2V_{CC}^{(3)}$ $0.3V_{CC}^{(3)}$	V
V_{IH3}	Input High-voltage, RESET pin as I/O	$V_{CC} = 1.8\text{V} - 2.4\text{V}$ $V_{CC} = 2.4\text{V} - 5.5\text{V}$	$0.7V_{CC}^{(2)}$ $0.6V_{CC}^{(2)}$		$V_{CC} + 0.5$ $V_{CC} + 0.5$	V
V_{OL}	Output Low-voltage, ⁽⁴⁾ Port B (except RESET) ⁽⁶⁾	$I_{OL} = 10\text{ mA}$, $V_{CC} = 5\text{V}$ $I_{OL} = 5\text{ mA}$, $V_{CC} = 3\text{V}$			0.6 0.5	V
V_{OH}	Output High-voltage ⁽⁵⁾ Port B (except RESET) ⁽⁶⁾	$I_{OH} = -10\text{ mA}$, $V_{CC} = 5\text{V}$ $I_{OH} = -5\text{ mA}$, $V_{CC} = 3\text{V}$	4.3 2.5			V

Figura 18.- Rangos de voltaje de entrada y salida del attiny85.

Cuenta con un convertidor analógico digital de 10 bits y diferentes voltajes de referencia hacia el ADC siendo Vcc, 1.1V, 2.56V o un voltaje externo suministrado en el pin AREF ([Figura 19](#)).

17.6.2 ADC Voltage Reference

The voltage reference for the ADC (V_{REF}) indicates the conversion range for the ADC. Single ended channels that exceed V_{REF} will result in codes close to 0x3FF. V_{REF} can be selected as either V_{CC} , or internal 1.1V / 2.56V voltage reference, or external AREF pin. The first ADC conversion result after switching voltage reference source may be inaccurate, and the user is advised to discard this result.

Figura 19.- Voltaje de referencia del ADC, attiny85 [18], (módulo A).

8.10 Arduino Nano

Placa de desarrollo fácilmente programable a través un puerto USB de la computadora y el IDE de Arduino. Cuenta con un ATmega323, regulador de voltaje a 5V, indicadores led para la transmisión y recepción de datos del puerto serial, un botón de Reset y un convertidor TTL para comunicar la placa de desarrollo con el USB de la computadora. Posee 22 pines digitales programables con una salida máxima de 40mA por cada pin.

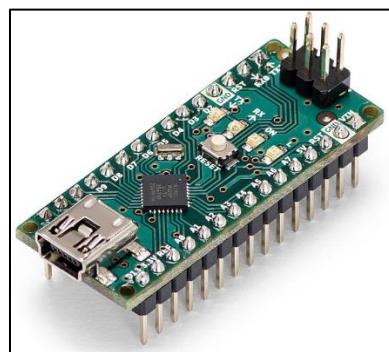


Figura 20.- Arduino Nano.

8.11 Módulo bluetooth HC-05 (ZS-040)

Este módulo implementa el software y hardware necesario para recibir y enviar información vía Bluetooth, la información es entregada a través de dos pines, uno de transmisión y otro de recepción de datos. El módulo está integrado por el chip EGBT-045MS el cual se encarga del protocolo Bluetooth y una placa de desarrollo la cual reduce el voltaje de trabajo de 5V a 3.3V para alimentar el chip, además cuenta con un botón para elegir el modo de emparejamiento siendo el Modo AT1 y AT2.

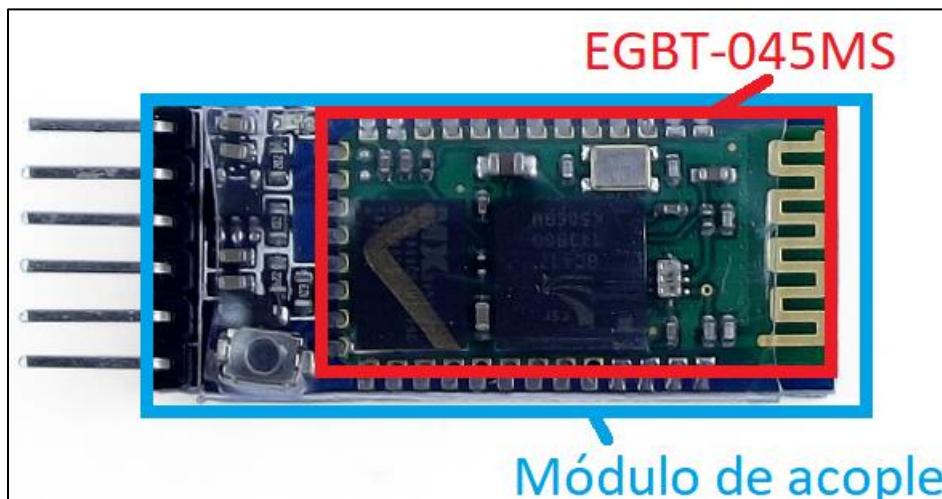


Figura 21.- Módulo HC-05

Cuenta con un led conectado en el pin 31 y 24 del EGBT-045MS/EGBT-046S el cual indica el estado de la tarjeta, al estar encontrarse en estos pines es posible usar la misma tarjeta para su versión HC-05 y HC-06.

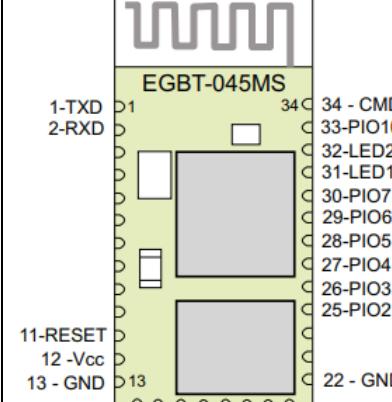
Tiene un pulsador conectado a 3.3V en el pin 34 usado para entrar en los diferentes modos AT al momento de energizar la tarjeta.

Uno de los pines que tiene disponible es el pin de STATUS, el cual está conectado el pin 32. Este pin se encuentra en estado bajo cuando el módulo no está conectado y en estado alto cuando esta energizado.

El pin ENABLE deshabilita el chip al ponerlo en estado bajo.

8.12 Módulo EGBT-045MS

Es el módulo en el cual se basa el protocolo bluetooth del HC-05, cuenta con pines de transmisión, recepción de datos, reset, indicadores led, puerto de comandos, así como puertos para ser programados por el usuario ([Figura 22](#)). El módulo se encuentra programado con firmware para una comunicación UART inalámbrica. El EGBT-045MS puede ser programado como maestro o como esclavo a través de comandos AT.



PIN	ID	DESCRIPTION
1	TXD	UART TXD Output
2	RXD	UART RXD Input
11	RESET	RESET Input
12	Vcc	+3.1 to 4.2VDC Power Input
13	GND	Common Ground
22	GND	Common Ground
25	PIO2	User programmable I/O
26	PIO3	User programmable I/O
27	PIO4	User programmable I/O
28	PIO5	User programmable I/O
29	PIO6	User programmable I/O
30	PIO7	User programmable I/O
33	PIO10	User programmable I/O
31	LED1	LED Status Indicator
32	LED2	LED Status Indicator
34	CMD	Command Mode

Figura 22.- Pinout EGBT-045MS [19].

Este módulo viene preconfigurado como esclavo teniendo la siguiente configuración:

- Baud Rate: 38400/9600 bps
- Tamaño de información: 8 bits
- Stop Bits: 1 bit
- Paridad: Ninguna
- Contraseña: 1234
- Nombre del dispositivo: HC-05

Los voltajes de alimentación van desde 3.1V hasta 4.2V, sin embargo, el voltaje de trabajo de cada uno de sus pines va desde 0V hasta 3.3V.

9 Desarrollo del proyecto

9.1 Metodología general

A continuación, se presenta un diagrama a bloques del sistema ([Figura 23](#))

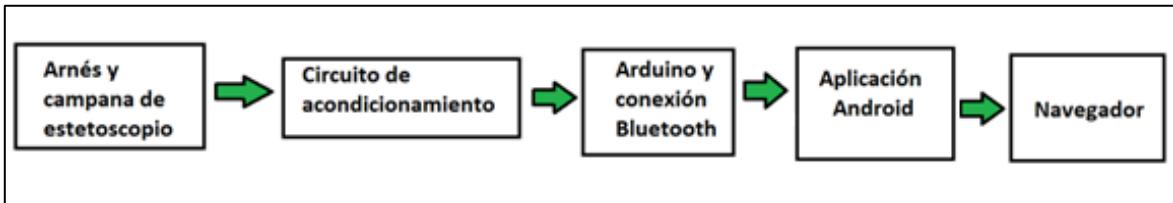


Figura 23.- Diagrama a bloques del sistema

Para la parte física y mecánica del prototipo, se usará un chaleco sujetador de velcro, que ejerza presión sobre el pecho del paciente y que contenga un orificio apuntando al corazón. Al interior de este orificio, se introducirá una campana de estetoscopio con un micrófono Electret cubierto por hule espuma, con lo cual se detectarán los sonidos emitidos por el paciente. Cabe mencionar que el hule espuma será caracterizado acústicamente mediante un tubo de impedancias para determinar sus coeficientes acústicos y evaluar que la espuma no presente un coeficiente de absorción alto para las frecuencias de interés. El estetoscopio se encargará de amplificar y guiar la señal acústica, el hule espuma filtrará cierto rango de frecuencias y el micrófono será el transductor encargado de convertir de manera proporcional la variación de presión en una señal eléctrica.

Se tienen tres prototipos para el circuito electrónico. Una placa utiliza un amplificador de instrumentación AD623 para el circuito de pre-amplificación ([10 Módulo A: Amplificador de instrumentación AD623](#)), otra un control de ganancia automática con el módulo MAX9814 de Adafruit ([11 Módulo B: Control de ganancia automática MAX9814](#)) y por último, la tercera placa presenta un funcionamiento similar con el MAX9814 con la diferencia de usar reguladores de voltaje más eficientes y tener una mejor distribución de componentes ([12 Módulo C: Control de ganancia automática MAX9814 y módulo Bluetooth EGBT-045MS](#)). Los circuitos de acondicionamiento de las placas pueden ser divididos en las siguientes etapas:

- Pre-amplificación: la señal registrada por el micrófono Electret se hará pasar por un pre-amplificador seguido, ya sea, de un control de ganancia automática MAX9814 el cual permita atenuar las señales mayores a cierto umbral de voltaje para trabajar solo con las señales de interés, o de un amplificador de instrumentación AD623 [12] el cual cuenta con una alta impedancia de entrada y alta relación de rechazo de modo común para garantizar una alta relación señal a ruido.
- Filtrado: esta etapa estará integrada por amplificadores operacionales con el propósito de filtrar la señal. Para ello, se diseñará un filtro mediante el software "Filter pro" con configuración Butterworth - Sally Key de segundo orden, tratando de mantener una salida constante hasta su frecuencia de corte, teniendo un rango de frecuencias de 20Hz a 1KHz, esto con el propósito de eliminar el ruido de frecuencias no deseadas.

- Amplificación: la señal filtrada se pasará por una etapa de ganancia para ajustar su amplitud de manera manual, de igual forma, se agregará una señal de DC para leer la señal en el microcontrolador.

Las etapas del circuito de acondicionamiento pueden ser vistas en la [Figura 24](#).

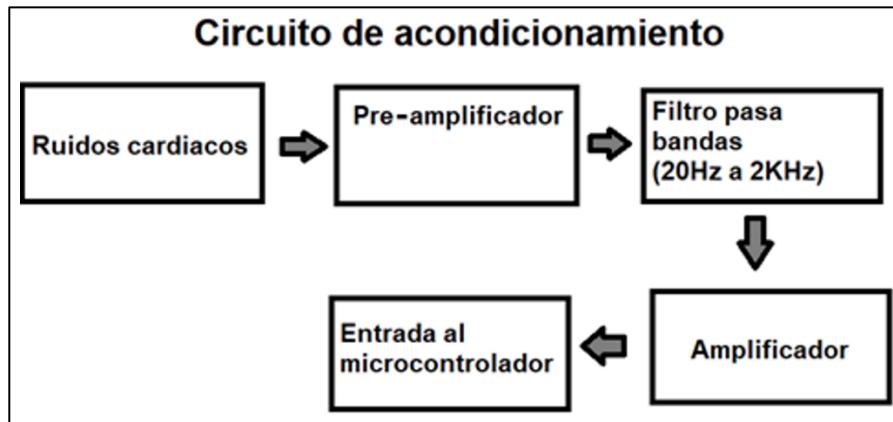


Figura 24.- Etapas del circuito de acondicionamiento

Además del circuito de acondicionamiento, se contará con una salida de tierra conectada al paciente para eliminar las interferencias que se puedan producir.

El circuito de acondicionamiento será conectado al microcontrolador (Attiny85), el cual enviará la información vía Bluetooth usando el protocolo RFCOMM. La aplicación del celular se encargará de recibir la información del microcontrolador, guardarla en un archivo de texto y desplegar una gráfica con los datos recibidos en tiempo real. La aplicación será implementada en el IDE de Android Studio en java. ([Figura 25](#)).

Por otro lado, se tendrá la opción de conectar la aplicación de celular de la red local. El celular funcionará como servidor recibiendo las peticiones HTTP mandadas por cualquier navegador. Al aceptar una conexión, se mandará un HTML en el cual viene la información necesaria para conectarse vía WebSocket y recibir la señal cardiaca. Además, dentro del HTML se encuentra un código que permite graficar la información recibida y guardarla en un archivo de texto. El WebSocket cuenta con multicast permitiendo mandar la información a diferentes clientes de forma simultánea. Cabe destacar que cada cliente realiza su propio procesamiento de datos por lo cual cada uno es independiente de otro.

El prototipo ensamblado contará con un filtro mecánico siendo el hule espuma, esto con el fin de evitar ruido generado por el roce de la superficie del micrófono.

Los voltajes de trabajo se encontrarán limitados por los voltajes máximos y mínimos que los amplificadores operacionales utilizados en el circuito, para una mayor referencia consulte la sección [8.7 Amplificador operacional LM324](#).

Se realizarán pruebas de conectividad, transmisión y recepción sobre ambos tipos de conexiones (Bluetooth y wifi), enviando diferentes paquetes de información previamente cargados para verificar la funcionalidad.

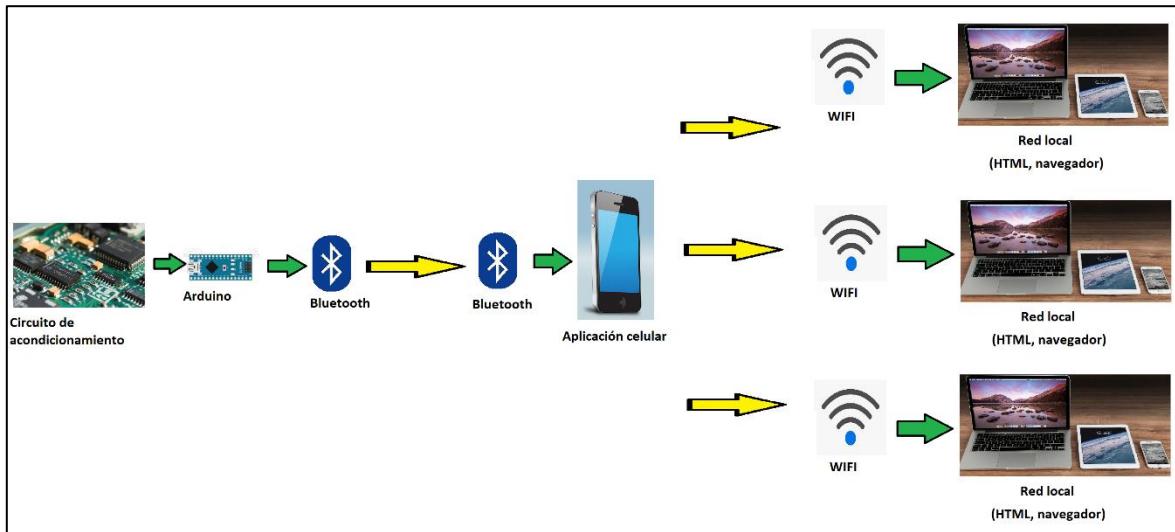


Figura 25.- Conexión del dispositivo.

9.2 Aclaraciones del proyecto

El proyecto "**Desarrollo de un fonocardiógrafo portátil inalámbrico conectado a dispositivos dentro de la red local y dispositivos móviles**" toma como base diferentes notas de aplicación de fabricantes, así como software de uso libre para diseñar el circuito y los programas utilizados en el desarrollo del proyecto.

El proyecto es un prototipo, por lo cual no cuenta con ninguna certificación de grado médico para ser usado o replicado. En caso de ser replicado, se recomienda usar componentes especiales (grado médico).

En el siguiente repositorio se puede encontrar toda la documentación del proyecto, la cual incluye el código fuente, así como los diagramas eléctricos:

<https://github.com/CloserPilot/Fonocardiograma>

9.3 Circuito electrónico

Para el desarrollo de este proyecto se presentarán tres módulos los cuales pueden dividirse en las siguientes secciones:

- Fuente de alimentación
- Indicador de batería baja
- Circuito de acondicionamiento
 - Pre-amplificador del micrófono
 - Voltaje de referencia offset
 - Filtro pasa banda 20Hz-1KHz
 - Etapa de ganancia
- Microcontrolador attiny85
 - Voltaje de referencia ADC
- Módulo Bluetooth HC-05/EGBT-045MS
- Tierra al paciente

Las diferencias principales entre los módulos son presentadas a continuación:

- Módulo A: Amplificador de instrumentación AD623.
Siendo el primer módulo implementado en el circuito, cuenta con una etapa de pre-amplificación lineal (AD623). A pesar de contar con una etapa de filtrado en el circuito de acondicionamiento, los golpes externos cuya frecuencia se encuentre en el rango de interés, son amplificados, debido a esto, gran parte del ruido externo se encuentra en la señal de salida.
- Módulo B: Control de ganancia automático MAX9814.
Debido a los problemas de ruido, se buscó otra alternativa para el pre-amplificador. Se hace uso del MAX9814 el cual es un control de ganancia automático. En caso de presentar un golpe externo en la campana del estetoscopio, esta señal ruidosa contará con mayor amplitud en comparación con la señal cardiaca; conociendo los límites de la señal de interés, es posible atenuar exponencialmente aquellas señales que superen cierto umbral y amplificar aquellas de menor amplitud. La señal de salida seguirá teniendo parte del ruido externo, pero éste será más controlado.
- Módulo C: Control de ganancia automático MAX9814 y módulo Bluetooth EGBT-045MS
Este circuito toma como base el diseño del Módulo B teniendo como pre-amplificador el MAX9814. En lugar de usar el módulo Bluetooth HC-05, se utiliza directamente el EGBT-045MS permitiendo reacomodar los componentes en la misma placa reduciendo el tamaño y mejorando la estética del circuito. Se integran reguladores de voltaje con menor caída de tensión, así como un diodo de protección en caso de invertir la fuente de alimentación.

Nota: Para una mejor estructura del reporte, se decidió dedicar una sección por cada placa. Algunas secciones del reporte pueden repetirse ya que las placas presentan grandes similitudes.

10 Módulo A: Amplificador de instrumentación AD623

El módulo A ([Figura 26](#), [Figura 27](#) y [Figura 28](#)) cuenta con las siguientes secciones:

- 10.1 Fuente de alimentación
 - 10.1.1 Regulador de voltaje L78L05
 - 10.1.2 Interruptor e indicador led de encendido
- 10.2 Indicador de batería baja
 - 10.2.1 Voltaje en la terminal negativa (Divisor de voltaje 9V)
 - 10.2.2 Voltaje en la terminal positiva (Divisor de voltaje 5V)
- 10.3 Pre-amplificador del micrófono AD623
 - 10.3.1 Resistencia de polarización y capacitor de acople del micrófono
 - 10.3.2 Etapa de ganancia (Amplificador de Transimpedancia)
- 10.4 Voltaje de referencia offset
- 10.5 Filtro pasa banda 20Hz-1KHz
 - 10.5.1 Filtro pasa altas 20Hz
 - 10.5.2 Filtro pasa bajas 1KHz
- 10.6 Etapa de ganancia
- 10.7 Microcontrolador attiny 85
 - 10.7.1 Voltaje de referencia ADC
- 10.8 Conexiones del attiny 85
 - 10.8.1 Conexión al programador
 - 10.8.2 Conexión al circuito de acondicionamiento
 - 10.8.3 Conexión al módulo Bluetooth
- 10.9 Conexión de tierra al paciente

Nota: Para la programación de la placa, ver la sección: [13 Arduino como ISP](#), [14 Programación del attiny85](#) y [15 Programación del módulo Bluetooth HC-05 / EGBT-045MS](#)

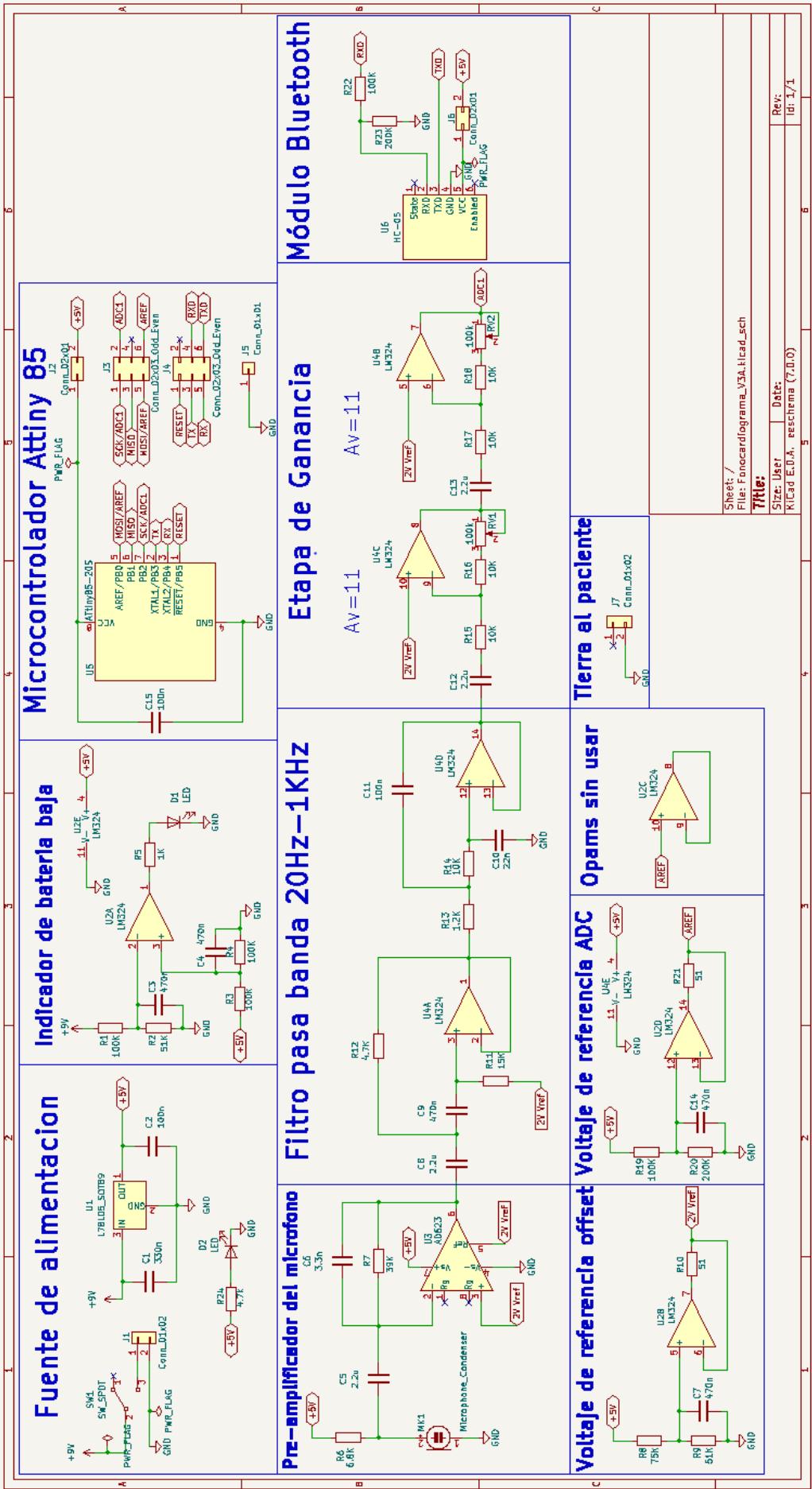


Figura 26.- Diagrama eléctrico, (módulo A).

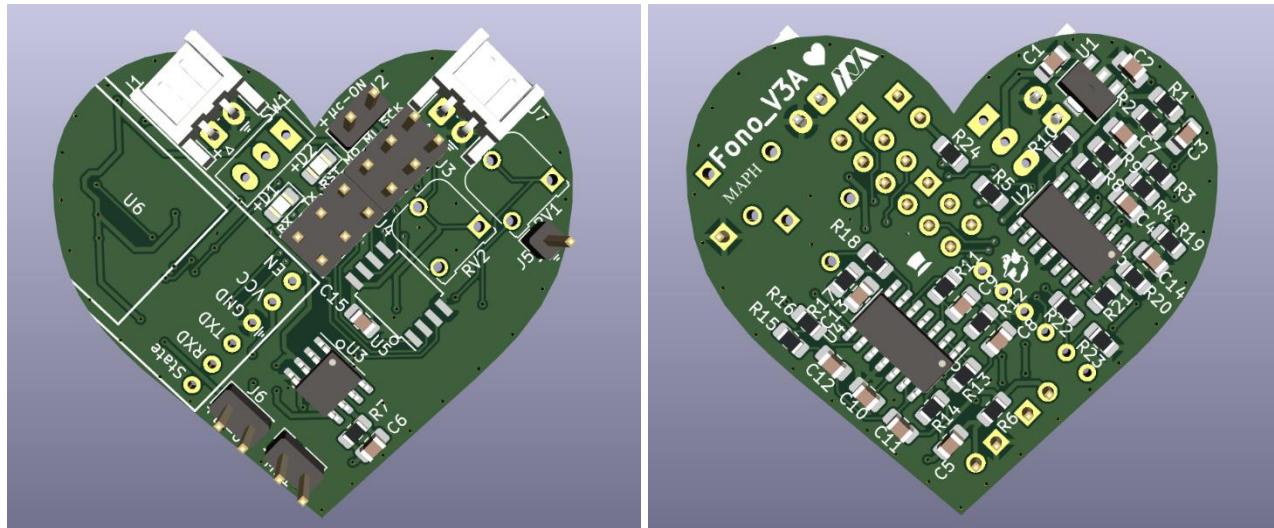


Figura 27.- Izquierda: Modelo 3D de la placa, vista frontal (módulo A).
Derecha: Modelo 3D de la placa, vista trasera (módulo A).

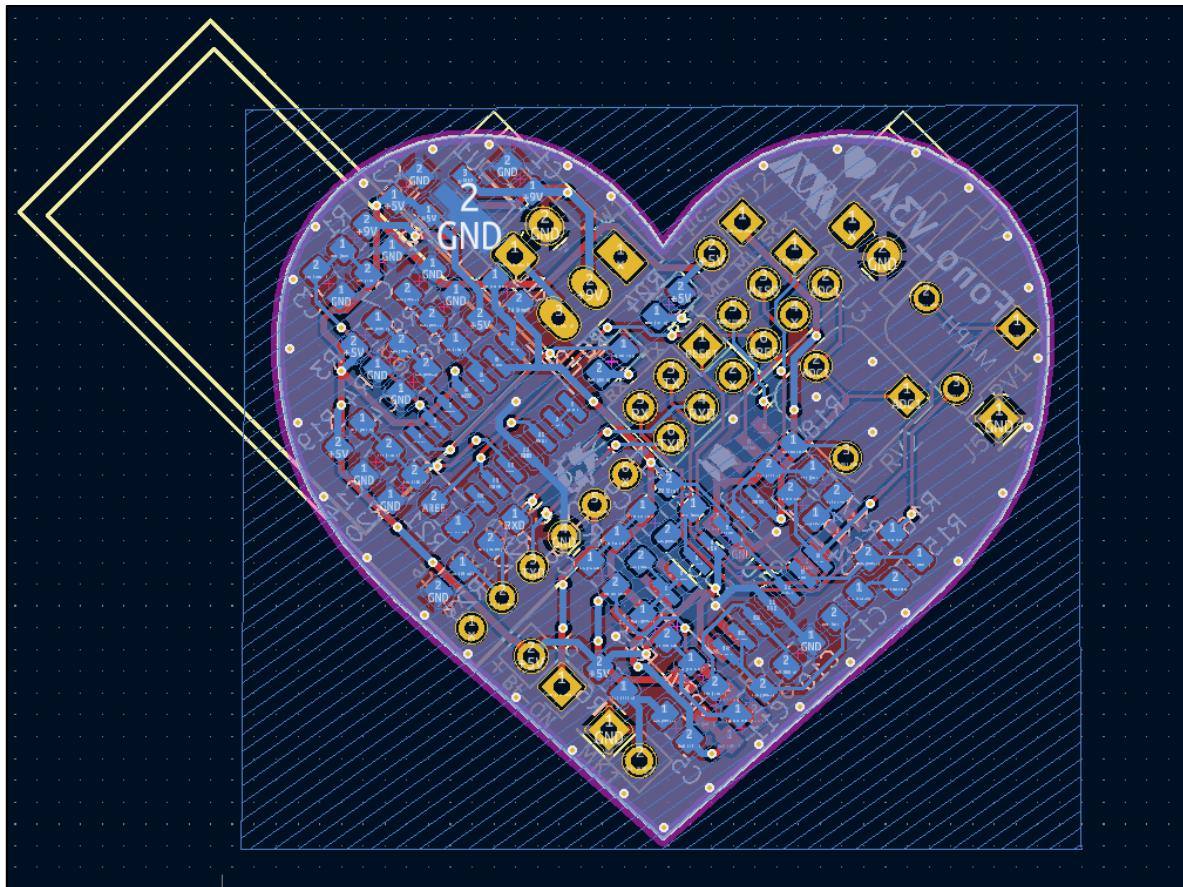


Figura 28.- Diagrama del PCB, (módulo A).

10.1 Fuente de alimentación

10.1.1 Regulador de voltaje L78L05

Tanto el circuito de acondicionamiento como el microcontrolador y el módulo de Bluetooth HC-05 se encuentran diseñados para trabajar con una alimentación unipolar de 5V, por lo cual se hace uso de un L78L05 y una batería de 9V para alimentar el circuito. ([Figura 29](#)).

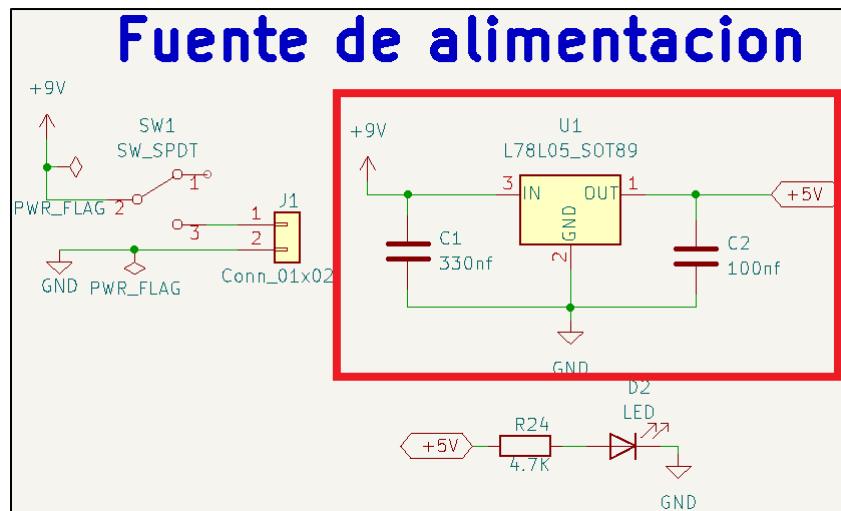


Figura 29.- Diagrama de conexiones, L78L05 (módulo A).

Se eligió un L78L05 ya que el consumo máximo visto en las pruebas de desempeño es menor de 100mA, siendo ésta la corriente máxima entregada por el regulador ([8.1 Regulador de Voltaje positivo L78L05](#)).

El circuito utilizado para el regulador consta de un L78L05 y dos capacitores, este circuito fue obtenido de su hoja de datos ([Figura 30](#)). Estos capacitores se encargan de regular cualquier pico de voltaje que pueda presentarse tanto la batería de 9V como en la salida del regulador, manteniendo una regulación estable.

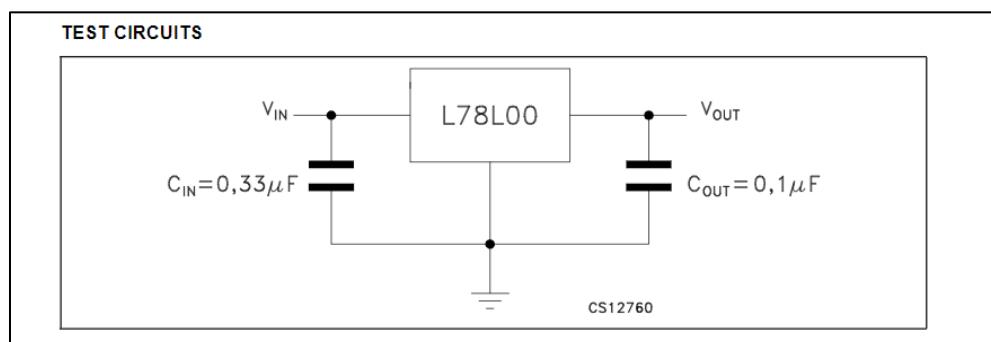


Figura 30.- Circuito de prueba, L78L05 [10], (módulo A).

10.1.2 Interruptor e indicador led de encendido

Se agrega un interruptor el cual permita encender o apagar el circuito sin necesidad de desconectar la batería de 9V. De igual forma, se agrega un led a la salida del regulador con el fin de tener un indicador cuando el circuito se encuentre encendido ([Figura 31](#)).

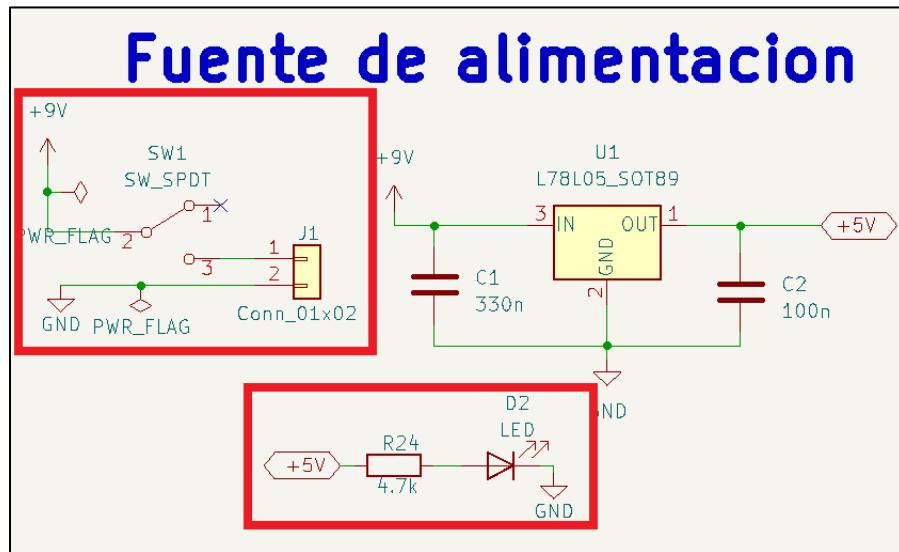


Figura 31.- Interruptor e indicador luminoso de encendido, (módulo A).

10.2 Indicador de batería baja

Para evitar un comportamiento errático debido a la caída de voltaje del desgaste de la batería, es necesario contar con un indicador de batería baja.

El indicador consta de un LM324 utilizado como comparador, dos divisores de voltaje y un led conectado en la salida del comparador ([Figura 32](#)).

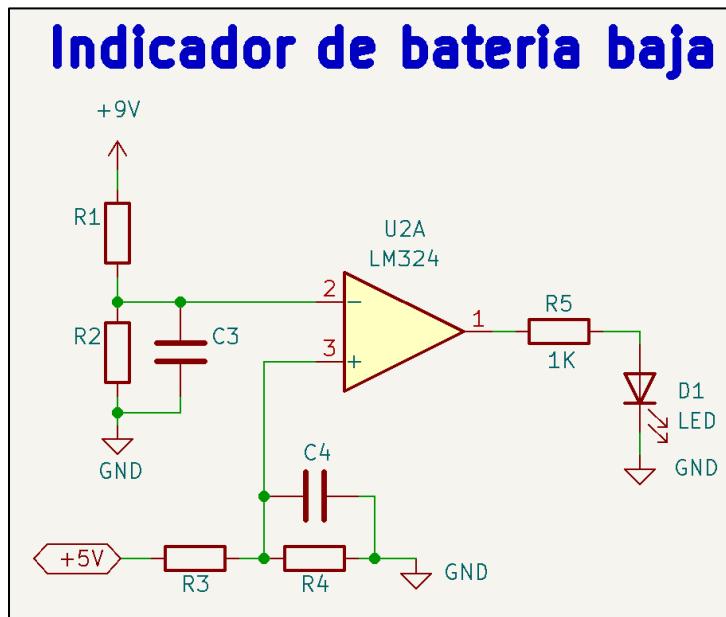


Figura 32.- Circuito indicador de batería baja, (módulo A).

El comparador es alimentado por el voltaje regulado del L78L05, es decir, se tiene una alimentación unipolar de 5V. Según lo propuesto en la sección [8.7 Amplificador operacional LM324](#), los voltajes de entrada en el comparador se encuentran en el rango [1V,3V].

Según lo propuesto en la sección [8.1 Regulador de Voltaje positivo L78L05](#), el voltaje mínimo para la operación del circuito es de 7.5V.

Dadas las especificaciones anteriores, mientras el voltaje de la batería se encuentre en el rango de [7.5 V, 9V], la salida del comparador estará referenciada al voltaje de alimentación negativo, cuando el voltaje de la batería decae por debajo de 7.5V, la salida del comparador cambiara hacia la alimentación positiva, teniendo un voltaje de 3.5V en su salida encendiendo el indicador led.

10.2.1 Voltaje en la terminal negativa (Divisor de voltaje 9V)

El primer divisor de voltaje se encuentra referenciado a la batería de 9V, este divisor debe reducir el voltaje de tal forma que se tenga un valor en la entrada del comparador en el rango de [1V,3V]. De igual forma, cuando el voltaje de la batería se encuentre alrededor de 7.5V, el voltaje en el divisor debe estar contenido en este rango de valores ([Figura 33](#)).

Indicador de batería baja, terminal negativa

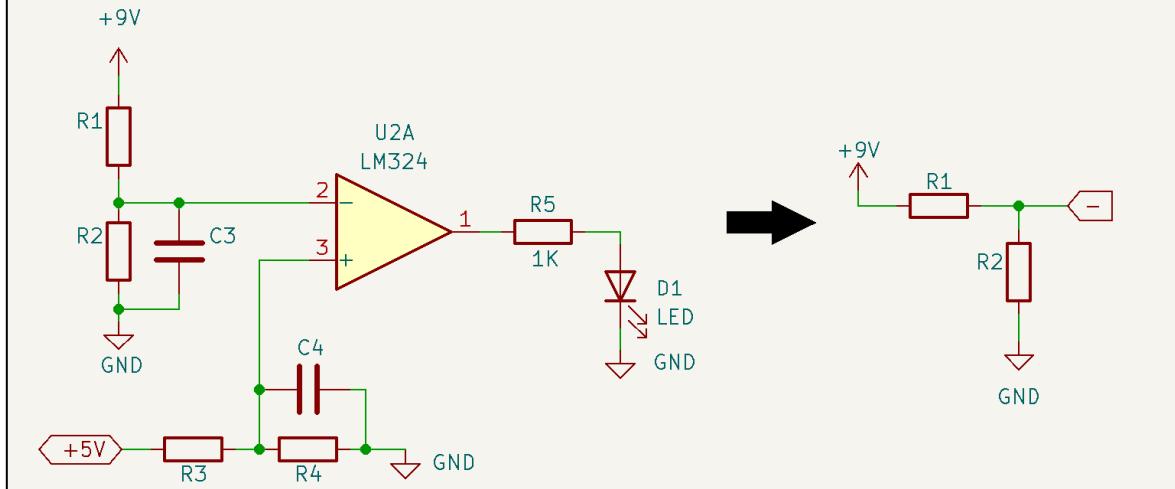


Figura 33.- Divisor de voltaje de 9V, terminal (-) del comparador, (módulo A).

Se propone tener un voltaje de 3V en el divisor cuando la batería de 9V se encuentra en su carga máxima, teniendo que:

$$V_{out} = \frac{R_2}{R_1 + R_2} V_{in} \rightarrow 3V = \frac{R_2}{R_1 + R_2} 9V \rightarrow 3R_1 + 3R_2 = 9R_2 \rightarrow R_1 = \frac{6}{3} R_2 \rightarrow R_1 = 2R_2$$

El valor de R1 debe ser 2 veces el valor de R2. Se propone usar un valor de R1=100KΩ y R2=51KΩ

Teniendo estos valores, se hace el cálculo cuando el voltaje en la batería es de 7.5V, este dato será usado para calcular el segundo divisor de voltaje.

$$V_{out} = \frac{R_2}{R_1 + R_2} V_{in} \rightarrow V_{out} = \frac{R_2}{R_1 + R_2} 7.5V \rightarrow V_{out} = \frac{51K\Omega}{100K\Omega + 51K\Omega} 7.5V = \frac{51K\Omega}{151K\Omega} 7.5V$$

$$V_{out} = 2.53V$$

Con los voltajes calculados se sabe que:

- Voltaje en la batería = 9V -> Voltaje en la terminal negativa = 3V
- Voltaje en la batería = 7.5V -> Voltaje en la terminal negativa = 2.5V

Por último, se agrega el capacitor C3, el cual filtra el ruido térmico y cualquier tipo de perturbación que pueda tener la fuente de alimentación.

El capacitor C3 y las resistencias R1 y R2 crean un filtro pasa bajas. La frecuencia de corte de este filtro deberá ser inferior a 20Hz, esto con el fin de no afectar el desempeño del circuito de audio. Proponiendo un capacitor de 470nF, se tiene:

$$f_c = \frac{1}{2\pi(100K\Omega||51K\Omega)470nf} = 10.02Hz$$

10.2.2 Voltaje en la terminal positiva (Divisor de voltaje 5V)

Para obtener un cambio en la polaridad de la salida del comparador es necesario introducir un voltaje de referencia similar al calculado en la sección [10.2.1 Voltaje en la terminal negativa \(Divisor de voltaje 9V\)](#) cuando el voltaje en la batería de 9V decae a su valor mínimo permitido de 7.5V. Siendo el voltaje calculado en la sección anterior de 2.5V, este será el valor de voltaje a introducir en la terminal positiva del comparador, de tal forma que, cuando el voltaje en la batería disminuya, el voltaje en la terminal negativa del comparador también disminuye, pasando por debajo del voltaje de referencia de 2.5V, al tener un voltaje mayor en la terminal positiva que en la terminal negativa, el voltaje en la salida del comparador cambia a ser referenciado hacia el voltaje de alimentación, encendiendo el led ([Figura 34](#)).

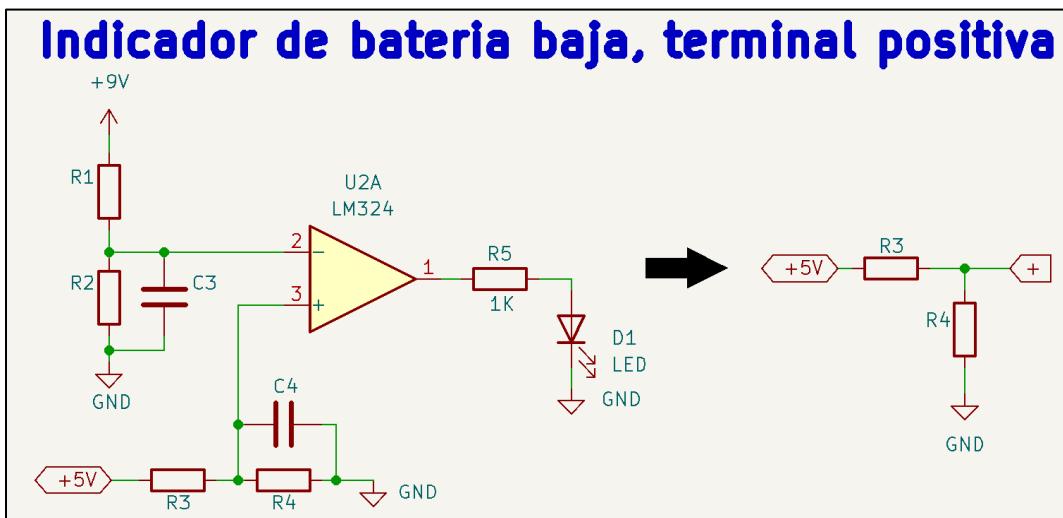


Figura 34.- Divisor de voltaje de 5V, terminal (+) del comparador, (módulo A).

Este segundo divisor necesita un voltaje fijo de operación, el cual será tomado de la salida del regulador L78L05:

$$V_{out} = \frac{R_4}{R_3 + R_4} V_{in} \rightarrow 2.5V = \frac{R_4}{R_3 + R_4} 5V \rightarrow 2.5R_3 + 2.5R_4 = 5R_4 \rightarrow R_3 = \frac{2.5}{2.5} R_4 \rightarrow R_3 = R_4$$

El valor de las resistencias R3 y R4 debe ser el mismo, se propone R3=R4=100 KΩ

De igual forma que en el divisor de 9V, se agrega un capacitor C4 para filtrar el ruido térmico y cualquier tipo de perturbación que pueda tener la fuente de alimentación.

El capacitor C4 y las resistencias R3 y R4 crean un filtro pasa bajas. La frecuencia de corte de este filtro deberá ser inferior a 20Hz, esto con el fin de no afectar el desempeño del circuito de audio. Proponiendo un capacitor de 470nF, se tiene:

$$f_c = \frac{1}{2\pi(100K\Omega||100K\Omega)470nf} = 6.77Hz$$

Finalmente, el circuito queda de la siguiente forma:

Indicador de batería baja

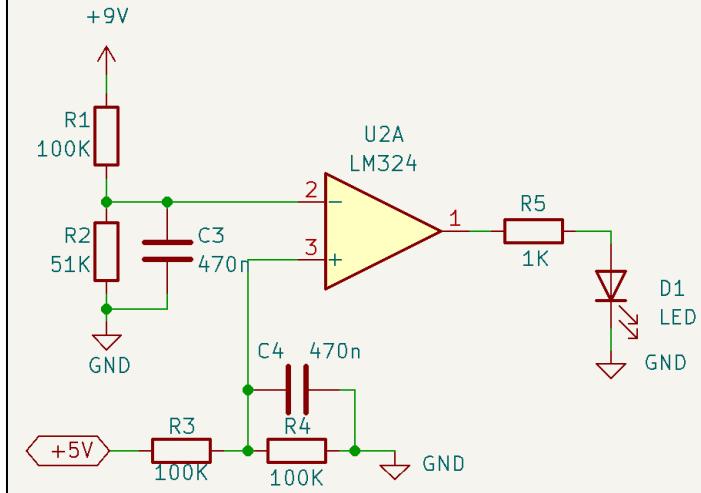


Figura 35.- Circuito indicador de batería baja con valores, (módulo A).

10.3 Pre-amplificador del micrófono AD623

El pre-amplificador consta de una resistencia de polarización conectada entre el micrófono y la fuente de alimentación, capacitores de acople y un amplificador de instrumentación AD623, el cual se encarga de recibir la señal del micrófono y llevarla a etapas posteriores ([Figura 36](#)).

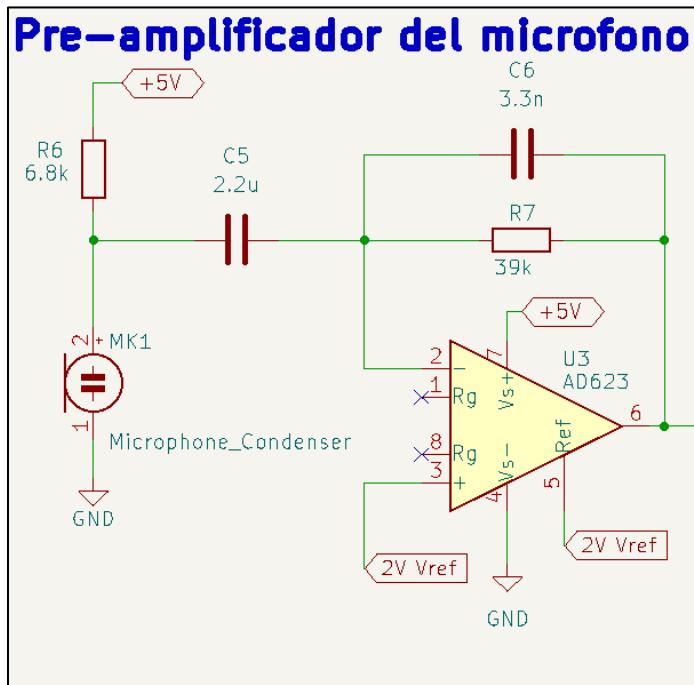


Figura 36.- Circuito Pre-amplificador, (módulo A).

Según la sección [8.7 Amplificador operacional LM324](#), los límites de voltaje del LM324 son:

- Límite inferior = $V_- + 1V = 1V$
- Límite superior = $+V - 2V = 3V$

Según la sección [8.3 Amplificador de instrumentación AD623](#), límites de voltaje del AD623 son:

- Límite inferior = $V_- + 0.15V = 0.15V$
- Límite superior = $+V - 1.5V = 3.5V$

Dado esto, se acotará los voltajes de trabajo del AD623 para entrar en el rango del LM324

El pre-amplificador puede ser dividido en dos secciones:

- Resistencia de polarización y capacitor de acople del micrófono.
- Etapa de ganancia (Amplificador de Transimpedancia).

Nota: Parte del diseño del pre-amplificador fue tomado de las notas del fabricante Texas Instruments "Single-Supply, Electret Microphone Pre-Amplifier Reference Design" [20].

10.3.1 Resistencia de polarización y capacitor de acople del micrófono

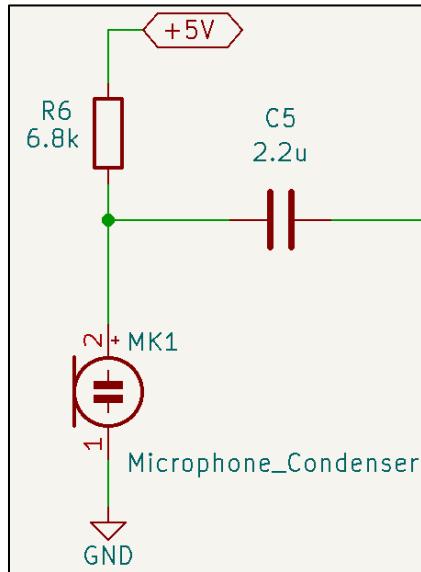


Figura 37.- Resistencia de polarización y capacitor de acople, (módulo A).

El JFET interno del micrófono Electret es polarizado por la resistencia R6 ([Figura 37](#)), el valor de esta resistencia puede ser calculado conociendo el voltaje de alimentación Vcc, así como el voltaje y corriente típica de operación del micrófono dados en sección [8.4 Micrófono Electret CMC-60158-42P](#) en la [Tabla 1](#):

$$R6 = \frac{V_{cc} - V_{mic}}{I_{mic}} = \frac{5V - 2V}{0.4mA} = 7.5K\Omega \rightarrow 6.8K\Omega$$

Se toma el valor inmediato inferior de resistencia siendo de $6.8K\Omega$ para compensar las variaciones de voltaje que pueda tener la fuente de alimentación y asegurar el correcto funcionamiento del micrófono.

Para extraer las señales eléctricas producidas por el micrófono (señales en AC), se utiliza un capacitor de acople C5. Este capacitor junto con la resistencia R6 forman un filtro pasa altas.

La señal de interés del corazón se encuentra en un rango de frecuencias de 20Hz a 1KHz ([Figura 4](#)), por lo cual, se propone una frecuencia de corte de 10Hz, teniendo la siguiente ecuación:

$$C5 = \frac{1}{2\pi R6 f_c} = \frac{1}{2\pi(6.8K\Omega)(10Hz)} = 2.3\mu F \rightarrow 2.2\mu F$$

La capacitancia de $2.2\mu F$ fue propuesta de tal forma que la impedancia vista por el capacitor en el rango de frecuencias deseado sea menor que el valor de resistencia R6.

Para 20Hz, la impedancia del capacitor C5 sería de:

$$Z_{c1} = \frac{1}{j\omega C} = \frac{1}{j2\pi(20Hz)2.2\mu F} = 3.6K\Omega$$

Para 1KHz, la impedancia del capacitor C5 viene siendo:

$$Z_{c1} = \frac{1}{j\omega C} = \frac{1}{j2\pi(1\text{KHz})2.2\mu\text{F}} = 72\Omega$$

10.3.2 Etapa de ganancia (Amplificador de Transimpedancia)

Esta etapa se encarga de amplificar la corriente de AC que circula por el capacitor C5, a través de la resistencia R7. Un capacitor C6 se conecta en paralelo con la resistencia R7 para compensar las capacitancias parásitas de la terminal inversora del opam. Por último, dado que el amplificador de instrumentación se encuentra alimentado con una fuente unipolar, se conecta un voltaje de referencia en la terminal VRef y en su terminal positiva, obteniendo una componente de DC en el voltaje de operación del amplificador. Gracias a este voltaje, es posible procesar la señal de AC proveniente de la etapa anterior del circuito y entregarla a una etapa posterior ([Figura 38](#)).

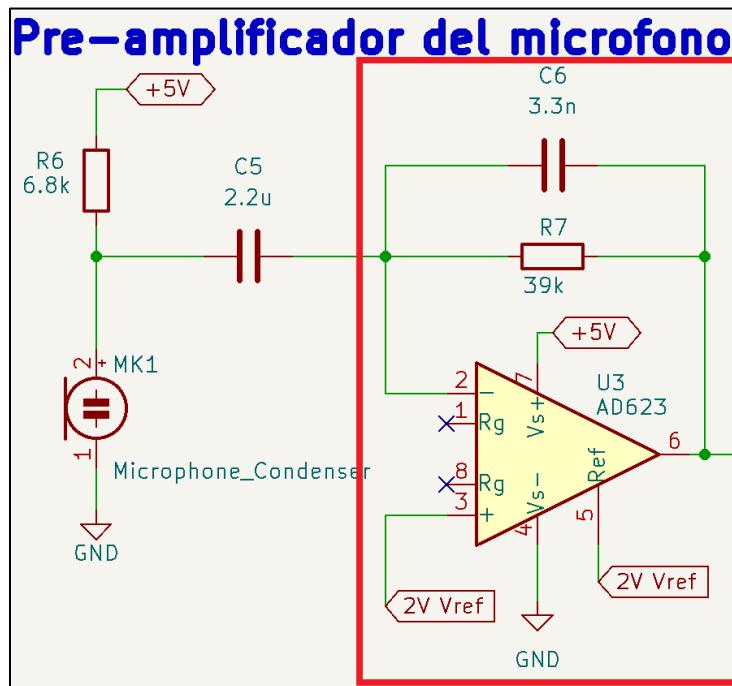


Figura 38.- Amplificador de Transimpedancia, (módulo A).

Para calcular la ganancia es necesario convertir la sensibilidad del micrófono de dB a un valor lineal, tomando la sensibilidad de la [Tabla 1](#), se tiene:

$$10^{\frac{-42\text{dB}}{20}} = 7.94\text{mV/Pa}$$

Debido a que el pre-amplificador es de tipo Transimpedancia, es necesario convertir este valor de sensibilidad a uno de corriente por Pascal.

Según la hoja de datos del micrófono CMC-6015-42P [13], la sensibilidad fue medida con una impedancia de 2.2KΩ, por lo cual la corriente por Pascal sería de:

$$\frac{\frac{7.94\text{mV}}{\text{Pa}}}{2.2\text{K}0\text{mhs}} = \frac{3.61\text{uA}}{\text{Pa}}$$

El cálculo de ganancia depende del valor máximo de sonido esperado en la entrada del micrófono. Para este diseño se tomará un valor de 100dB SPL el valor de sonido máximo a 0.3V, esto con el fin de amplificar la señal sin llegarla a saturar y reducir la amplificación de ruido en el circuito. Conociendo que 100dB SPL corresponden a 2 Pascales, se tiene:

$$\frac{3.61\text{uA}}{\text{Pa}} 2\text{Pa} = 7.22\text{uA}$$

La ganancia del amplificador de Transimpedancia quedaría como:

$$V_{\text{OUT}} = I_{\text{In}} * R_7 \rightarrow R_7 = \frac{V_{\text{out}}}{I_{\text{In}}} = \frac{0.3\text{V}}{7.22\text{uA}} = 41.55\text{K}\Omega \rightarrow 39\text{K}\Omega$$

Para este diseño, la ganancia de ruido del amplificador viene dado por la fórmula:

$$A_N = 1 + \frac{R_7}{R_6} = 1 + \frac{39\text{K}\Omega}{6.8\text{K}\Omega} = 6.7$$

Entre menor sea la resistencia R7 menor será la presencia de ruido en el circuito, dado esto, se incorpora una etapa de ganancia al final del circuito de acondicionamiento para compensar la amplitud de la señal en caso de requerirlo.

El capacitor C6 forma un filtro pasa bajas junto con la resistencia R7, la frecuencia de corte de este filtro debe ser lo suficientemente alta para no afectar las frecuencias de interés (fc= 1KHz). Dado esto, el cálculo del capacitor C6 queda como:

$$C_6 = \frac{1}{2\pi f_c R_7} = \frac{1}{2\pi(1\text{KHz})(39\text{K}\Omega)} = 4.04\text{nf} \rightarrow 3.3\text{nf}$$

10.4 Voltaje de referencia offset

Para introducir las variaciones de voltaje AC provenientes del micrófono a las etapas posteriores del circuito, es necesario agregar una componente de offset ya que el circuito está diseñado para trabajar con voltajes positivos. Esta componente puede ser agregada en la terminal no inversora de los opamps usando un divisor de tensión y un amplificador operacional como buffer (Figura 39).

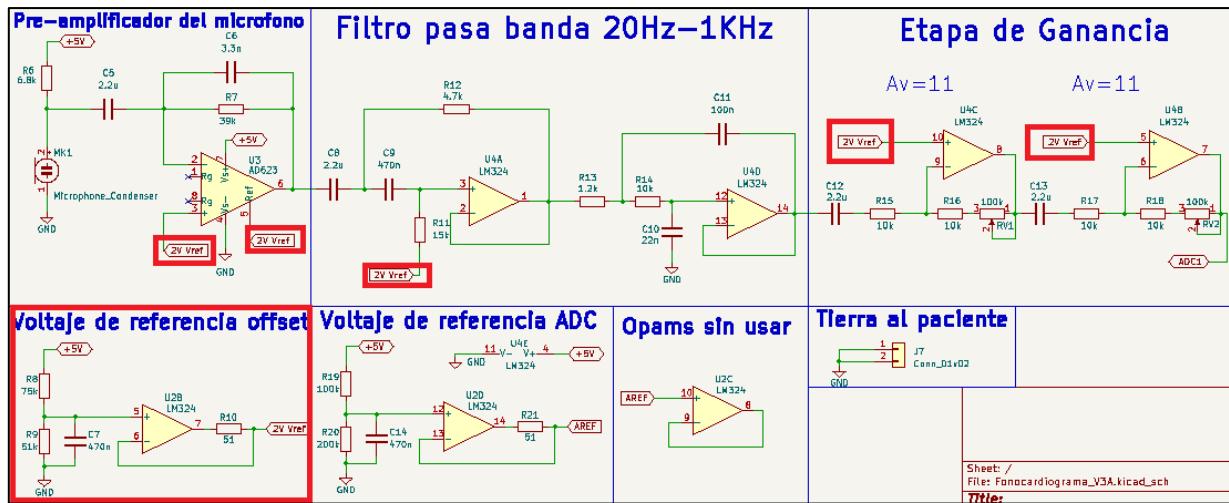


Figura 39.- Red de polarización de opamps, (módulo A).

Conociendo que los voltajes de operación discutidos en la sección [8.7 Amplificador operacional LM324](#) son de límite inferior 1V y límite superior 3V, la componente de offset deberá ubicarse entre estos valores, de tal forma que se tenga el mismo rango de voltaje hacia los límites. Dicho esto, la componente será de 2V, teniendo $\pm 1V$ de variación.

Para calcular las resistencias se simplificará el circuito, teniendo un divisor de tensión simple (Figura 40)

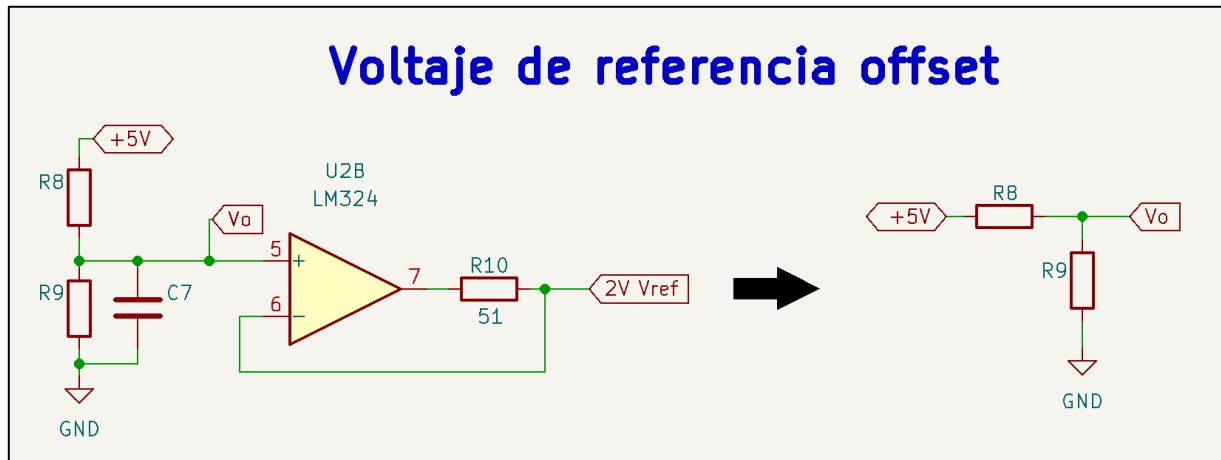


Figura 40.- Divisor de tensión, voltaje de referencia offset, (módulo A).

Conociendo que:

$$V_o = \frac{R_9}{R_8 + R_9} 5V \rightarrow 2V = \frac{R_9}{R_8 + R_9} 5V \rightarrow 2R_8 + 2R_9 = 5R_9 \rightarrow R_8 = \frac{3}{2} R_9$$

El valor de R8 debe ser 3/2 veces el valor de R9. Se propone usar un valor de R8=75K y R9=51K

Por último, se agrega el capacitor C7, el cual filtra el ruido térmico y cualquier tipo de perturbación que pueda tener la fuente de alimentación.

El capacitor C7 y las resistencias R8 y R9 crean un filtro pasa bajas. La frecuencia de corte de este filtro deberá ser inferior a 20Hz, esto con el fin de no afectar el desempeño del circuito de audio. Proponiendo un capacitor de 470nF, se tiene:

$$f_c = \frac{1}{2\pi(75k||51k)470nf} = 11.15\text{Hz}$$

Teniendo los valores calculados, el circuito de voltaje de referencia offset se muestra a continuación ([Figura 41](#)).

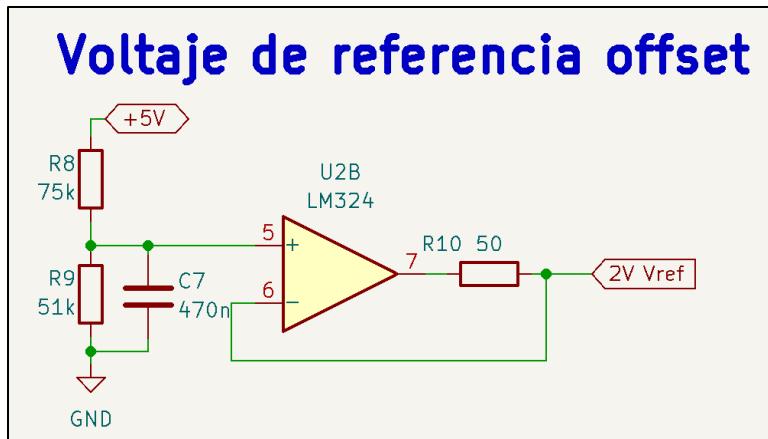


Figura 41.- Voltaje de referencia offset con valores , (módulo A).

10.5 Filtro pasa banda 20Hz-1KHz

El filtro pasa bandas del fonocardiograma se dividió se dos secciones:

- Filtro pasa altas 20Hz
- Filtro pasa bajas 1KHz

En ambos filtros se usan dos polos con una configuración Sallen-Key de tipo Butterworth. Los valores de los componentes fueron obtenidos gracias al software de “Filter Pro”. Los filtros obtenidos por este medio tienen como base una alimentación simétrica. Para pasar de una fuente dual a unipolar se hace uso de las notas de aplicación de Texas Instruments *“Single-supply, 2nd-order, Sallen-Key band-pass filter circuit”* [21].

Según las notas de aplicación, es necesario agregar una componente de offset en la entrada del filtro pasa altas, el circuito con esta modificación puede ser visto a continuación ([Figura 42](#)).

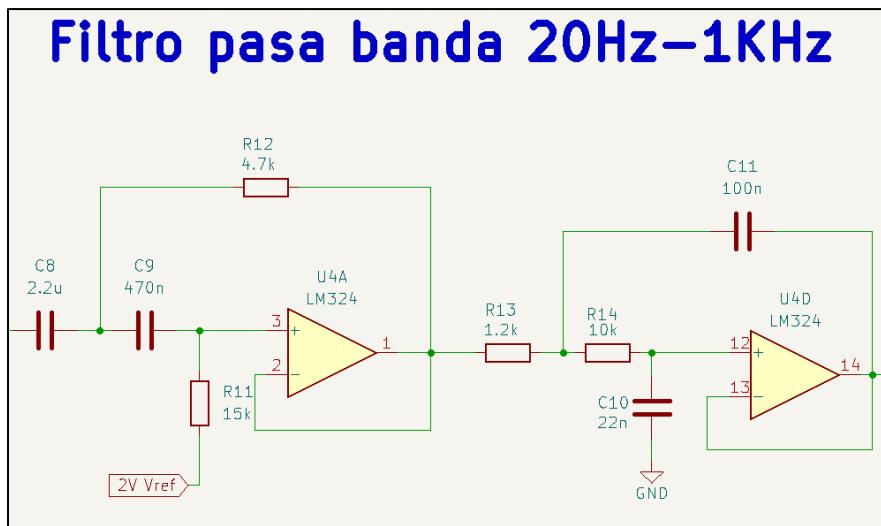


Figura 42.- Filtro pasa bandas 20Hz-1KHz, (módulo A).

De igual forma, se agrega un capacitor de acople C8 para introducir las variaciones de AC de partes anteriores del circuito en el filtro pasa bandas.

10.5.1 Filtro pasa altas 20Hz

Para el filtro pasa altas, se eligió una frecuencia de corte de 20Hz. Según los datos proporcionados por el software, se necesita un amplificador con una ganancia ancho de banda mínima de 2KHz ([Figura 43](#)). Se propone usar un LM324 ya que posee un GBP de 1.2MHz.

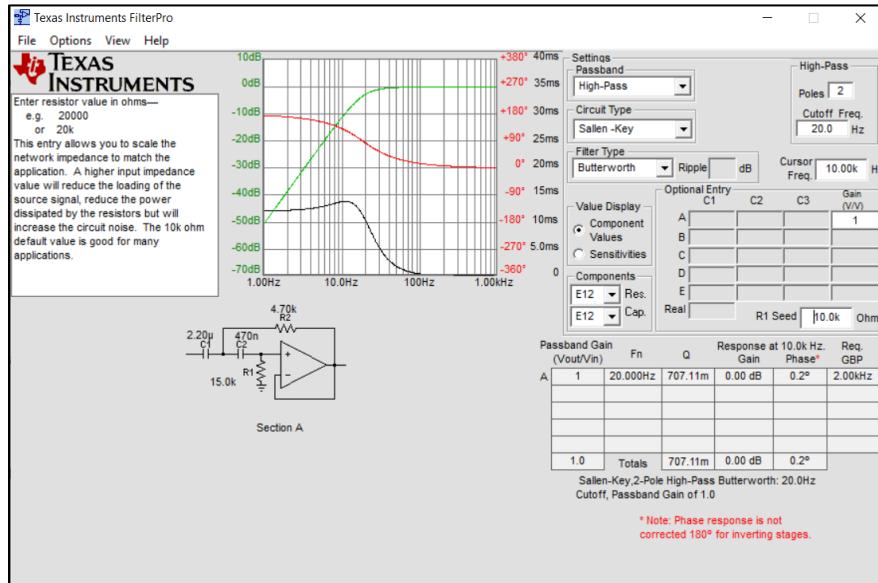


Figura 43.- Filtro pasa altas fc=20Hz, (módulo A).

10.5.2 Filtro pasa bajas 1KHz

Para el filtro pasa bajas, la frecuencia de corte escogida es de 1Khz. Para realizar este filtro se necesita una ganancia-ancho de banda de 100Khz ([Figura 44](#)). Se propone usar el LM324 ya que su GBP es de 1.2MHz.

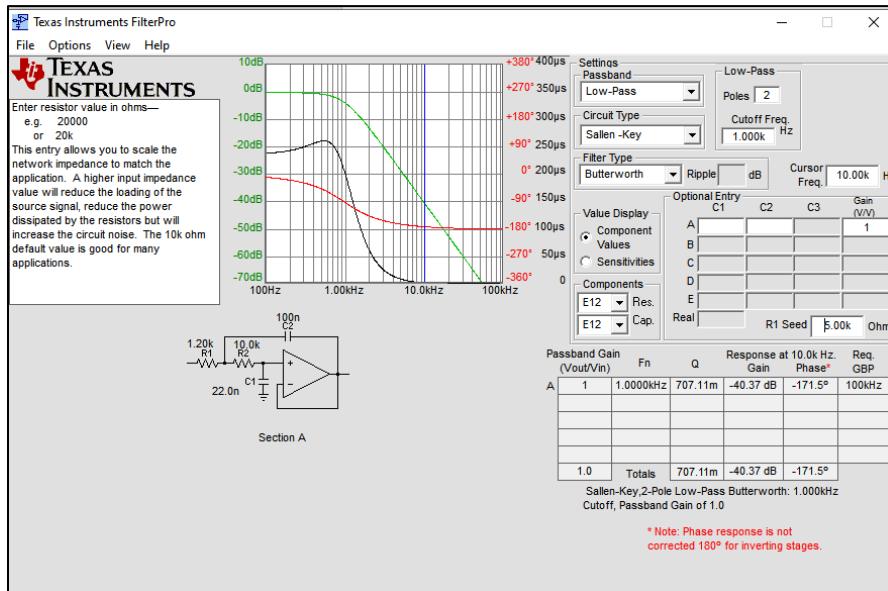


Figura 44.- Filtro pasa bajas fc=1Khz, (módulo A).

10.6 Etapa de ganancia

Una vez filtrada la señal, ésta debe ser amplificada antes de ser entregada al microcontrolador. Se propone tener una amplificación máxima $A_v=1$ hasta $A_v=121$, esto con el fin de ajustar manualmente la intensidad de la señal sin llegar a saturarla.

Según la hoja de datos del LM324 [16] la ganancia en lazo abierto en 1KHz es de 40dB o $A_v=100$ (*Figura 45*), debido a que el valor de ganancia se encuentra en el límite a la frecuencia requerida, se implementarán dos etapas de ganancia.

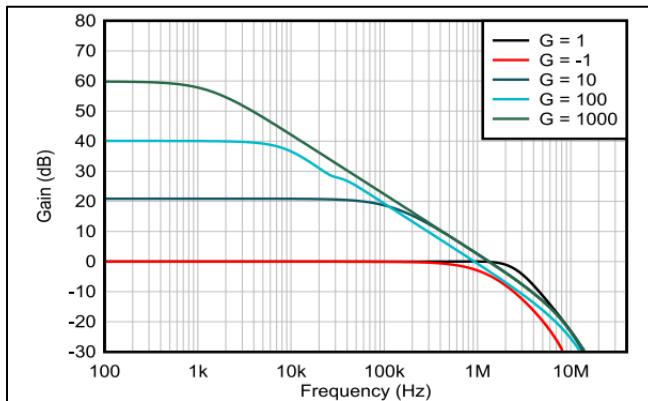


Figura 45.- Relación ganancia-ancho de banda en lazo abierto LM324, (módulo A).

El circuito de ganancia puede ser visto en la *Figura 46*, éste se encuentra dividido en 2 etapas, cada una con una ganancia de $A_v=1$ hasta $A_v=11$.

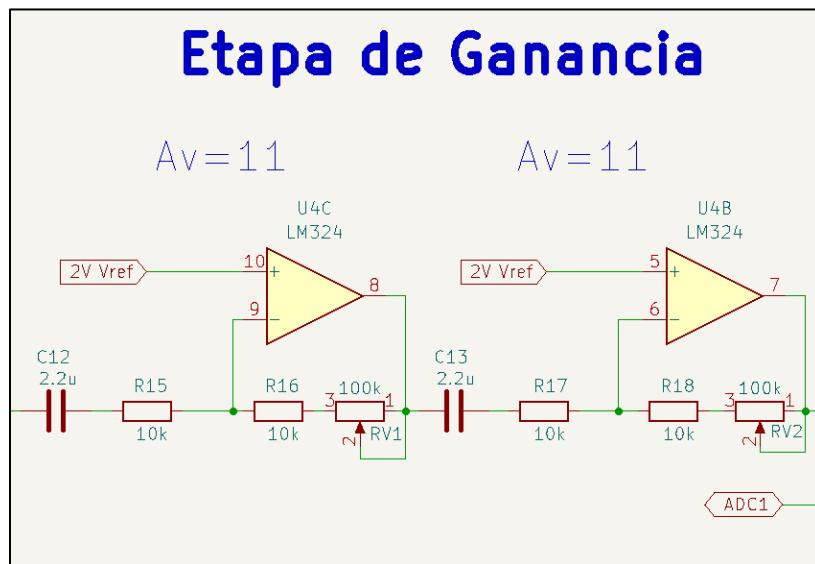


Figura 46.- Etapa de ganancia, (módulo A).

Para el análisis matemático de la etapa de ganancia, se trabajará con los valores del amplificador de la izquierda (*Figura 46*). El análisis y los resultados obtenidos pueden ser extrapolados para el amplificador de la derecha (*Figura 46*).

Para obtener una ganancia unitaria, al ajustar el potenciómetro a su valor mínimo (corto circuito), es necesario que el valor de la resistencia en el lazo de retroalimentación sea igual al de la resistencia de entrada de la señal, por lo cual se hace uso de la resistencia R16, la cual se mantiene fija en todo momento, teniendo que:

$$Av = -\frac{R_{16} + R_{V1}}{R_{15}} = -\frac{R_{16} + 0}{R_{15}} = -\frac{R_{16}}{R_{15}} = -1 \rightarrow R_{16} = R_{15}$$

Para calcular la ganancia variable de 11, se ajusta el potenciómetro al máximo valor, teniendo:

$$Av = -\frac{R_{16} + R_{V1}}{R_{15}} \rightarrow R_{16} + R_{V1} = (Av)R_{15} \rightarrow R_{V1} = (Av)R_{15} - R_{16}$$

Sabiendo que $R_{15}=R_{16}$ y que $Av=11$

$$R_{V1} = (11)R_{15} - R_{15} \rightarrow R_{V1} = 10R_{15}$$

La resistencia RV1 debe ser 10 veces más grande que R15, se propone una resistencia $R_{15}=R_{16}$ de $10K\Omega$ y $RV1$ de $100K\Omega$.

Por último, los capacitores C12 y C13 se encargan de acoplar la señal de etapas anteriores con la etapa de ganancia.

10.7 Microcontrolador attiny 85

El microcontrolador utilizado en este proyecto es un attiny85 de la marca Atmel, se eligió este microcontrolador ya que cuenta con 6 puertos de entrada/salida y convertidores ADC, además de permitir el ser programado bajo el IDLE de Arduino.

El trabajo del microcontrolador es el que transformar la señal analógica del circuito de acondicionamiento en un valor digital y mandarlo a través del puerto serial al módulo Bluetooth.

Para programar el attiny85 es necesario contar con una placa Arduino externa ya que ésta realizará el trabajo de ISP para programar el microcontrolador.

10.7.1 Voltaje de referencia ADC

Para no perder resolución en la conversión analógico-digital de la señal del corazón se propone modificar el voltaje de referencia del ADC a través del pin AREF del microcontrolador, (para más información ver la sección [8.9 Microcontrolador Attiny 85](#)).

Según los límites establecidos en la sección [8.7 Amplificador operacional LM324](#), el voltaje máximo de trabajo será de 3V, por lo cual se propone dar un margen de error referenciando el ADC a 3.3V.

Para tener un voltaje estable, se utiliza un seguidor de voltaje referenciado a un divisor de voltaje. El cálculo de este divisor se muestra en el siguiente circuito ([Figura 47](#)):

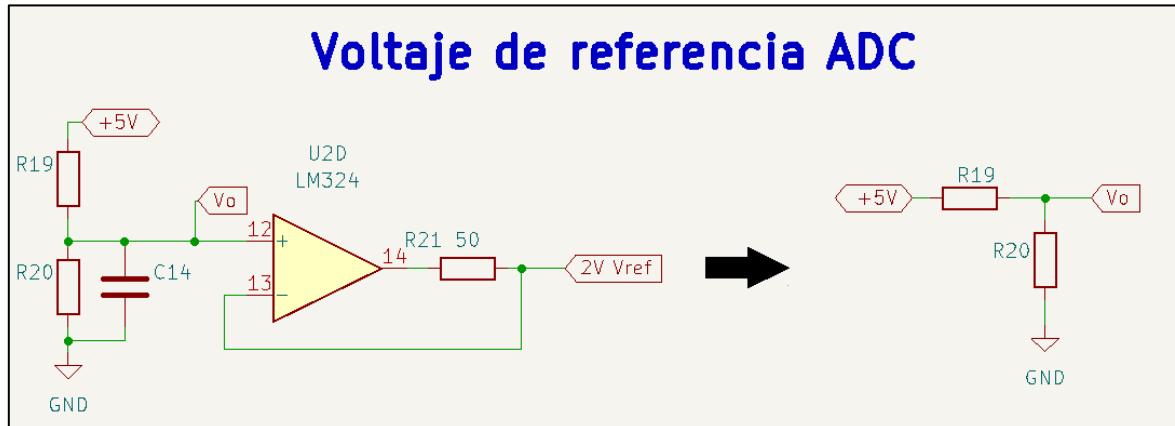


Figura 47.-Divisor de tensión, voltaje de referencia ADC, (módulo A).

Conociendo que:

$$V_o = \frac{R_{20}}{R_{19} + R_{20}} 5V \rightarrow 3.3V = \frac{R_{20}}{R_{19} + R_{20}} 5V \rightarrow 3.3R_{19} + 3.3R_{20} = 5R_{20}$$

$$R_{19} = \frac{1.7}{3.3} R_{20} \rightarrow R_{19} \approx \frac{1}{2} R_{20}$$

El valor de R19 debe ser ≈0.5 veces el valor de R20. Se propone usar un valor de R19=100KΩ y R20=200KΩ.

Se agrega el capacitor C14, el cual filtra el ruido térmico y cualquier tipo de perturbación que pueda tener la fuente de alimentación.

El capacitor C14 y las resistencias R19 y R20 crean un filtro pasa bajas. La frecuencia de corte de este filtro deberá ser inferior a 20Hz para no afectar el desempeño del circuito de audio. Proponiendo un capacitor de 470nF, se tiene:

$$f_c = \frac{1}{2\pi(200K\Omega||100K\Omega)470nf} = 5.07Hz$$

Finalmente, el circuito de voltaje de referencia ADC puede ser visto a continuación ([Figura 48](#)).

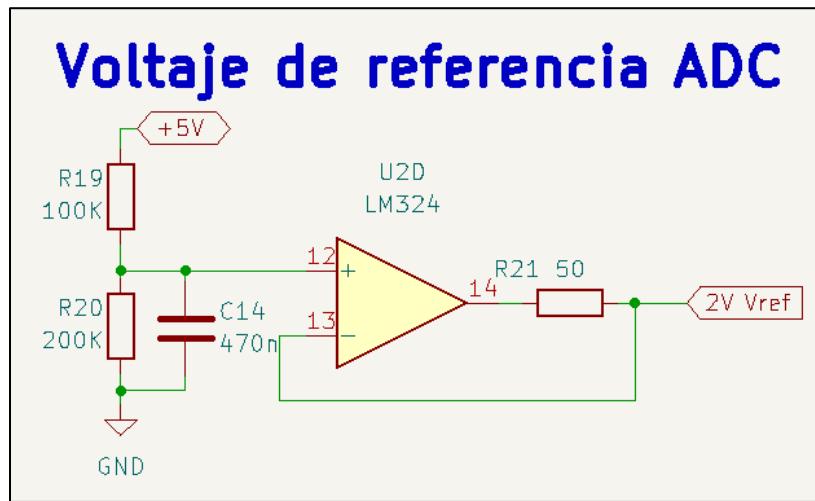


Figura 48.- Voltaje de referencia ADC con valores, (módulo A).

10.8 Conexiones del attiny 85

Debido a que el attiny85 cuenta únicamente con 6 pines de entrada/salida, se tienen dos modos de configuración, el modo programador y el modo de funcionamiento normal. Para elegir entre estos modos, se conectan terminales dobles en cada uno de sus pines, esto con el fin de conectar o desconectar el microcontrolador del circuito a través de jumpers conectados en sus terminales, en la [Figura 49](#) se muestran de color azul las terminales a puentear.

De esta forma, es posible conectar el microcontrolador al circuito de acondicionamiento puenteando sus terminales con jumpers (modo de funcionamiento normal), o dejar las terminales libres para ser programado de manera externa (modo programador).

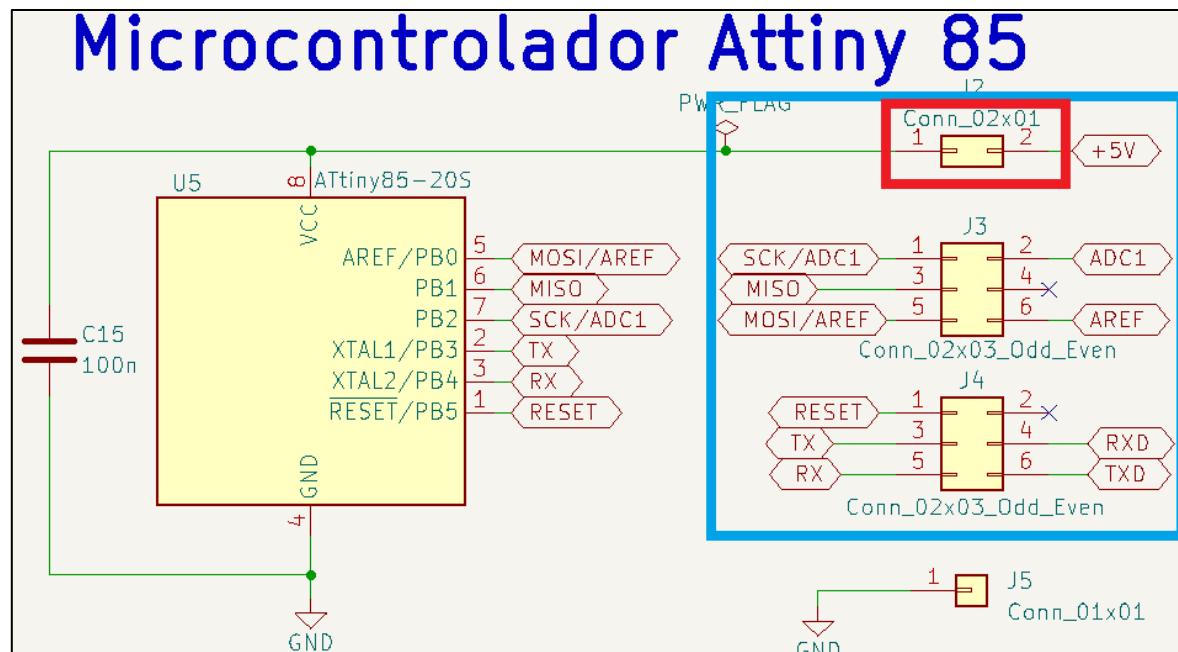


Figura 49.- Terminales de conexión del microcontrolador attiny85, (módulo A).

En la [Figura 49](#), de color rojo se muestra una terminal doble entre la conexión Vcc a +5V del attiny85, esta conexión sirve para encender o apagar el microcontrolador independientemente del circuito de acondicionamiento, esto con fin de probar por separado diferentes partes del circuito al momento de ensamblarlo.

10.8.1 Conexión al programador

Para conectar el attiny85 al programador, es necesario quitar los jumpers del microcontrolador, dejando las terminales al aire. Se conectan de manera externa cables hacia las terminales SCK/AD1, MISO, MOSI/AREF y RESET ([Figura 50](#)). El diagrama de conexiones será discutido en la sección [14 Programación del attiny85](#).

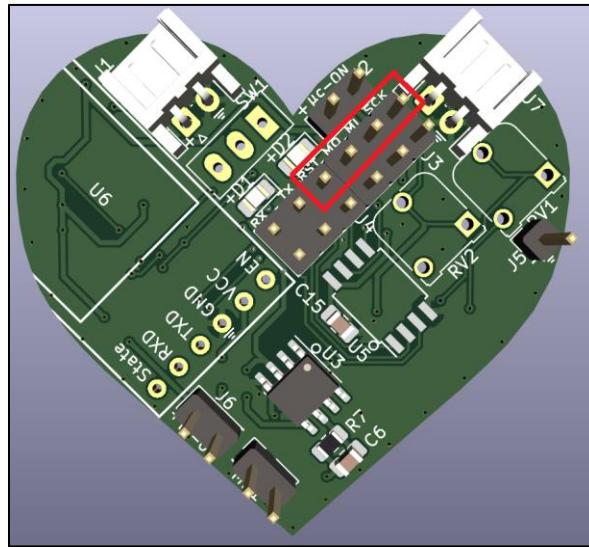


Figura 50.- Terminales de programación, (módulo A).

10.8.2 Conexión al circuito de acondicionamiento

La terminal MOSI/AREF se conecta a la salida del voltaje de referencia ADC visto en la sección [10.7.1 Voltaje de referencia ADC](#) siendo esta terminal en donde se suministra el voltaje de referencia para el convertidor analógico-digital. Por otra parte, la terminal SCK/AD1 se conecta a la salida del circuito de acondicionamiento, siendo el pin entrada de la señal analógica al microcontrolador ([Figura 51](#)).

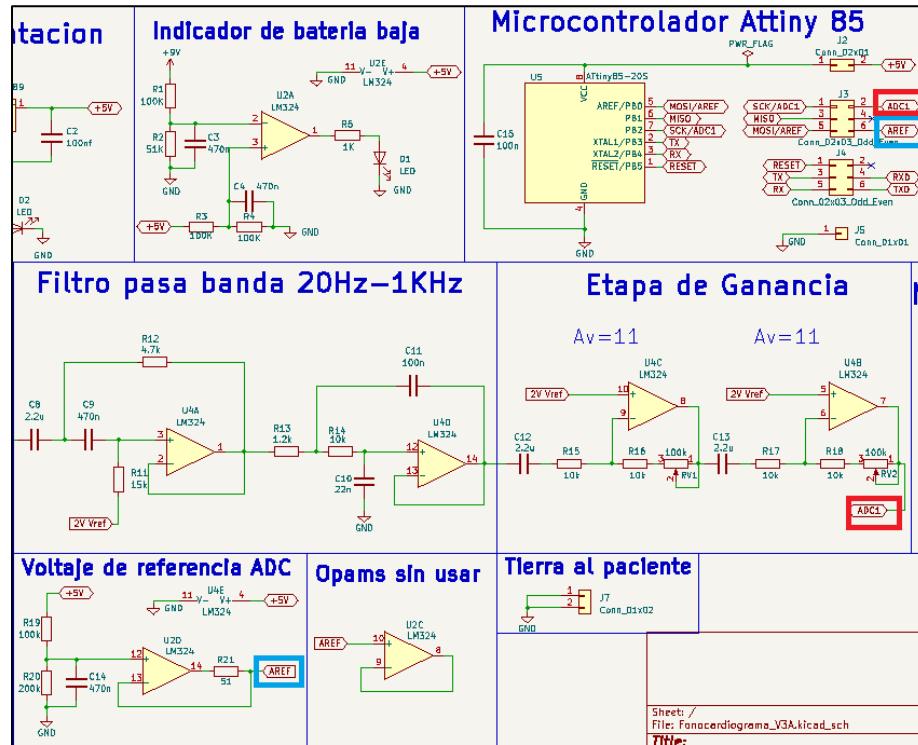


Figura 51.- Conexiones del microcontrolador al circuito de acondicionamiento, (módulo A).

En la [Figura 51](#) se tiene:

- De color rojo la conexión en entre el circuito de acondicionamiento y el microcontrolador.
- De color azul la conexión entre el voltaje de referencia y el microcontrolador.

10.8.3 Conexión al módulo Bluetooth

Algunos módulos Bluetooth poseen una etapa de acoplamiento integrada ya que los voltajes de trabajo típicos para un microcontrolador son de 0V a 5V, mientras que los voltajes de trabajo del módulo Bluetooth van desde 0V hasta 3.3V. La versión de HC-05 usada en este proyecto no posee un acoplamiento de voltajes, por lo cual es necesario implementarla para no dañar el dispositivo.

Según la sección [8.9 Microcontrolador Attiny 85](#), los voltajes de trabajo del attiny85 son:

- Entrada baja -> [-0.5 , 1.5]V
- Entrada alta -> [3.0 , 5.5]V
- Salida baja -> [0.0 , 0.6]V
- Salida alta -> [4.3 , 5.0]V

El voltaje de salida bajo del HC-05 se encuentra dentro del rango de valores del attiny85 para ser reconocido como entrada baja, de igual forma el voltaje de salida alto logra entrar en el rango de valores de entrada alta del microcontrolador, por lo cual no es necesario agregar un acoplamiento entre TXD del HC-05 y RX del attiny85.

El voltaje de entrada bajo del HC-05 se encuentra en el mismo rango de valores que el de salida baja del attiny85, sin embargo, el voltaje de entrada alto del módulo Bluetooth se encuentra por debajo del voltaje de salida alto del microcontrolador, por lo cual es necesario disminuir este voltaje para evitar que el HC-05 pueda dañarse. Para esto se utilizará un divisor de tensión.

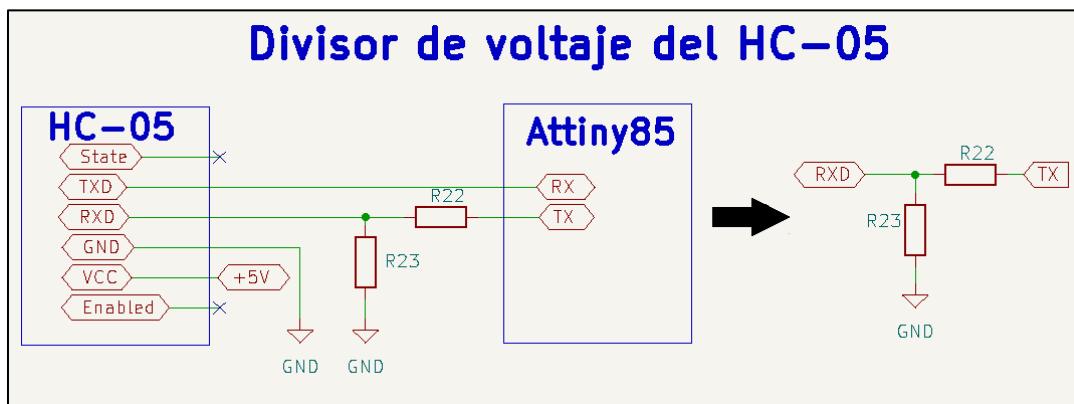


Figura 52.- Divisor de voltaje para el acoplamiento entre el attiny85 y el HC-05, (módulo A).

Se sabe que el pin TX del attiny 85 tendrá un voltaje de 5V y que el voltaje de trabajo del HC-05 en el pin RXD es de 3.3V. Con base en el divisor de la [Figura 52](#), se tiene:

$$R_{XD} = \frac{R_{23}}{R_{22} + R_{23}} T_X \rightarrow 3.3V = \frac{R_{23}}{R_{22} + R_{23}} 5V \rightarrow 3.3R_{22} + 3.3R_{23} = 5R_{23}$$

$$R_{22} = \frac{1.7}{3.3} R_{23} \rightarrow R_{22} \approx \frac{1}{2} R_{23}$$

El valor de R22 debe ser ≈ 0.5 veces el valor de R23. Se propone R22=100K Ω y R23=200K Ω .

Finalmente, el acoplamiento del módulo Bluetooth puede ser visto a continuación ([Figura 53](#)).

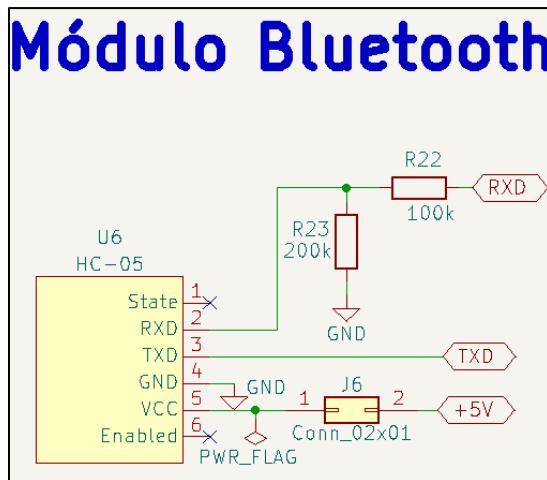


Figura 53.- Conexiones de acoplamiento del módulo Bluetooth, (módulo A).

Se agrega una terminal doble entre la conexión Vcc a +5V del HC-05 para apagar o encender el módulo independientemente del circuito de acondicionamiento, esto con fines de pruebas al ensamblar el circuito impreso.

Las conexiones para configurar el módulo Bluetooth por separado, así como los comandos AT para programarlo serán discutidos en la sección [15 Programación del módulo Bluetooth HC-05 / EGBT-045MS](#).

10.9 Conexión de tierra al paciente

Para eliminar cualquier tipo de perturbación que pueda ser inducida por el paciente, se propone conectar un cable hacia GND, logrando referenciar la señal del circuito.

Según la hoja de datos del amplificador de instrumentación AD620 [12], para un circuito de aplicación biomédica ECG, se recomienda conectar una tierra virtual y una etapa de protección entre el circuito y los electrodos del paciente ([Figura 54](#)).

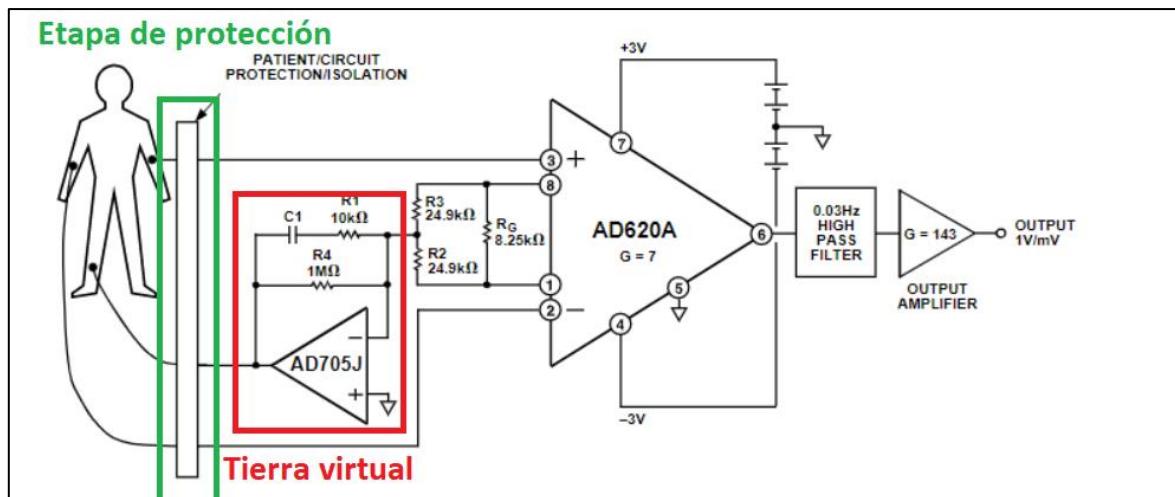


Figura 54.- Circuito de monitor ECG, (módulo A).

Debido a las limitaciones de los amplificadores LM324 utilizados en el circuito de acondicionamiento, no es posible conseguir una salida hacia GND con una alimentación unipolar. Para generar una tierra virtual sería necesario integrar una nueva etapa al circuito que permita alimentaciones bipolares o modificar los valores previamente calculados para agregar esta nueva referencia. Dadas las complicaciones no se integrará la etapa de tierra virtual, sin embargo, la etapa de protección entre GND y el paciente puede ser generada.

Se propone utilizar una pulsera antiestática conectada desde GND hasta alguna extremidad del paciente. La pulsera cuenta con una impedancia de aproximadamente $1M\Omega$, permitiendo una protección de corto circuito y a su vez, referenciar el circuito con el paciente, eliminando cualquier tipo de ruido que este pueda generar.

Para conectar la pulsera se hace uso de un conector JST XH2.54 el cual está conectado directamente a tierra ([Figura 55](#)).

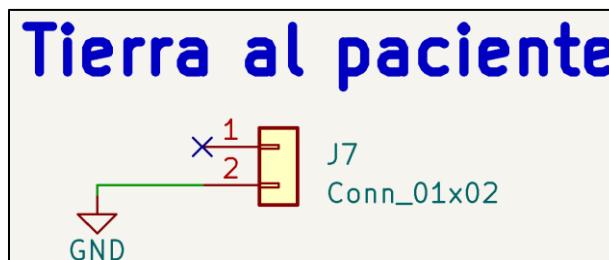


Figura 55.- Tierra al paciente, (módulo A).

11 Módulo B: Control de ganancia automático

MAX9814

El módulo B ([Figura 56](#), [Figura 57](#) y [Figura 58](#)) cuenta con las siguientes secciones:

- 11.1 Fuente de alimentación
 - 11.1.1 Regulador de voltaje L78L05
 - 11.1.2 Interruptor e indicador led de encendido
- 11.2 Indicador de batería baja
 - 11.2.1 Voltaje en la terminal negativa (Divisor de voltaje 9V)
 - 11.2.2 Voltaje en la terminal positiva (Divisor de voltaje 5V)
- 11.3 Pre-amplificador del micrófono MAX9814
 - 11.3.1 Microphone Bias Resistor
 - 11.3.2 Bias Capacitor
 - 11.3.3 Attack Time and Release Time
 - 11.3.4 Ajuste de offset
 - 11.3.5 Power-Supply Bypassing
 - 11.3.6 Threshold Voltaje
 - 11.3.7 Capacitor de acople a la salida
 - 11.3.8 Ganancia del circuito
 - 11.3.9 Modificaciones físicas al módulo de Adafruit
- 11.4 Voltaje de referencia offset
- 11.5 Filtro pasa banda 20Hz-1KHz
 - 11.5.1 Filtro pasa altas 20Hz
 - 11.5.2 Filtro pasa bajas 1KHz
- 11.6 Etapa de ganancia
- 11.7 Microcontrolador attiny 85
 - 11.7.1 Voltaje de referencia ADC
- 11.8 Conexiones del attiny 85
 - 11.8.1 Conexión al programador
 - 11.8.2 Conexión al circuito de acondicionamiento
 - 11.8.3 Conexión al módulo Bluetooth
- 11.9 Conexión de tierra al paciente

Nota: Para la programación de la placa, ver la sección: [13 Arduino como ISP](#), [14 Programación del attiny85](#) y [15 Programación del módulo Bluetooth HC-05 / EGBT-045MS](#)

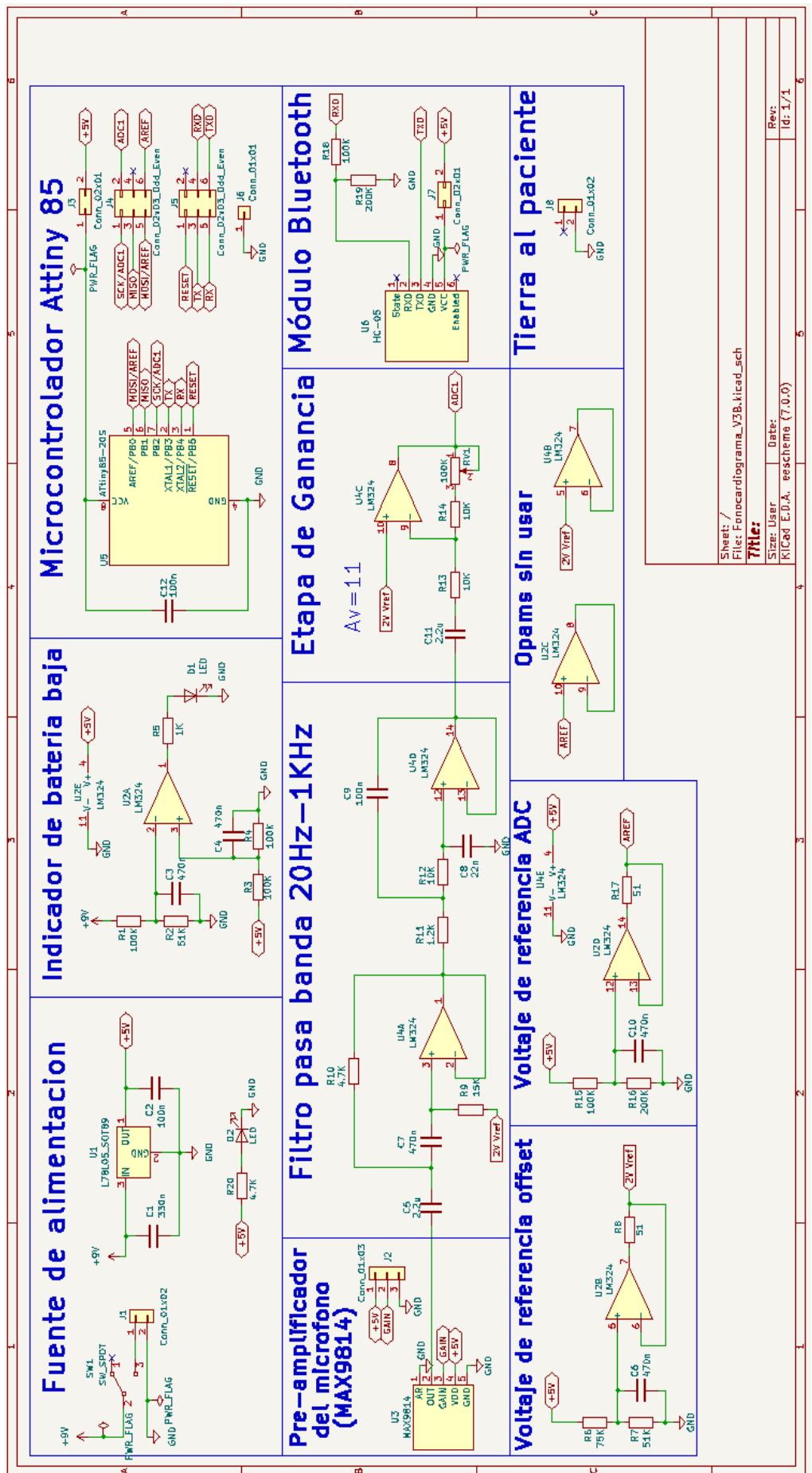


Figura 56.- Diagrama eléctrico, (módulo B).

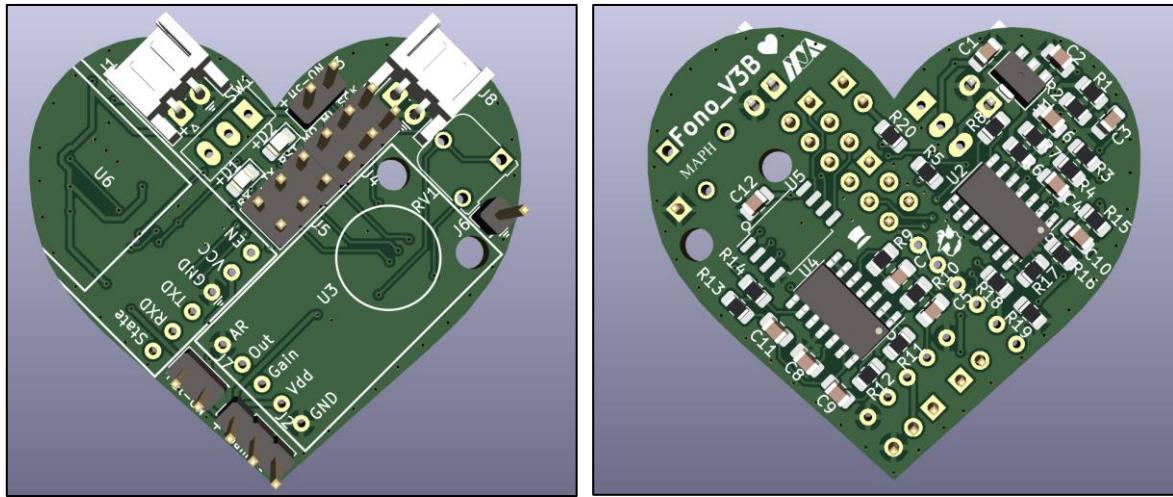


Figura 57.- Izquierda: Modelo 3D de la placa, vista frontal (módulo B).
Derecha: Modelo 3D de la placa, vista trasera (módulo B).

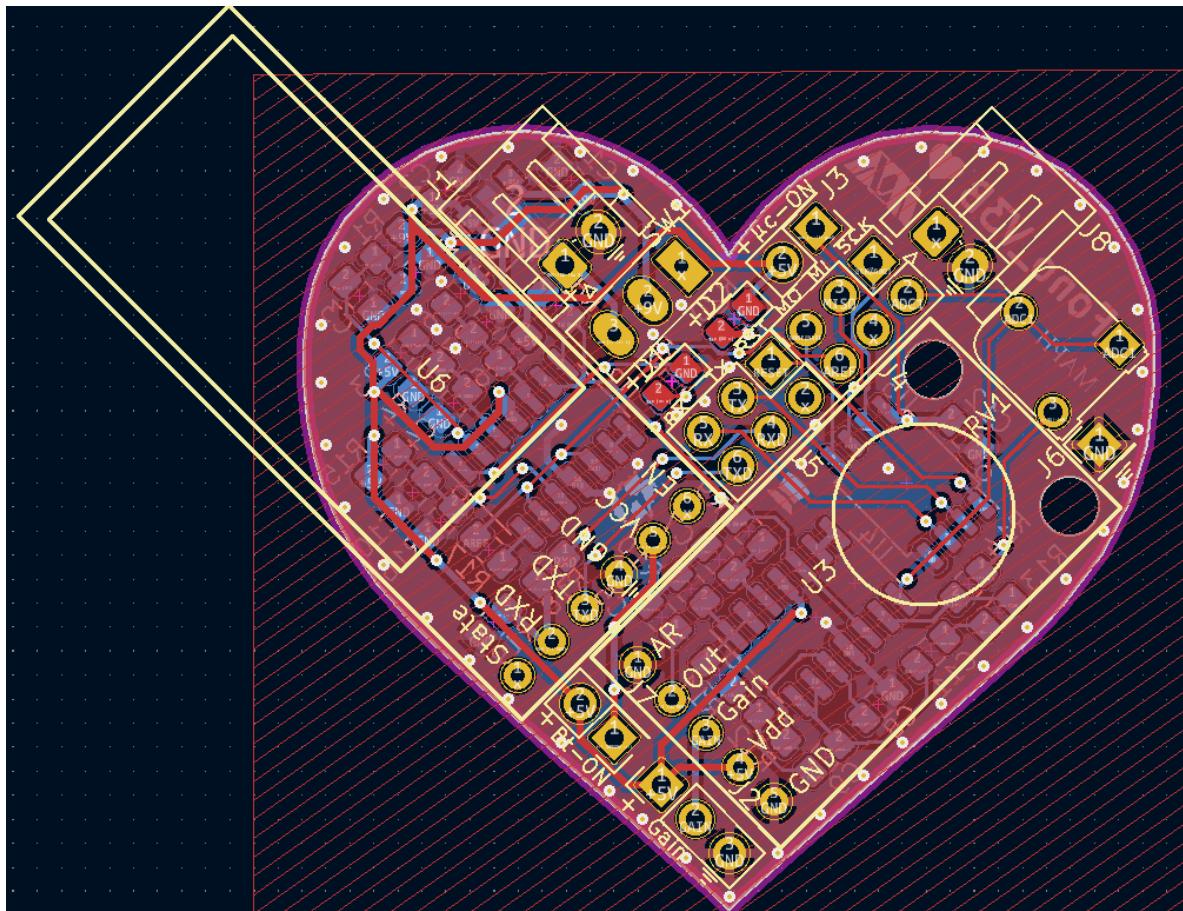


Figura 58.- Diagrama del PCB, (módulo B).

11.1 Fuente de alimentación

11.1.1 Regulador de voltaje L78L05

Tanto el circuito de acondicionamiento como el microcontrolador y el módulo de Bluetooth HC-05 se encuentran diseñados para trabajar con una alimentación unipolar de 5V, por lo cual se hace uso de un L78L05 y una batería de 9V para alimentar el circuito. ([Figura 59](#)).

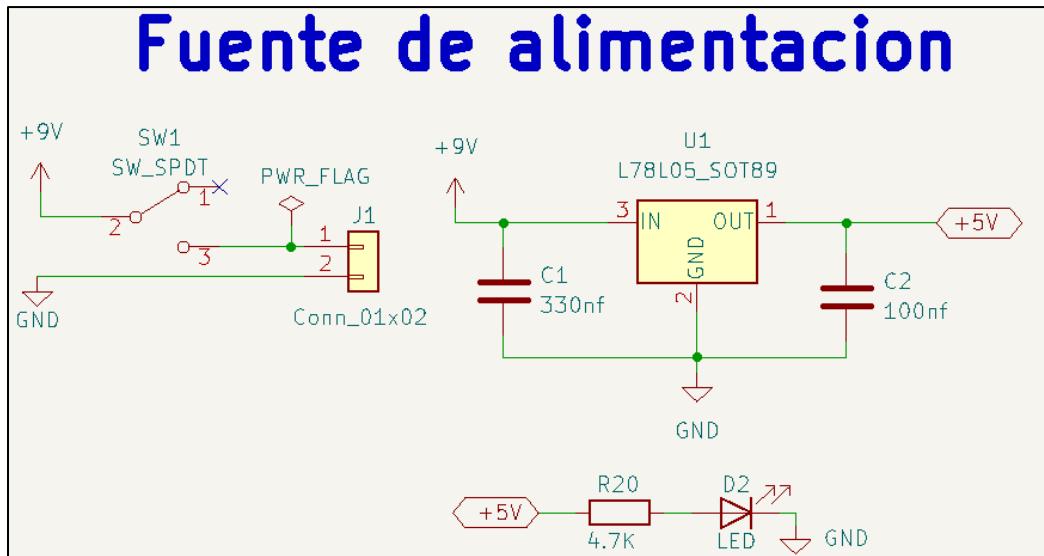


Figura 59.- Diagrama de conexiones L78L05, (módulo B).

Se eligió un L78L05 ya que el consumo máximo visto en las pruebas de desempeño es menor de 100mA, siendo ésta la corriente máxima entregada por el regulador ([8.1 Regulador de Voltaje positivo L78L05](#)).

El circuito utilizado para el regulador consta de un L78L05 y dos capacitores, este circuito fue obtenido de su hoja de datos ([Figura 60](#)). Estos capacitores se encargan de regular cualquier pico de voltaje que pueda presentarse tanto la batería de 9V como en la salida del regulador, manteniendo una regulación estable.

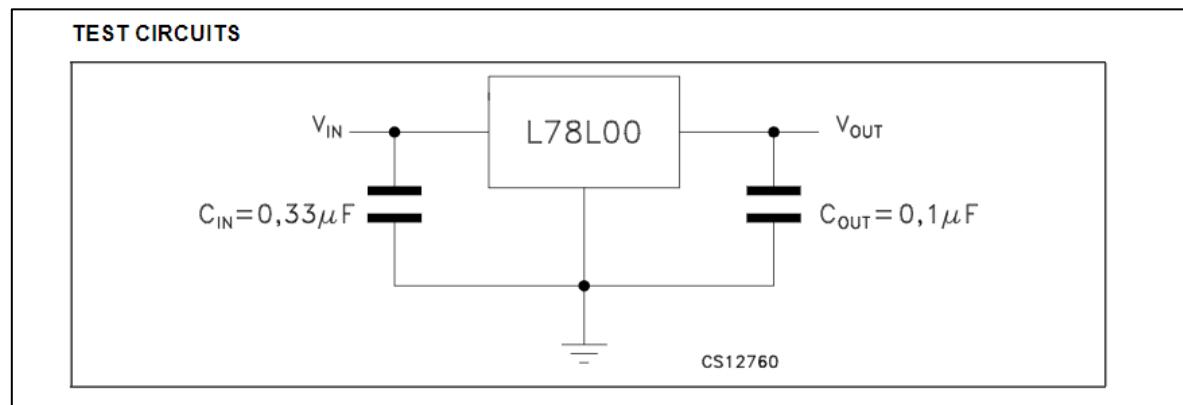


Figura 60.- Circuito de prueba, L78L05 [17], (módulo B).

11.1.2 Interruptor e indicador led de encendido

Se agrega un interruptor el cual permita encender o apagar el circuito sin necesidad de desconectar la batería de 9V. De igual forma, se agrega un led a la salida del regulador con el fin de ser un indicador cuando el circuito se encuentre encendido ([Figura 61](#)).

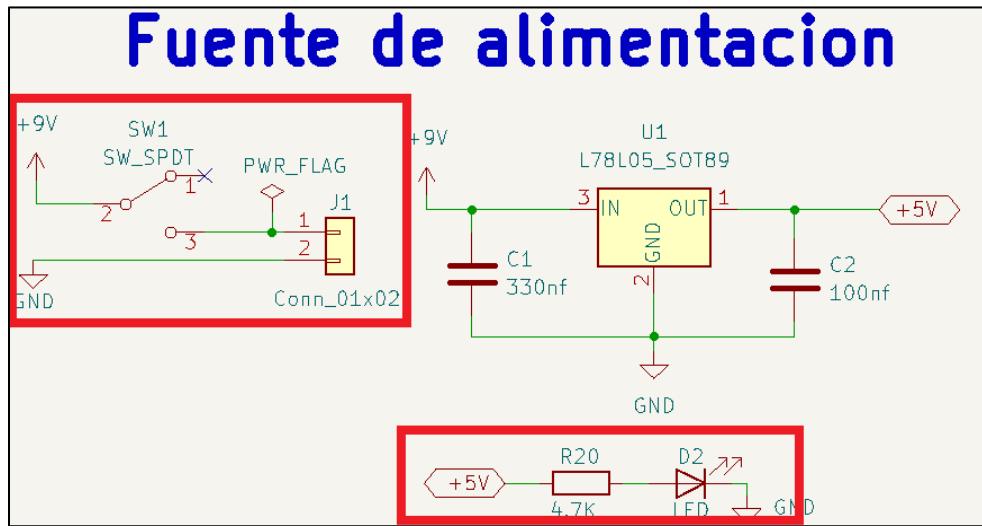


Figura 61.- Interruptor e indicador luminoso de encendido, (módulo B).

11.2 Indicador de batería baja

Para evitar un comportamiento errático debido a la caída de voltaje del desgaste de la batería, es necesario contar con un indicador de batería baja.

El indicador consta de un LM324 utilizado como comparador, dos divisores de voltaje y un led conectado en la salida del comparador ([Figura 62](#)).

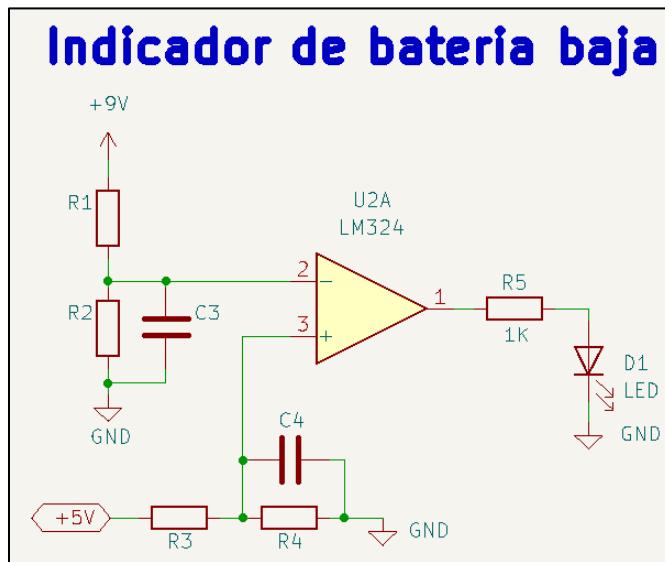


Figura 62.- Circuito indicador de batería baja, (módulo B).

El comparador es alimentado por el voltaje regulado del L78L05, es decir, se tiene una alimentación unipolar de 5V. Según lo propuesto en la sección [8.7 Amplificador operacional LM324](#), los voltajes de entrada en el comparador se encuentran en el rango [1V,3V].

Según lo propuesto en la sección [8.1 Regulador de Voltaje positivo L78L05](#), el voltaje mínimo para la operación del circuito es de 7.5V.

Dadas las especificaciones anteriores, mientras el voltaje de la batería se encuentre en el rango de [7.5 V, 9V], la salida del comparador estará referenciada al voltaje de alimentación negativo, cuando el voltaje de la batería decaea por debajo de 7.5V, la salida del comparador cambiara hacia la alimentación positiva, teniendo un voltaje de 3.5V en su salida encendiendiendo el indicador led.

11.2.1 Voltaje en la terminal negativa (Divisor de voltaje 9V)

El primer divisor de voltaje se encuentra referenciado a la batería de 9V, este divisor debe reducir el voltaje de tal forma que se tenga un valor en la entrada del comparador en el rango de [1V,3V]. De igual forma, cuando el voltaje de la batería se encuentre alrededor de 7.5V, el voltaje en el divisor debe estar contenido en este rango de valores ([Figura 63](#)).

Indicador de batería baja, terminal negativa

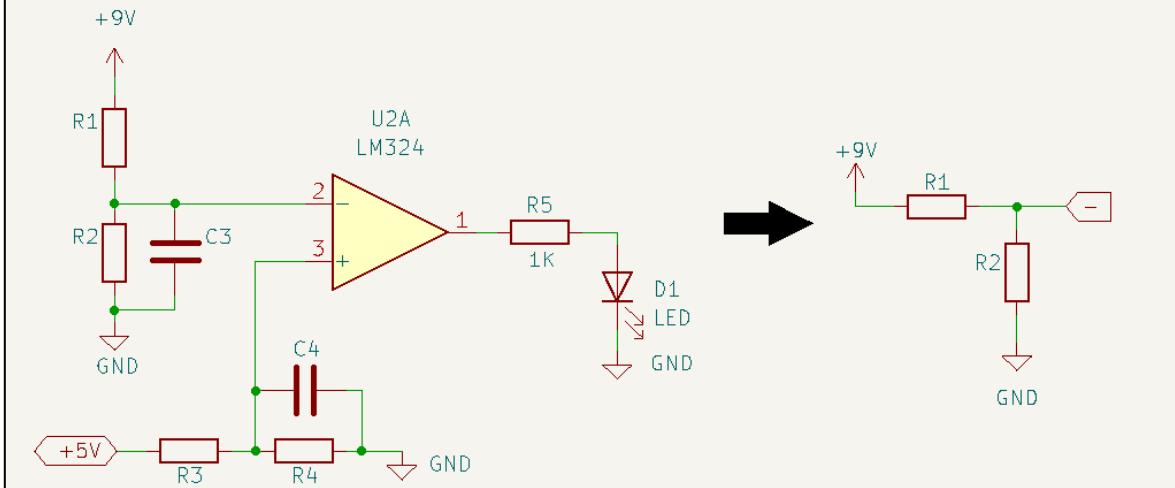


Figura 63.- Divisor de voltaje de 9V, terminal (-) del comparador, (módulo B).

Se propone tener un voltaje de 3V en el divisor cuando la batería de 9V se encuentra en su carga máxima, teniendo que:

$$V_{out} = \frac{R_2}{R_1 + R_2} V_{in} \rightarrow 3V = \frac{R_2}{R_1 + R_2} 9V \rightarrow 3R_1 + 3R_2 = 9R_2 \rightarrow R_1 = \frac{6}{2} R_2 \rightarrow R_1 = 2R_2$$

El valor de R1 debe ser 2 veces el valor de R2. Se propone usar un valor de R1=100KΩ y R2=51KΩ

Teniendo estos valores, se hace el cálculo cuando el voltaje en la batería es de 7.5V, este dato será usado para calcular el segundo divisor de voltaje.

$$V_{out} = \frac{R_2}{R_1 + R_2} V_{in} \rightarrow V_{out} = \frac{R_2}{R_1 + R_2} 7.5V \rightarrow V_{out} = \frac{51K\Omega}{100K\Omega + 51K\Omega} 7.5V = \frac{51K\Omega}{151K\Omega} 7.5V$$

$$V_{out} = 2.53V$$

Con los voltajes calculados se sabe que:

- Voltaje en la batería = 9V -> Voltaje en la terminal negativa = 3V
- Voltaje en la batería = 7.5V -> Voltaje en la terminal negativa = 2.5V

Por último, se agrega el capacitor C3, el cual filtra el ruido térmico y cualquier tipo de perturbación que pueda tener la fuente de alimentación.

El capacitor C3 y las resistencias R1 y R2 crean un filtro pasa bajas. La frecuencia de corte de este filtro deberá ser inferior a 20Hz, esto con el fin de no afectar el desempeño del circuito de audio. Proponiendo un capacitor de 470nF, se tiene:

$$f_c = \frac{1}{2\pi(100K\Omega||51K\Omega)470nf} = 10.02Hz$$

11.2.2 Voltaje en la terminal positiva (Divisor de voltaje 5V)

Para obtener un cambio en la polaridad de la salida del comparador es necesario introducir un voltaje de referencia similar al calculado en la sección [11.2.1 Voltaje en la terminal negativa \(Divisor de voltaje 9V\)](#) cuando el voltaje en la batería de 9V decae a su valor mínimo permitido de 7.5V. Siendo el voltaje calculado en la sección anterior de 2.5V, este será el valor de voltaje a introducir en la terminal positiva del comparador, de tal forma que, cuando el voltaje en la batería disminuya, el voltaje en la terminal negativa del comparador también disminuye, pasando por debajo del voltaje de referencia de 2.5V, al tener un voltaje mayor en la terminal positiva que en la terminal negativa, el voltaje en la salida del comparador cambia a ser referenciado hacia el voltaje de alimentación, encendiendo el led ([Figura 64](#)).

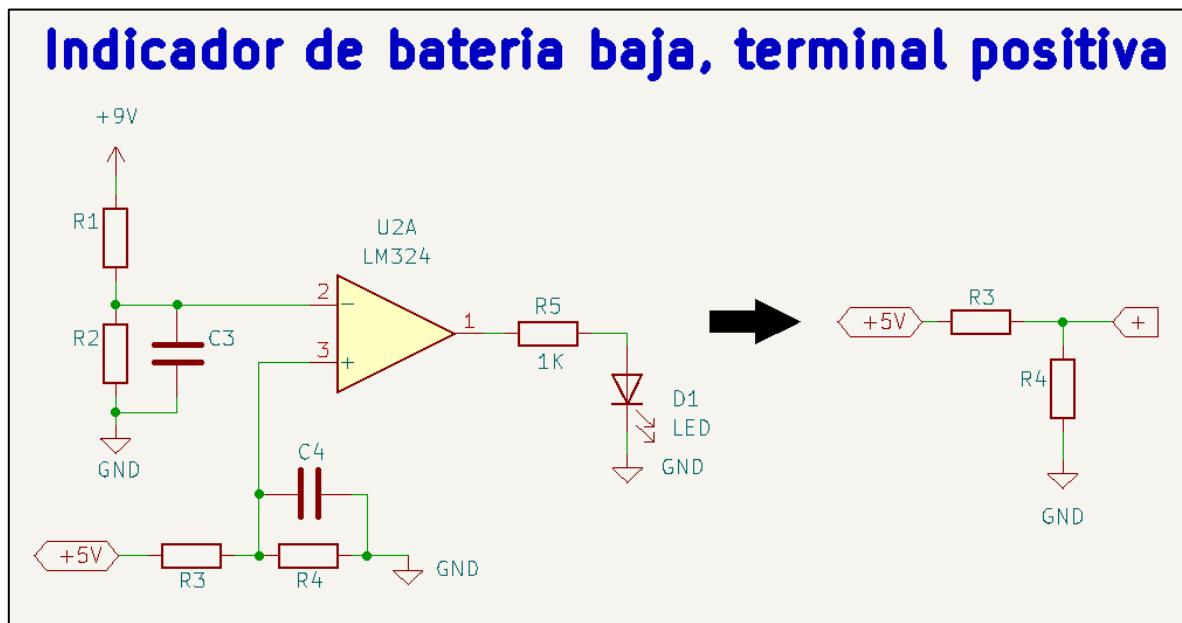


Figura 64.- Divisor de voltaje de 5V, terminal (+) del comparador, (módulo B).

Este segundo divisor necesita un voltaje fijo de operación, el cual será tomado de la salida del regulador L78L05:

$$V_{out} = \frac{R_4}{R_3 + R_4} V_{in} \rightarrow 2.5V = \frac{R_4}{R_3 + R_4} 5V \rightarrow 2.5R_3 + 2.5R_4 = 5R_4 \rightarrow R_3 = \frac{2.5}{2.5} R_4 \rightarrow R_3 = R_4$$

El valor de las resistencias R3 y R4 debe ser el mismo, se propone $R_3=R_4=100\text{ K}\Omega$

De igual forma que en el divisor de 9V, se agrega un capacitor C4 para filtrar el ruido térmico y cualquier tipo de perturbación que pueda tener la fuente de alimentación.

El capacitor C4 y las resistencias R3 y R4 crean un filtro pasa bajas. La frecuencia de corte de este filtro deberá ser inferior a 20Hz, esto con el fin de no afectar el desempeño del circuito de audio. Proponiendo un capacitor de 470nF, se tiene:

$$f_c = \frac{1}{2\pi(100K\Omega||100K\Omega)470nf} = 6.77\text{Hz}$$

Finalmente, el circuito queda de la siguiente forma:

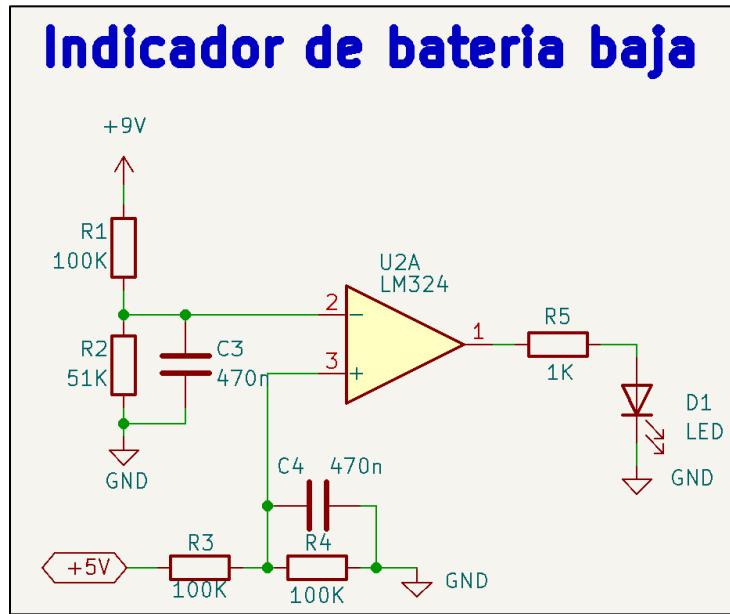


Figura 65.- Circuito indicador de batería baja (con valores), (módulo B).

11.3 Pre-amplificador del micrófono MAX9814

Según la sección [8.6 Pre-amplificador de micrófono MAX9814](#), el tipo de empaquetado para el MAX9814 es el TDFN, debido a la complejidad de integrar este componente a la placa, se decidió hacer uso de la placa de Adafruit ([8.5 Módulo MAX9814 de Adafruit](#)) la cual ya cuenta con el MAX9814 soldado. Para ajustar el comportamiento del módulo a las necesidades de la placa, se desoldarán ciertos componentes que trae el módulo de Adafruit.

11.3.1 Microphone Bias Resistor

Según la hoja de datos del MAX9814 [15], se recomienda usar una resistencia de 2.2KΩ para la polarización del micrófono, siendo común en aplicaciones de micrófonos Electret. ([Figura 66](#)).

Microphone Amplifier with AGC and Low-Noise Microphone Bias

Setting the AGC Threshold

To set the output-voltage threshold at which the microphone output is clamped, an external resistor-divider must be connected from MICBIAS to ground with the output of the resistor-divider applied to TH. The voltage V_{TH} determines the peak output-voltage threshold at which the output becomes clamped. The maximum signal swing at the output is then limited to two times V_{TH} and remains at that level until the amplitude of the input signal is reduced. To disable AGC, connect TH to MICBIAS.

Microphone Bias Resistor

MICBIAS is capable of sourcing 20mA. Select a value for RMICBIAS that provides the desired bias current for the electret microphone. A value of 2.2kΩ is usually sufficient for a microphone of typical sensitivity. Consult the microphone data sheet for the recommended bias resistor.

Choose C_{IN} such that f_{-3dB_IN} is well below the lowest frequency of interest. Setting f_{-3dB_IN} too high affects the amplifier's low-frequency response. Use capacitors with low-voltage coefficient dielectrics. Aluminum electrolytic, tantalum, or film dielectric capacitors are good choices for AC-coupling capacitors. Capacitors with high-voltage coefficients, such as ceramics (non-C0G dielectrics), can result in increased distortion at low frequencies.

Output Capacitor

The output of the MAX9814 is biased at 1.23V. To eliminate the DC offset, an AC-coupling capacitor (C_{OUT}) must be used. Depending on the input resistance (R_L) of the following stage, C_{OUT} and R_L effectively form a highpass filter. The -3dB point of the highpass filter, assuming zero output impedance, is given by:

MAX9814

Figura 66.- Microphone Bias Voltaje [15], (módulo B).

Para extraer las señales eléctricas producidas por el micrófono, se utiliza un capacitor de acople. Según el diagrama proporcionado por Adafruit ([Figura 12](#)), la resistencia utilizada en la polarización del micrófono es de 2.2KΩ, esta resistencia junto con el capacitor de acople forma un filtro pasa altas.

La señal de interés del corazón se encuentra en un rango de frecuencias de 20Hz a 1KHz ([Figura 4](#)), por lo cual, se propone una frecuencia de corte de 15Hz, teniendo la siguiente ecuación:

$$C = \frac{1}{2\pi R f_c} = \frac{1}{2\pi(2.2K\Omega)(15Hz)} = 4.8\mu F \rightarrow 4.4\mu F$$

Dicho esto, se modificará el capacitor de acople del módulo de Adafruit, pasando de un valor de 0.1uF a dos capacitores en paralelo de 2.2uF.

11.3.2 Bias Capacitor

Según la hoja de datos del MAX9814 [15], se recomienda utilizar un capacitor de polarización de 470nF conectado a tierra ([Figura 67](#)), el modelo visto en la [Figura 12](#) de Adafruit ya cuenta con este capacitor, por lo cual no se modificará.

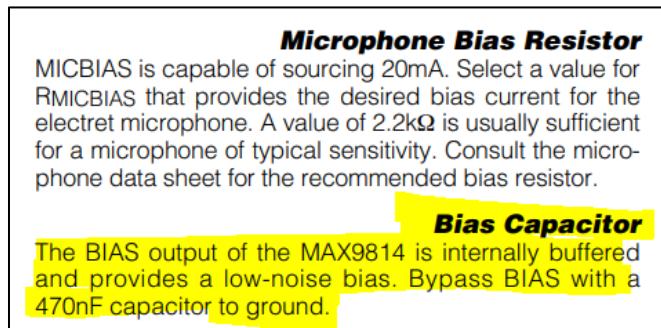


Figura 67.- Bias Capacitor [15], (módulo B).

11.3.3 Attack Time and Release Time

Para ajustar los tiempos de activación (Attack Time) y el tiempo que le toma al control de ganancia automático de regresar a su valor normal de operación (Release Time), se utiliza el capacitor CT y la terminal A/R.

En este caso, se requiere que tanto el “Attack Time” y el “Release Time” sean lo más rápido posible permitiendo disminuir las señales generadas por golpes accidentales.

Según la hoja de datos del MAX9814 [15], se propone utilizar un capacitor de 22nF y conectar el pin A/R a GND, teniendo un “Attack Time” de 0.05ms con un “Release” Time de 25ms ([Figura 68](#)).

The attack and release times can be selected by utilizing the corresponding capacitances listed in Table 2.				
CCT	tATTACK (ms)	tRELEASE (ms)		
		A/R = GND	A/R = V _{DD}	A/R = UNCONNECTED
22nF	0.05	25	100	200
47nF	0.11	55	220	440
68nF	0.16	80	320	640
100nF	0.24	120	480	960
220nF	0.53	265	1060	2120
470nF	1.1	550	2200	4400
680nF	1.63	815	3260	6520
1 μ F	2.4	1200	4800	9600

Figura 68.- Attack Time y Release Time [15], (módulo B).

El circuito proporcionado por Adafruit ([Figura 12](#)), cuenta con un capacitor CT de 100nF y con el pin A/R sin conectar, en este caso, se tiene que cambiar el capacitor y conectar el pin a GND.

11.3.4 Ajuste de offset

Según la hoja de datos [15], es necesario conectar un capacitor de 2.2uF entre la terminal CG y GND, esto con el fin de garantizar un offset de cero en la salida. ([Figura 69](#)).

Pin Description			
PIN		NAME	FUNCTION
TDFN	UCSP		
1	A1	CT	Timing Capacitor Connection. Connect a capacitor to CT to control the Attack and Release times of the AGC.
2	B2	SHDN	Active-Low Shutdown Control
3	A2	CG	Amplifier DC Offset Adjust. Connect a 2.2 μ F capacitor to GND to ensure zero offset at the output.
4, 11	—	N.C.	No Connection. Connect to GND.

Figura 69.- Offset Adjust [15], (módulo B).

El circuito proporcionado por Adafruit ([Figura 12](#)), ya cuenta con este capacitor, por lo que no es necesario realizar alguna modificación.

11.3.5 Power-Supply Bypassing

Según la hoja de datos del MAX9814 [15] se recomienda conectar un capacitor de bypass de 0.1uF entre la fuente de alimentación y GND ([Figura 70](#)). El circuito de Adafruit ([Figura 12](#)), cuenta con un capacitor de 2.2uF, por lo cual no se realizará alguna modificación.

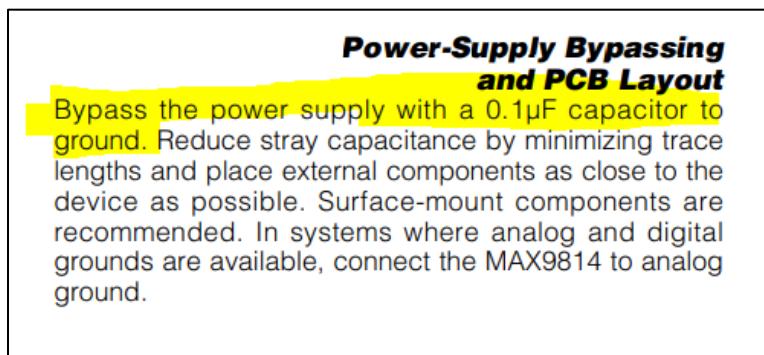


Figura 70.- Power-Supply Bypassing [15], (módulo B).

11.3.6 Threshold Voltaje

Dentro del módulo, existe la terminal “TH” a la cual se le suministra un voltaje constante, este voltaje es comparado internamente con respecto al voltaje de entrada. En caso de que el voltaje de entrada sea inferior a este voltaje de umbral, la señal será amplificada, caso contrario, la señal es atenuada ([Figura 71](#)).

Setting the AGC Threshold

To set the output-voltage threshold at which the microphone output is clamped, an external resistor-divider must be connected from MICBIAS to ground with the output of the resistor-divider applied to TH. The voltage V_{TH} determines the peak output-voltage threshold at which the output becomes clamped. The maximum signal swing at the output is then limited to two times V_{TH} and remains at that level until the amplitude of the input signal is reduced. To disable AGC, connect TH to MICBIAS.

Figura 71.- AGC Threshold Voltaje [15], (módulo B).

Para conocer este voltaje de umbral, es necesario probar el circuito de polarización del micrófono por sí solo. Se dessoldó el micrófono integrado en el módulo de Adafruit y se conectó el micrófono utilizado en el fonocardiograma (CMC-60158-42P). Una vez hecho esto, se midió la amplitud de la señal cardiaca antes de entrar al MAX9814.

Como puede observarse en la [Figura 72](#), la amplitud de la señal cardiaca es de 50mVp, se tomará este voltaje como referencia para la señal V_{TH} .

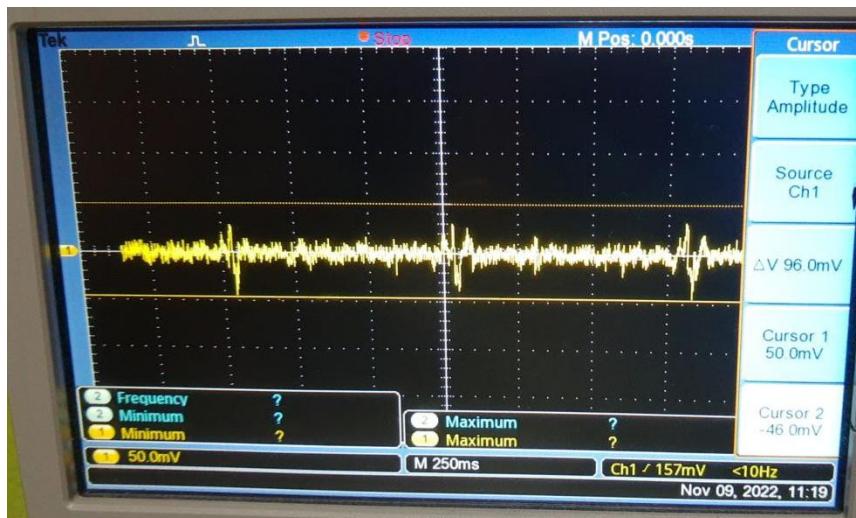


Figura 72.- Amplitud de la señal cardiaca antes del MAX9814, (módulo B).

Para tener un margen de seguridad respecto a la señal cardiaca, se propone un factor de seguridad de 2 unidades, es decir:

$$V_{th} = 2 * V_{cardiaco} = 100\text{mVp}$$

Una vez teniendo el valor de voltaje, se calcula el valor de las resistencias tomando como referencia el circuito de la [Figura 73](#).

$$V_{TH} = \frac{R_{TH2}}{R_{TH1} + R_{TH2}} V_{MICBIAS} \rightarrow 100\text{mVp} = \frac{R_{TH2}}{R_{TH1} + R_{TH2}} 2\text{V} \rightarrow 0.05(R_{TH1} + R_{TH2}) = R_{TH2}$$

$$0.05R_{TH1} = 0.95R_{TH2} \rightarrow R_{TH1} = 19R_{TH2}$$

El valor de RTH1 debe ser 19 veces mayor a RTH2, se propone RTH1 de 100K Ω y RTH2 de 5.1K Ω

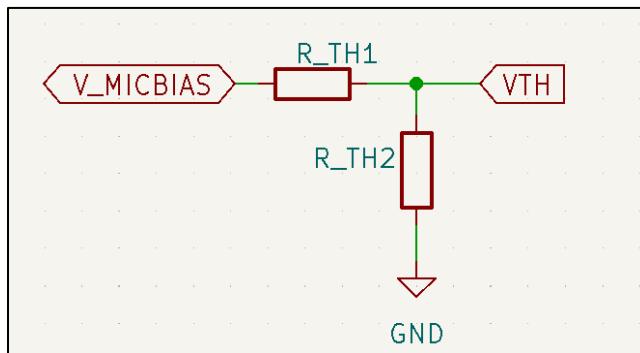


Figura 73.- Divisor de voltaje para VTH, (módulo B).

El circuito proporcionado por Adafruit ([Figura 12](#)) cuenta con una resistencia RTH1=150K Ω , RTH2=100k Ω , por lo cual, es necesario cambiar ambas resistencias.

11.3.7 Capacitor de acople a la salida

Por último, se propone usar un capacitor de 2.2uF en la salida del integrado para acoplar la señal sonora a las demás etapas del circuito. Este capacitor es externo al módulo de Adafruit.

11.3.8 Ganancia del circuito

La ganancia del circuito puede ser modificada conectando la terminal “Gain” a VDD, GND o dejándola desconectada.

VDD-> 40dB

GND->50dB

Desconectada->60dB

Realizando una prueba de funcionamiento sobre el circuito ([Figura 74](#)), al conectar la terminal de ganancia a GND, se tiene una señal con una variación de 1.56V, la cual se encuentra dentro de las cotas establecidas en la sección [8.7 Amplificador operacional LM324](#).



Figura 74.- Amplitud de la señal cardiaca, circuito de Adafruit modificado, (módulo B).

De igual forma, se dejará la opción para puentear la terminal “Gain” a las 3 configuraciones diferentes dentro del diseño final.

11.3.9 Modificaciones físicas al módulo de Adafruit

El diagrama modificado del módulo MAX9814 se muestra a continuación ([Figura 75](#)).

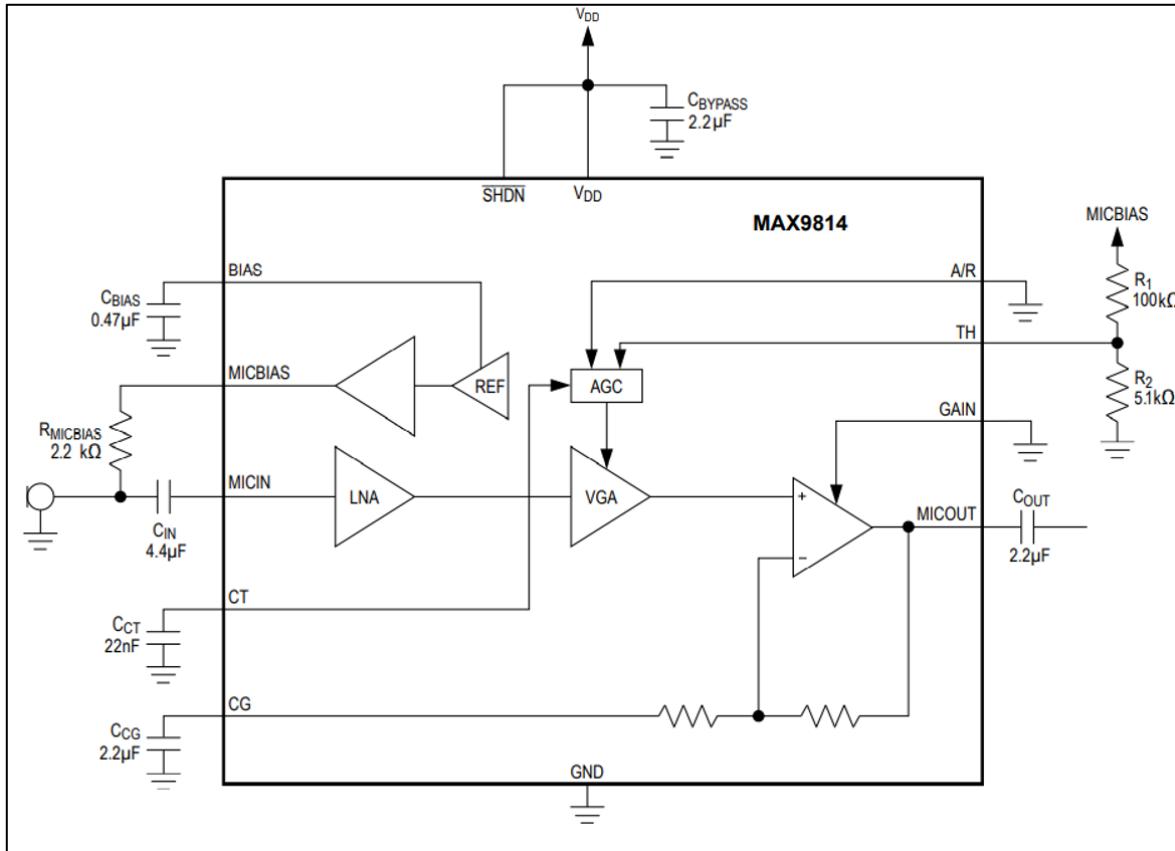


Figura 75.- Diagrama modificado del MAX9814, (módulo B).

En la [Figura 76](#), se encuentran marcados los componentes a modificar dentro del módulo de Adafruit, los cuales son:

- Resistencia 150K: La resistencia RTH1 en el divisor de voltaje visto en la terminal VTH.
- Resistencia 100K: La resistencia RTH2 en el divisor de voltaje visto en la terminal VTH.
- Capacitor 100nF: Capacitor CT para ajustar los tiempos de “Attack Time” y “Release Time”.
- Capacitor 0.1uF: Capacitor de acople en la entrada del micrófono.

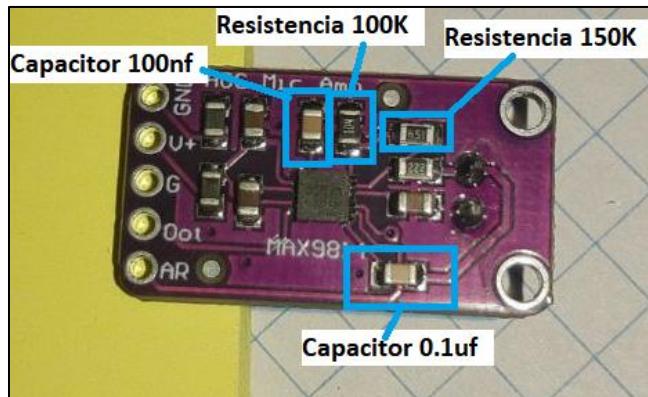


Figura 76.- Componentes a modificar en la placa de Adafruit MAX9814, (módulo B).

En la [Figura 77](#), se encuentran marcados los componentes modificados del módulo de Adafruit, los cuales son:

- Resistencia 100K: La resistencia RTH1
- Resistencia 5.1K: La resistencia RTH2
- Capacitor 22nF: Capacitor CT
- Capacitor 4.4uF: Capacitor de acople del micrófono.

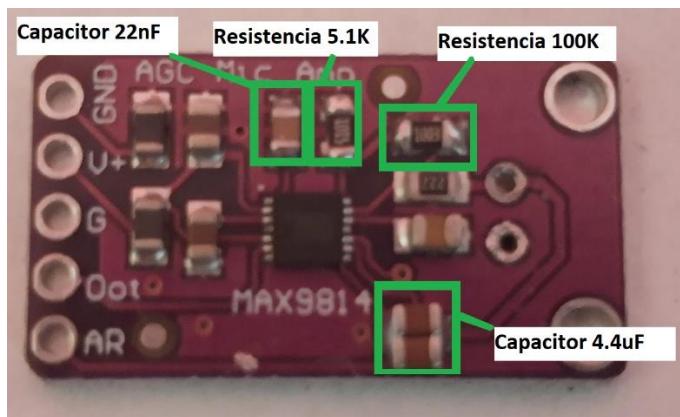


Figura 77.- Componentes modificados en la placa de Adafruit MAX9814, (módulo B).

11.4 Voltaje de referencia offset

Es necesario introducir las variaciones de voltaje AC provenientes del MAX9814 a las etapas posteriores del circuito, para ello se utiliza una componente de offset ya que el circuito está diseñado para trabajar con voltajes positivos. Esta componente puede ser agregada en la terminal no inversora de los opamps usando un divisor de tensión ([Figura 78](#)).

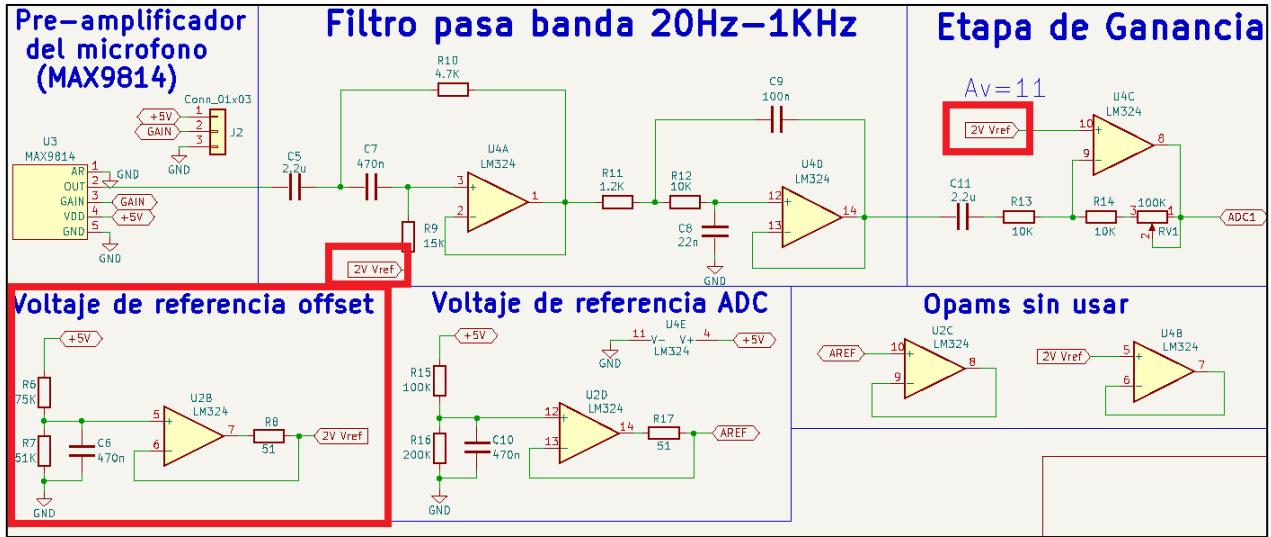


Figura 78.- Red de polarización de opamps, (módulo B).

Según los voltajes de operación discutidos en la sección [8.7 Amplificador operacional LM324](#), se tiene como límite superior 3V e inferior 1V. Dado esto, la componente de offset deberá ubicarse entre estos valores de tal forma que se tenga el mismo rango de voltaje hacia los límites. Se propone una componente de offset de 2V teniendo $\pm 1V$ de variación.

Para calcular las resistencias se simplificará el circuito, teniendo un divisor de voltaje simple ([Figura 79](#)).

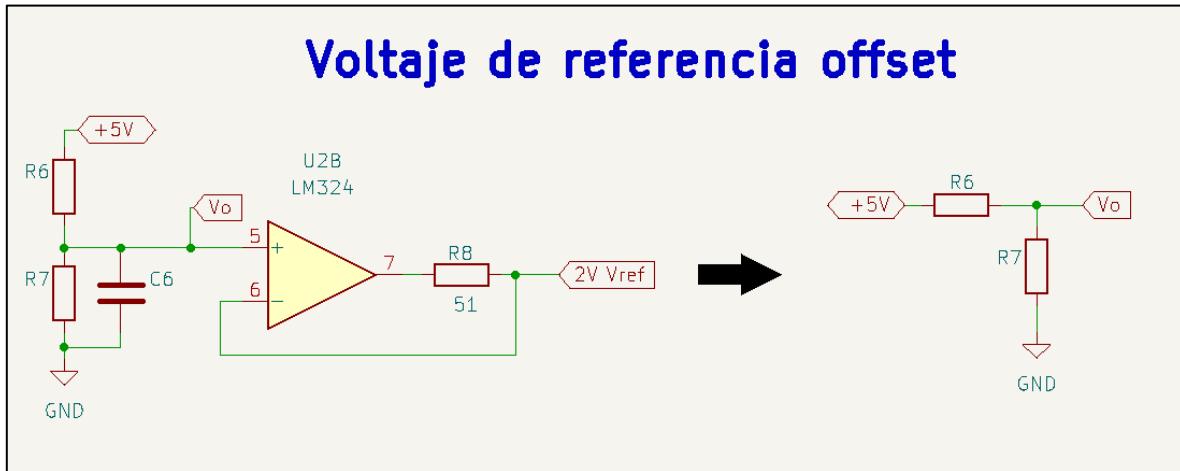


Figura 79.- Divisor de tensión, voltaje de referencia offset, (módulo B).

Conociendo que:

$$V_o = \frac{R_7}{R_6 + R_7} 5V \rightarrow 2V = \frac{R_7}{R_6 + R_7} 5V \rightarrow 2R_6 + 2R_7 = 5R_7 \rightarrow R_6 = \frac{3}{2} R_7$$

El valor de R6 debe ser 3/2 veces el valor de R7. Se propone usar un valor de R6=75KΩ y R7=51KΩ

Se agrega el capacitor C6, el cual filtra el ruido térmico y cualquier tipo de perturbación que pueda tener la fuente de alimentación.

El capacitor C6 y las resistencias R6 y R7 crean un filtro pasa bajas. La frecuencia de corte de este filtro deberá ser inferior a 20Hz para no afectar el desempeño del circuito de audio. Proponiendo un capacitor de 470nF, se tiene:

$$f_c = \frac{1}{2\pi(75K\Omega||51K\Omega)470nf} = 11.15Hz$$

Finalmente, el circuito de voltaje de referencia offset puede ser visto a continuación ([Figura 80](#))

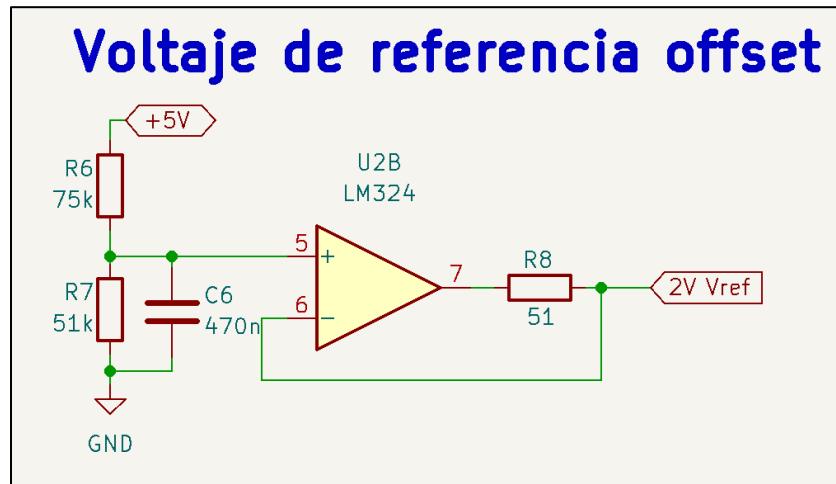


Figura 80.- Voltaje de referencia offset con valores, (módulo B).

11.5 Filtro pasa banda 20Hz-1KHz

Este filtro se divide en dos secciones:

- Filtro pasa altas 20Hz
- Filtro pasa bajas 1KHz

En ambos filtros se usan dos polos con una configuración Sallen-Key de tipo Butterworth. Los valores de los componentes fueron obtenidos gracias al software de “Filter Pro”. Los filtros obtenidos por este medio tienen como base una alimentación simétrica. Para pasar de una fuente dual a unipolar se hace uso de las notas de aplicación de Texas Instruments *“Single-supply, 2nd-order, Sallen-Key band-pass filter circuit”* [21].

Es necesario agregar una componente de offset en la entrada del filtro pasa altas, el circuito con esta modificación puede ser visto a continuación ([Figura 81](#)).

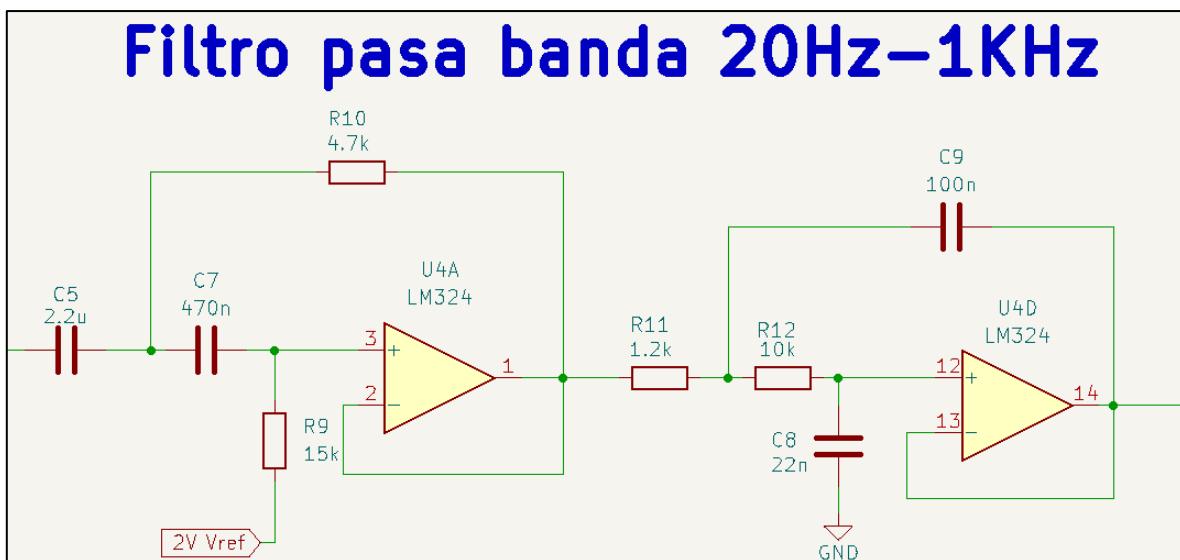
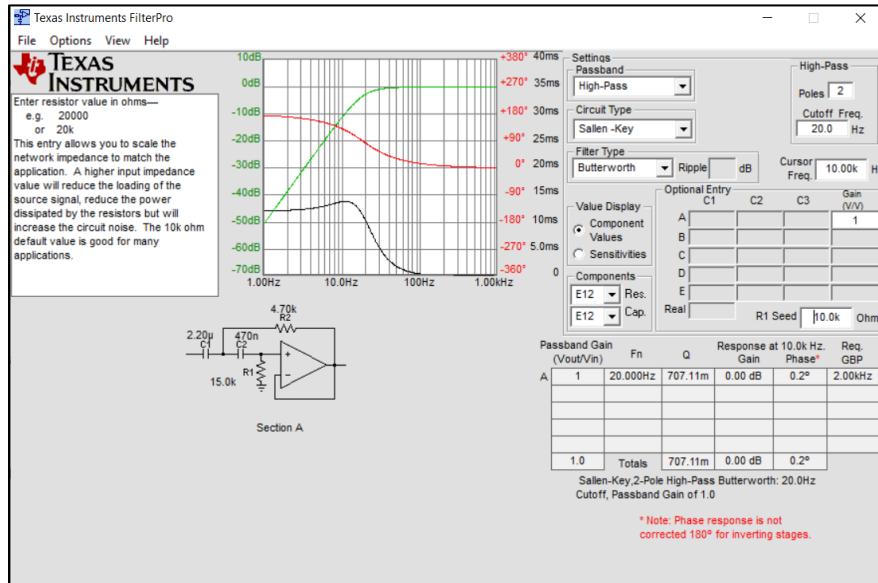


Figura 81.- Filtro pasa bandas 20Hz-1KHz, (módulo B).

De igual forma, se agrega un capacitor de acople C8 para introducir las variaciones de AC de partes anteriores del circuito en el filtro pasa bandas.

11.5.1 Filtro pasa altas 20Hz

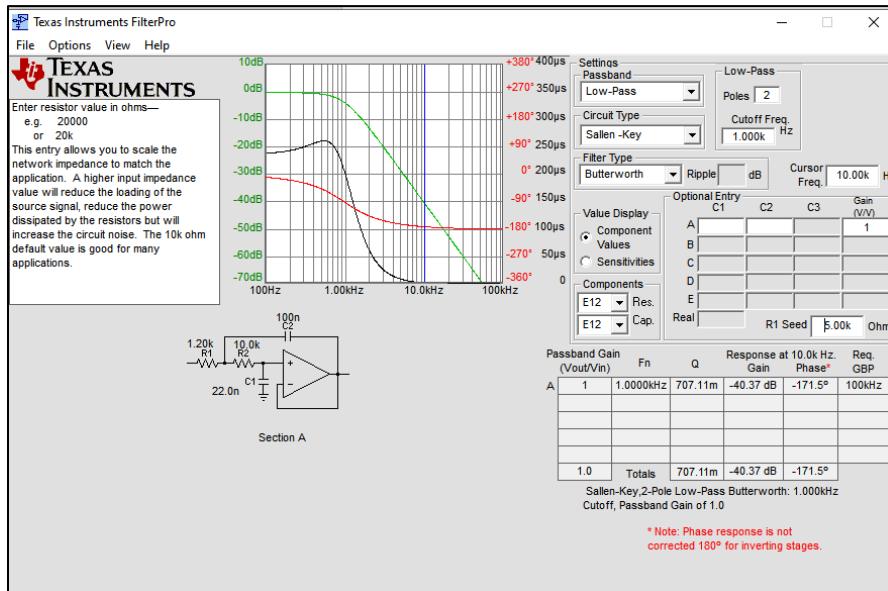
Para el filtro pasa altas, se eligió una frecuencia de corte de 20Hz. Según los datos proporcionados por el software, se necesita un amplificador con una ganancia ancho de banda mínima de 2KHz ([Figura 82](#)). Se propone usar un LM324 ya que posee un GBP de 1.2MHz.



[Figura 82.- Filtro pasa altas fc=20Hz, \(módulo B\).](#)

11.5.2 Filtro pasa bajas 1KHz

Para el filtro pasa bajas, la frecuencia de corte escogida es de 1Khz. Para realizar este filtro se necesita una ganancia-ancho de banda de 100Khz ([Figura 83](#)). Se propone usar el LM324 ya que su GBP es de 1.2MHz.



[Figura 83.- Filtro pasa bajas fc=1Khz, \(módulo B\).](#)

11.6 Etapa de ganancia

Por último, se amplifica la señal filtrada para hacerla coincidir con los rangos de voltaje establecidos en la sección [8.7 Amplificador operacional LM324](#). Según la sección [11.3.8 Ganancia del circuito](#), el voltaje de salida de la señal es de 1.54Vpp, ya que este voltaje es cercano a los 2Vpp propuestos en el rango de valores, se utilizará una ganancia máxima $A_v=1$ hasta $A_v=11$.

Según la hoja de datos del LM324 [16] la ganancia en lazo abierto en 1KHz es de 40dB o $A_v=100$ ([Figura 84](#)), por lo cual se utilizará una única etapa de ganancia ([Figura 85](#)).

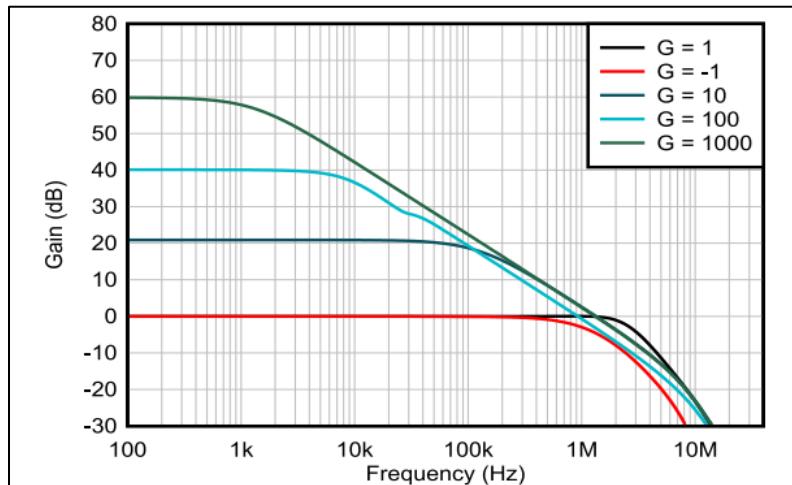


Figura 84.- Relación ganancia-ancho de banda en lazo abierto LM324, (módulo B).

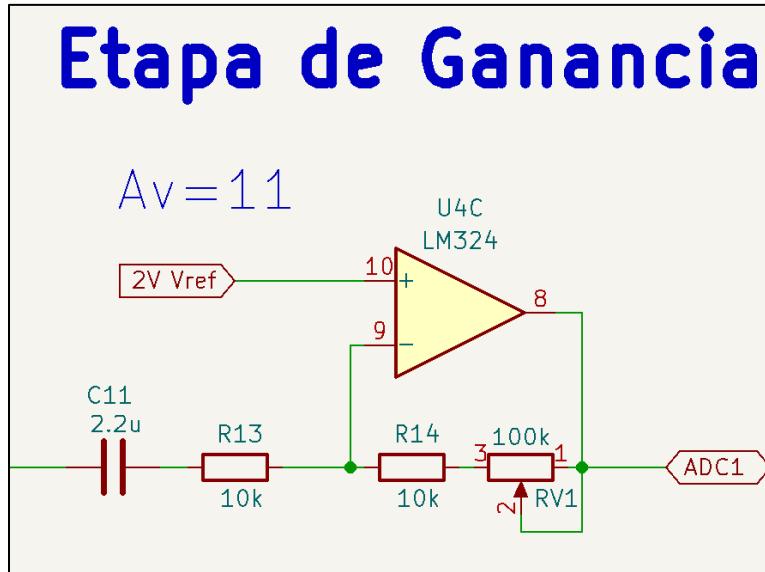


Figura 85.- Etapa de ganancia, (módulo B).

De manera similar a la sección [10.6 Etapa de ganancia](#), para obtener una ganancia unitaria al ajustar el potenciómetro a su valor mínimo (corto circuito), es necesario que el valor de la resistencia en el lazo de retroalimentación sea igual al de la resistencia de entrada de la señal, por lo cual se hace uso de la resistencia R14, la cual se mantiene fija en todo momento, teniendo que:

$$Av = -\frac{R_{14} + R_{V1}}{R_{13}} = -\frac{R_{14} + 0}{R_{13}} = -\frac{R_{14}}{R_{13}} = -1 \rightarrow R_{14} = R_{13}$$

Para calcular la ganancia variable de 11, se ajusta el potenciómetro al máximo valor, teniendo:

$$Av = -\frac{R_{14} + R_{V1}}{R_{13}} \rightarrow R_{14} + R_{V1} = (Av)R_{13} \rightarrow R_{V1} = (Av)R_{13} - R_{14}$$

Sabiendo que $R_{14}=R_{13}$ y que $Av=11$

$$R_{V1} = (11)R_{14} - R_{14} \rightarrow R_{V1} = 10R_{14}$$

La resistencia R_{V1} debe ser 10 veces más grande que R_{14} , se propone una resistencia $R_{13}=R_{14}$ de $10K\Omega$ y R_{V1} de $100K\Omega$.

Por último, los capacitores C_9 se encarga de acoplar la señal de entrada la etapa anterior.

11.7 Microcontrolador attiny 85

El microcontrolador utilizado en este proyecto es un attiny85 de la marca Atmel, se eligió este microcontrolador ya que cuenta con 6 puertos de entrada/salida y convertidores ADC, además de permitir el ser programado bajo el IDLE de Arduino.

El trabajo del microcontrolador es el que transformar la señal analógica del circuito de acondicionamiento en un valor digital y mandarlo a través del puerto serial al módulo Bluetooth.

Para programar el attiny85 es necesario contar con una placa Arduino externa ya que ésta realizará el trabajo de ISP para programar el microcontrolador.

11.7.1 Voltaje de referencia ADC

Para no perder resolución en la conversión analógico-digital de la señal del corazón se propone modificar el voltaje de referencia del ADC a través del pin AREF del microcontrolador, (para más información ver la sección [8.9 Microcontrolador Attiny 85](#)).

Según los límites establecidos en la sección [8.7 Amplificador operacional LM324](#), el voltaje máximo de trabajo será de 3V, por lo cual se propone dar un margen de error referenciando el ADC a 3.3V.

Para tener un voltaje estable, se utiliza un seguidor de voltaje referenciado a un divisor de voltaje. El cálculo de este divisor se muestra en el siguiente circuito ([Figura 86](#)):

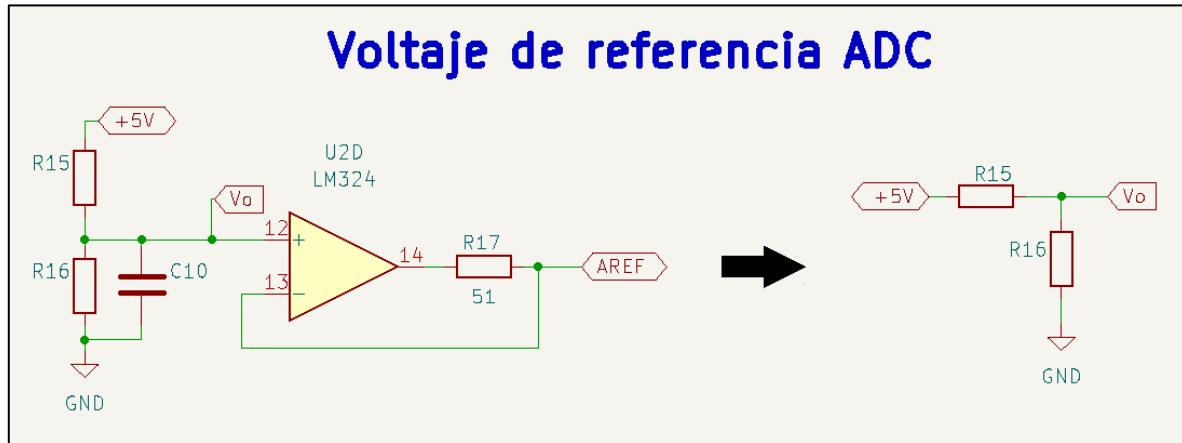


Figura 86.-Divisor de tensión, voltaje de referencia ADC, (módulo B).

Conociendo que:

$$V_o = \frac{R_{16}}{R_{15} + R_{16}} 5V \rightarrow 3.3V = \frac{R_{16}}{R_{15} + R_{16}} 5V \rightarrow 3.3R_{15} + 3.3R_{16} = 5R_{16}$$

$$R_{15} = \frac{1.7}{3.3} R_{16} \rightarrow R_{15} \approx \frac{1}{2} R_{16}$$

El valor de R₁₅ debe ser ≈0.5 veces el valor de R₁₆. Se propone usar un valor de R₁₅=100KΩ y R₁₆=200KΩ.

Se agrega el capacitor C10, el cual filtra el ruido térmico y cualquier tipo de perturbación que pueda tener la fuente de alimentación.

El capacitor C10 y las resistencias R15 y R16 crean un filtro pasa bajas. La frecuencia de corte de este filtro deberá ser inferior a 20Hz para no afectar el desempeño del circuito de audio. Proponiendo un capacitor de 470nF, se tiene:

$$f_c = \frac{1}{2\pi(200K\Omega||100K\Omega)470nf} = 5.07Hz$$

Finalmente, el circuito de voltaje de referencia ADC puede ser visto a continuación ([Figura 80](#))

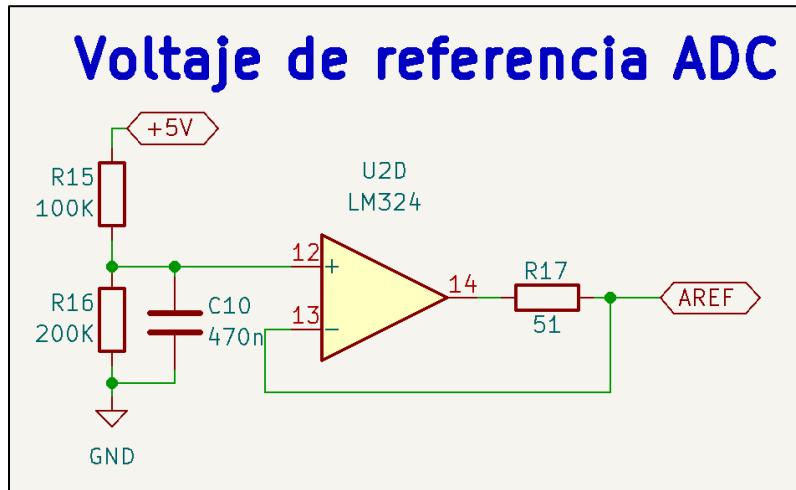


Figura 87.- Voltaje de referencia ADC con valores, (módulo B).

11.8 Conexiones del attiny 85

Debido a que el attiny85 cuenta únicamente con 6 pines de entrada/salida, se tienen dos modos de configuración, el modo programador y el modo de funcionamiento normal. Para elegir entre estos modos, se conectan terminales dobles en cada uno de sus pines, esto con el fin de conectar o desconectar el microcontrolador del circuito a través de jumpers conectados en sus terminales, en la [Figura 88](#) se muestran de color azul las terminales a puentear.

De esta forma, es posible conectar el microcontrolador al circuito de acondicionamiento puenteando sus terminales con jumpers (modo de funcionamiento normal), o dejar las terminales libres para ser programado de manera externa (modo programador).

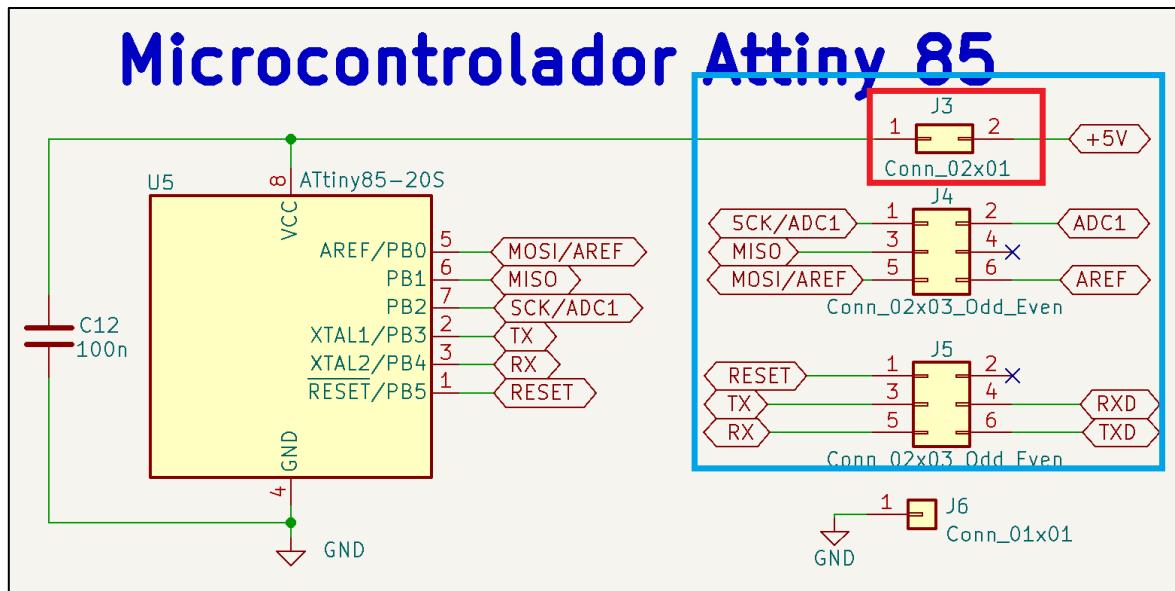


Figura 88.- Terminales de conexión del microcontrolador attiny85, (módulo B).

En la [Figura 88](#), de color rojo se muestra una terminal doble entre la conexión Vcc a +5V del attiny85, esta conexión sirve para encender o apagar el microcontrolador independientemente del circuito de acondicionamiento, esto con fin de probar por separado diferentes partes del circuito al momento de ensamblarlo.

11.8.1 Conexión al programador

Para conectar el attiny85 al programador, es necesario quitar los jumpers del microcontrolador, dejando las terminales al aire. Se conectan de manera externa cables hacia las terminales SCK/AD1, MISO, MOSI/AREF y RESET ([Figura 89](#)). El diagrama de conexiones será discutido en la sección [14 Programación del attiny85](#).

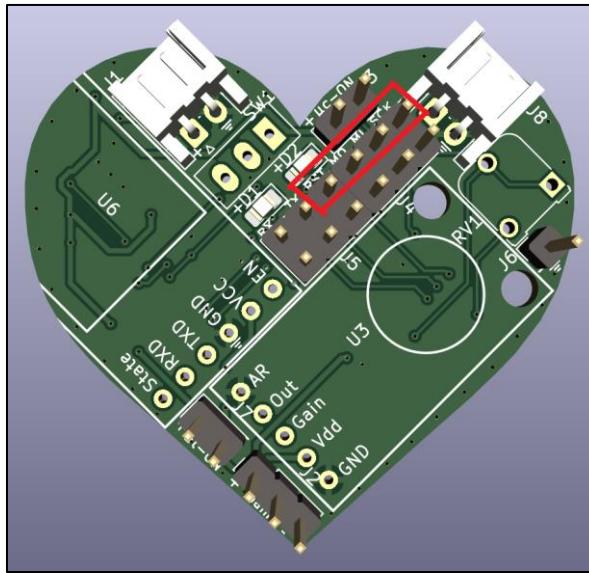


Figura 89.- Terminales de programación, (módulo B).

11.8.2 Conexión al circuito de acondicionamiento

La terminal MOSI/AREF se conecta a la salida del voltaje de referencia ADC visto en la sección [11.7.1 Voltaje de referencia ADC](#) siendo esta terminal en donde se suministra el voltaje de referencia para el convertidor analógico-digital. Por otra parte, la terminal SCK/AD1 se conecta a la salida del circuito de acondicionamiento, siendo el pin entrada de la señal analógica al microcontrolador ([Figura 90](#)).

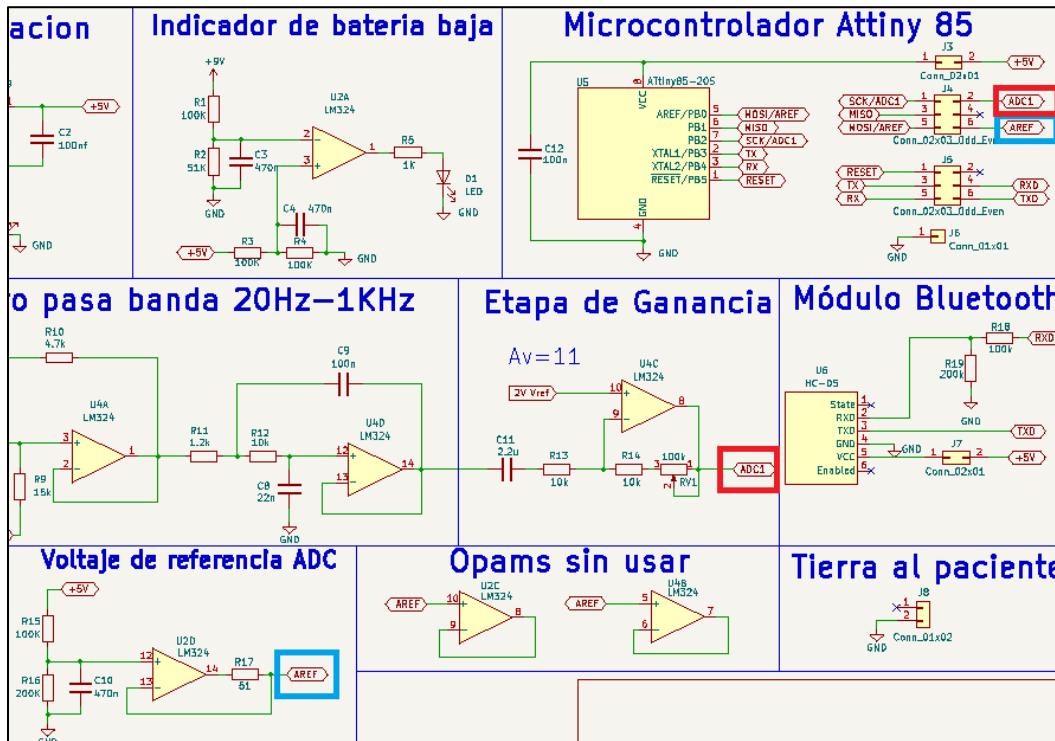


Figura 90.- Conexiones del microcontrolador al circuito de acondicionamiento., (módulo B).

En la [Figura 90](#) se tiene:

- De color rojo la conexión en entre el circuito de acondicionamiento y el microcontrolador.
- De color azul la conexión entre el voltaje de referencia y el microcontrolador.

11.8.3 Conexión al módulo Bluetooth

Algunos módulos Bluetooth poseen una etapa de acoplamiento integrada ya que los voltajes de trabajo típicos para un microcontrolador son de 0V a 5V, mientras que los voltajes de trabajo del módulo Bluetooth van desde 0V hasta 3.3V. La versión de HC-05 usada en este proyecto no posee un acoplamiento de voltajes, por lo cual es necesario implementarla para no dañar el dispositivo.

Según la sección [8.9 Microcontrolador Attiny 85](#), los voltajes de trabajo del attiny85 son:

- Entrada baja -> [-0.5 , 1.5]V
- Entrada alta -> [3.0 , 5.5]V
- Salida baja -> [0.0 , 0.6]V
- Salida alta -> [4.3 , 5.0]V

El voltaje de salida bajo del HC-05 se encuentra dentro del rango de valores del attiny85 para ser reconocido como entrada baja, de igual forma el voltaje de salida alto logra entrar en el rango de valores de entrada alta del microcontrolador, por lo cual no es necesario agregar un acoplamiento entre TXD del HC-05 y RX del attiny85.

El voltaje de entrada bajo del HC-05 se encuentra en el mismo rango de valores que el de salida baja del attiny85, sin embargo, el voltaje de entrada alto del módulo Bluetooth se encuentra por debajo del voltaje de salida alto del microcontrolador, por lo cual es necesario disminuir este voltaje para evitar que el HC-05 pueda dañarse. Para esto se utilizará un divisor de tensión.

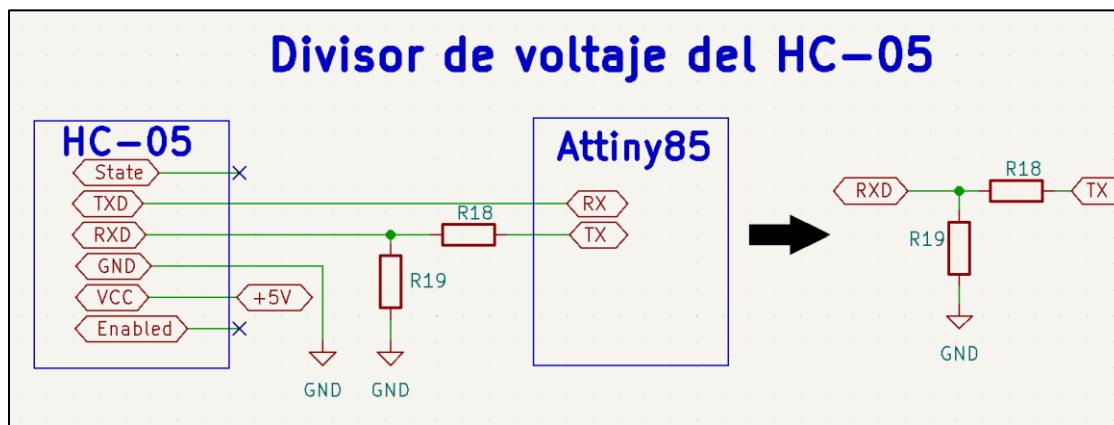


Figura 91.- Divisor de voltaje para el acoplamiento entre el attiny85 y el HC-05, (módulo B).

Se sabe que el pin TX del attiny 85 tendrá un voltaje de 5V y que el voltaje de trabajo del HC-05 en el pin RXD es de 3.3V. Con base en el divisor de la [Figura 91](#), se tiene:

$$R_{XD} = \frac{R_{19}}{R_{18} + R_{19}} T_X \rightarrow 3.3V = \frac{R_{19}}{R_{18} + R_{19}} 5V \rightarrow 3.3R_{18} + 3.3R_{19} = 5R_{19}$$

$$R_{18} = \frac{1.7}{3.3} R_{19} \rightarrow R_{18} \approx \frac{1}{2} R_{19}$$

El valor de R18 debe ser ≈ 0.5 veces el valor de R19. Se propone R18=100K Ω y R19=200K Ω .

Finalmente, el acoplamiento del módulo Bluetooth puede ser visto a continuación ([Figura 92](#)).

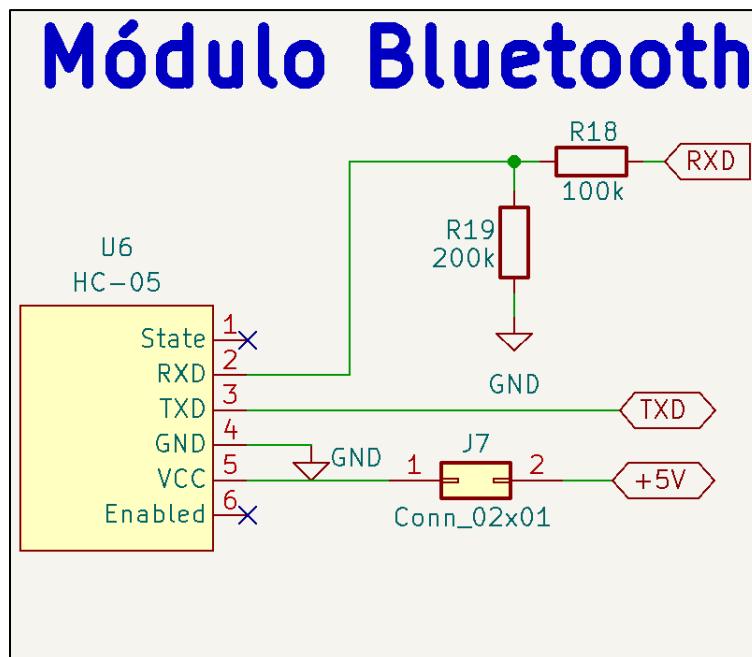


Figura 92.- Conexiones de acoplamiento del módulo Bluetooth, (módulo B).

Se agrega una terminal doble entre la conexión Vcc a +5V del HC-05 para apagar o encender el módulo independientemente del circuito de acondicionamiento, esto con fines de pruebas al ensamblar el circuito impreso.

Las conexiones para configurar el módulo Bluetooth por separado, así como los comandos AT para programarlo serán discutidos en la sección [15 Programación del módulo Bluetooth HC-05 / EGBT-045MS](#).

11.9 Conexión de tierra al paciente

Para eliminar cualquier tipo de perturbación que pueda ser inducida por el paciente, se propone conectar un cable hacia GND, logrando referenciar la señal del circuito.

Según la hoja de datos del amplificador de instrumentación AD620 [12], para un circuito de aplicación biomédica ECG, se recomienda conectar una tierra virtual y una etapa de protección entre el circuito y los electrodos del paciente (*Figura 93*).

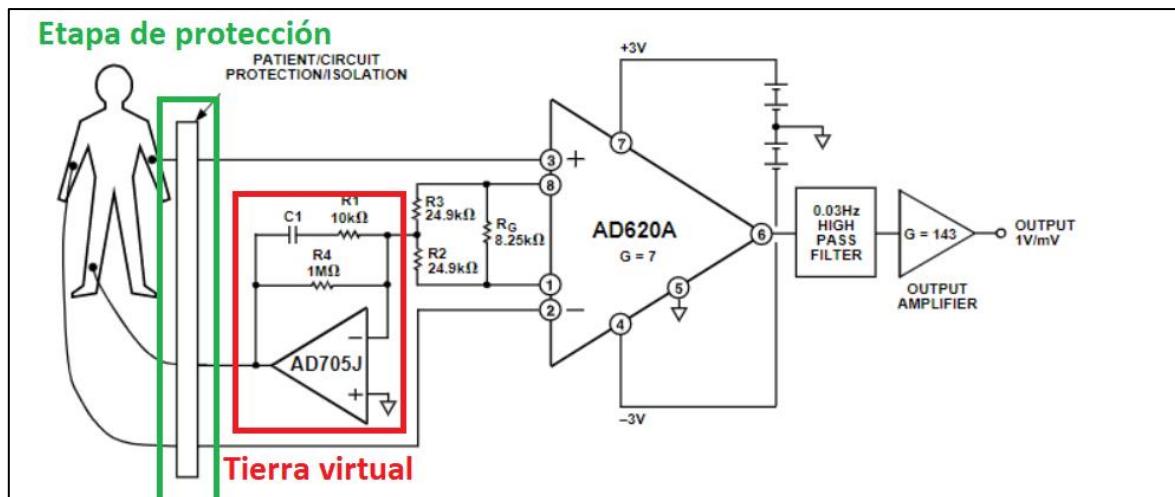


Figura 93.- Circuito de monitor ECG, (módulo B).

Debido a las limitaciones de los amplificadores LM324 utilizados en el circuito de acondicionamiento, no es posible conseguir una salida hacia GND con una alimentación unipolar. Para generar una tierra virtual sería necesario integrar una nueva etapa al circuito que permita alimentaciones bipolares o modificar los valores previamente calculados para agregar esta nueva referencia. Dadas las complicaciones no se integrará la etapa de tierra virtual, sin embargo, la etapa de protección entre GND y el paciente puede ser generada.

Se propone utilizar una pulsera antiestática conectada desde GND hasta alguna extremidad del paciente. La pulsera cuenta con una impedancia de aproximadamente $1M\Omega$, permitiendo una protección de corto circuito y a su vez, referenciar el circuito con el paciente, eliminando cualquier tipo de ruido que este pueda generar.

Para conectar la pulsera se hace uso de un conector JST XH2.54 el cual está conectado directamente a tierra (*Figura 94*).

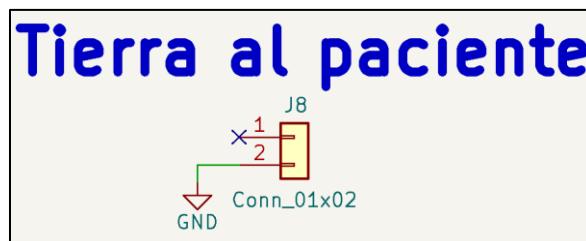


Figura 94.- Tierra al paciente. (módulo B).

12 Módulo C: Control de ganancia automático

MAX9814 y módulo Bluetooth EGBT-045MS

El módulo C ([Figura 95](#), [Figura 96](#) y [Figura 97](#)) cuenta con las siguientes secciones:

- 12.1 Fuente de alimentación
 - 12.1.1 Regulador de voltaje LP2985-33
 - 12.1.2 Regulador de voltaje LP2985-5
 - 12.1.3 Interruptor, diodo de protección e indicador led de encendido
- 12.2 Indicador de batería baja
 - 12.2.1 Voltaje en la terminal negativa (Divisor de voltaje a 9V)
 - 12.2.2 Voltaje en la terminal positiva (Divisor de voltaje 5V)
- 12.3 Pre-amplificador del micrófono MAX9814
 - 12.3.1 Microphone Bias Resistor
 - 12.3.2 Bias Capacitor
 - 12.3.3 Attack Time and Release Time
 - 12.3.4 Ajuste de offset
 - 12.3.5 Power-Supply Bypassing
 - 12.3.6 Threshold Voltaje
 - 12.3.7 Capacitor de acople a la salida
 - 12.3.8 Ganancia del circuito
 - 12.3.9 Modificaciones físicas al módulo de Adafruit
- 12.4 Voltaje de referencia offset
- 12.5 Filtro pasa banda 20Hz-1KHz
 - 12.5.1 Filtro pasa altas 20Hz
 - 12.5.2 Filtro pasa bajas 1KHz
- 12.6 Etapa de ganancia
- 12.7 Módulo Bluetooth EGBT-045MS
 - 12.7.1 Pin de transmisión de datos TXD
 - 12.7.2 Pin de recepción de datos RXD
 - 12.7.3 Pin indicador led de estatus
 - 12.7.4 Pulsador de modos AT
 - 12.7.5 Conexiones del módulo EGBT-045MS
- 12.8 Microcontrolador attiny 85
 - 12.8.1 Voltaje de referencia ADC
- 12.9 Conexiones del attiny 85
 - 12.9.1 Conexión al programador
 - 12.9.2 Conexión al circuito de acondicionamiento
 - 12.9.3 Conexión al módulo Bluetooth
- 12.10 Conexión de tierra al paciente

Nota: Para la programación de la placa, ver la sección: [13 Arduino como ISP](#), [14 Programación del attiny85](#) y [15 Programación del módulo Bluetooth HC-05 / EGBT-045MS](#)

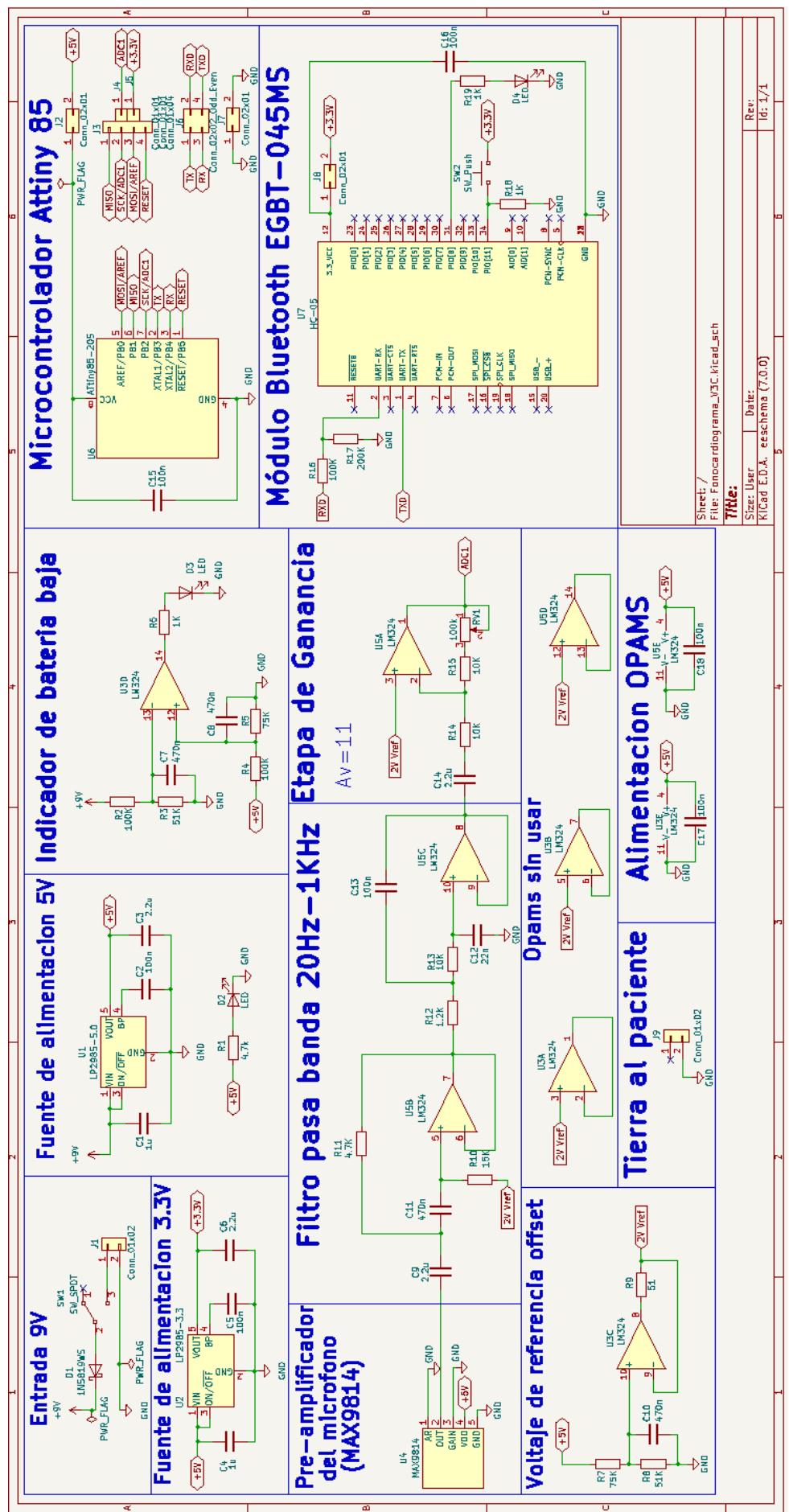


Figura 95.- Diagrama eléctrico, (módulo C).

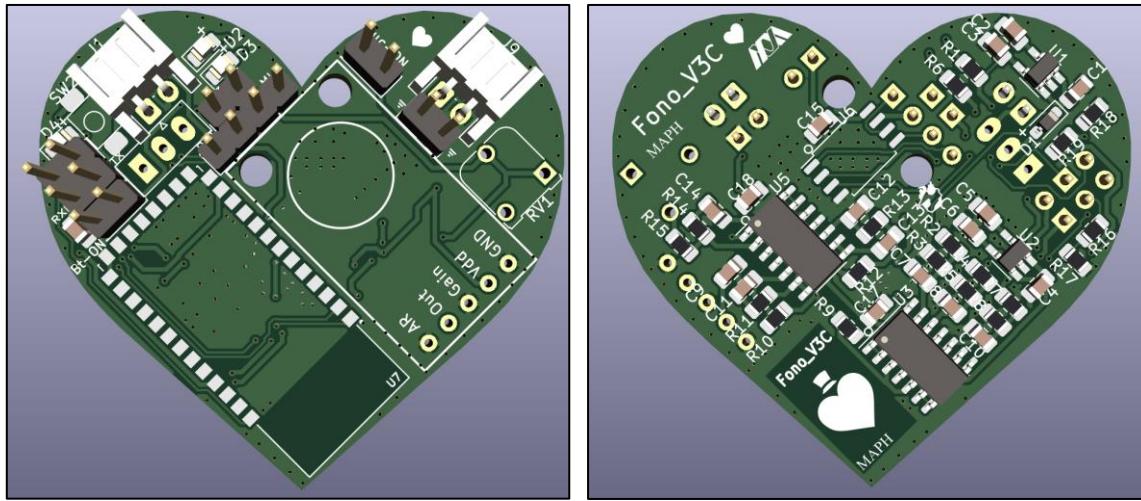


Figura 96.- Izquierda: Modelo 3D de la placa, vista frontal (módulo C).
Derecha: Modelo 3D de la placa, vista trasera (módulo C).

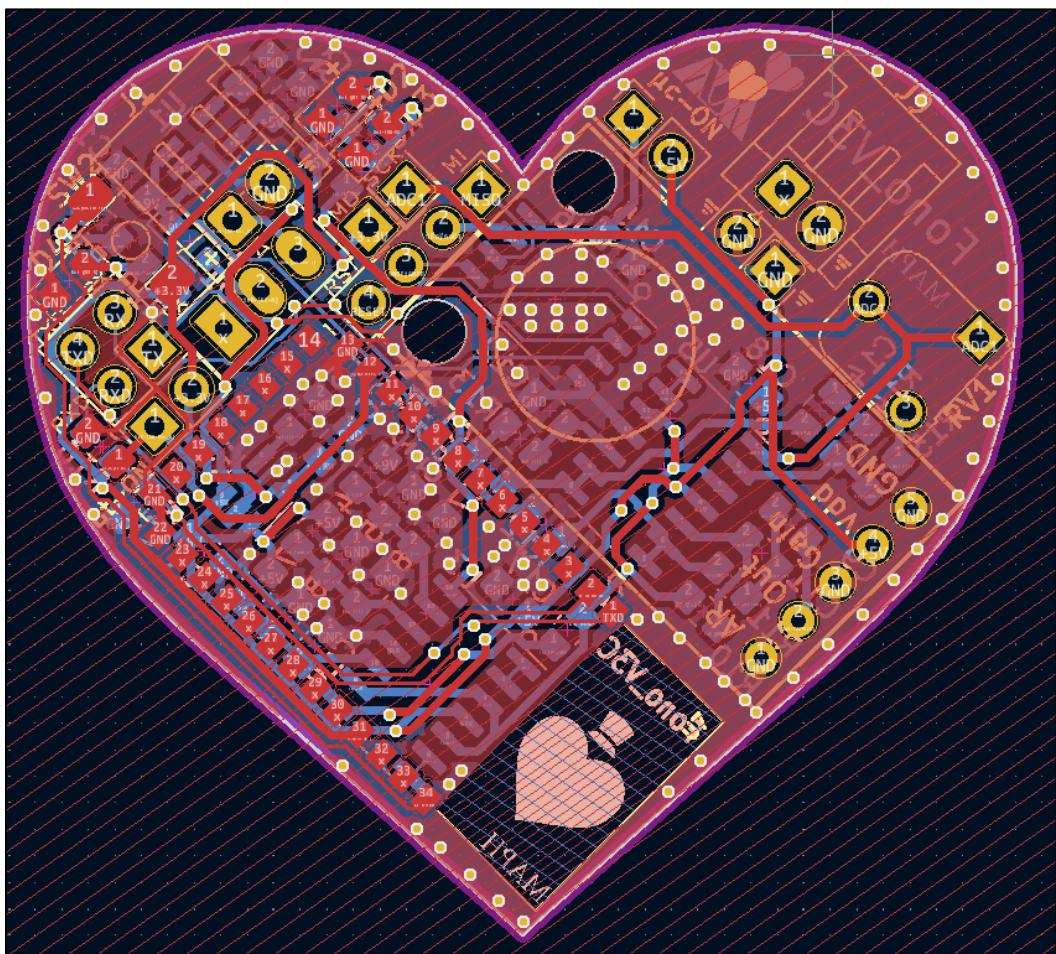


Figura 97.- Diagrama del PCB, (módulo C).

12.1 Fuente de alimentación

12.1.1 Regulador de voltaje LP2985-33

El módulo Bluetooth usado en este modelo de placa (EGBT-045MS), tiene un voltaje de trabajo de 3.3V, por lo cual se hace uso de un LP2985-33 para alimentarlo ([Figura 98](#)).

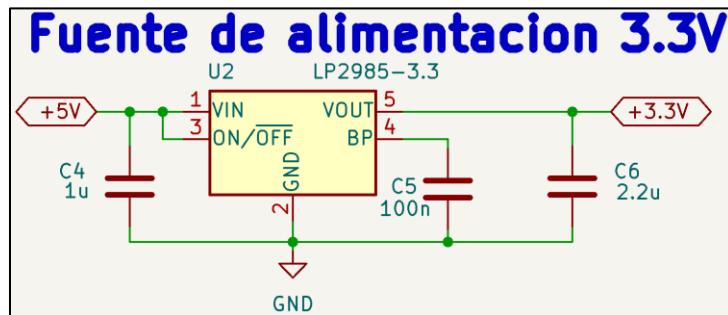


Figura 98.- Diagrama de conexiones LP2985-33, (módulo C).

Se eligió un regulador de la familia LP2985 ya que el consumo energético del Bluetooth es de aproximadamente 40-50mA siendo inferior a la corriente máxima suministrada por el regulador (250mA). De igual forma al ser un regulador tipo LDO, permite tener una alimentación de 5V la cual será suministrada por el regulador visto en la sección [8.2 Regulador de voltaje positivo LP2985](#).

El circuito consta de un regulador con tres capacitores. Parte del circuito fue obtenido de su hoja de datos ([Figura 99](#)). Estos capacitores se encargan de regular cualquier pico de voltaje que pueda presentarse tanto en la entrada como en la salida del regulador.

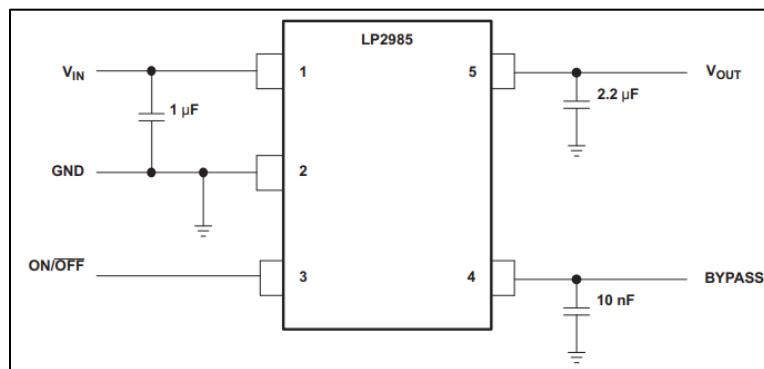


Figura 99.- Circuito de prueba, L2985 [11], (módulo C).

12.1.2 Regulador de voltaje LP2985-5

Tanto del circuito de acondicionamiento como el microcontrolador y la alimentación del regulador para el módulo Bluetooth, se encuentran diseñados para trabajar con una alimentación de 5V, por lo cual se hace uso de un LP2985-5 y una batería de 9V para alimentar el circuito ([Figura 100](#)).

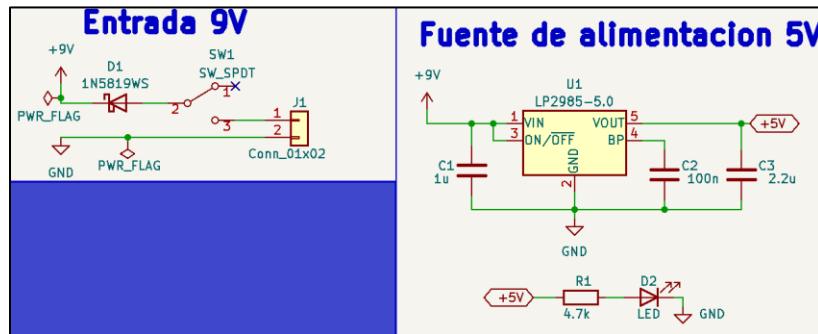


Figura 100.- Diagrama de conexiones LP2985-5, (módulo C).

Se eligió un L2985-5 ya que el consumo máximo visto en las pruebas de desempeño es menor a 250mA, siendo ésta la corriente máxima entregada por el regulador. De igual forma, al ser un regulador tipo LDO, se logra una mayor vida útil de la batería de 9V comparado con el L78L05, el cual tiene una mayor caída de voltaje.

El circuito consta de un LP2985-5 y tres capacitores, estos capacitores se encargan de regular cualquier tipo de perturbación que se pueda tener en la entrada o en la salida del regulador. Parte del circuito fue obtenido de su hoja de datos ([Figura 99](#)).

12.1.3 Interruptor, diodo de protección e indicador led de encendido

Se agrega un interruptor entre la batería de 9V y la entrada del regulador, el cual permite encender o apagar el circuito sin necesidad de desconectarlo. De igual forma, se agrega un diodo de bajo consumo de voltaje ([8.8 Diodo Schottky 1N5819](#)) como protección al circuito en caso de conectar al revés la batería. Por último, se agrega un led a la salida del regulador de 5V con el fin de ser un indicador cuando el circuito se encuentre encendido ([Figura 101](#)).

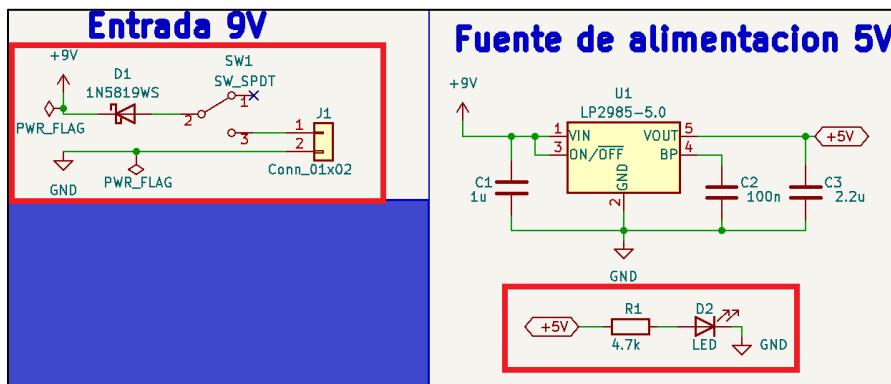


Figura 101.- Interruptor e indicador luminoso de encendido, (módulo C).

12.2 Indicador de batería baja

A medida que se desgaste la batería de 9V, el voltaje comenzará a disminuir, para evitar un comportamiento errático debido a este desgaste, se hace uso de un indicador de batería baja.

El indicador consta de un LM324 utilizado como comparador, dos divisores de voltaje y un led conectado en la salida del comparador ([Figura 102](#)).

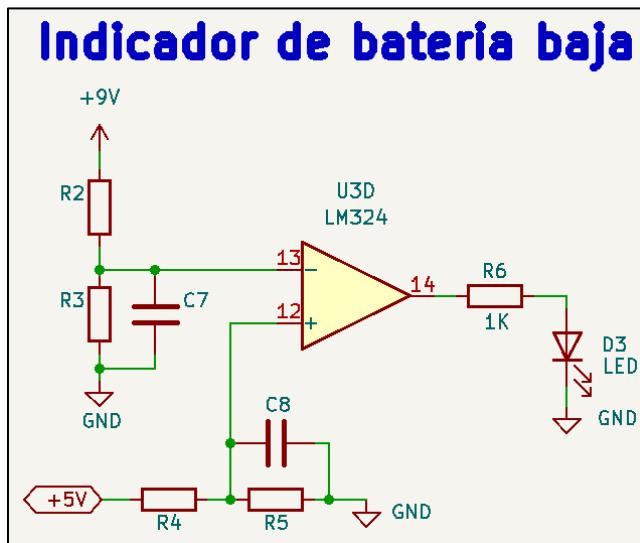


Figura 102.- Circuito indicador de batería baja, (módulo C).

El comparador es alimentado por el voltaje regulado del LP2985-5, es decir, tiene una alimentación unipolar de 5V. Según lo propuesto es la sección [8.7 Amplificador operacional LM324](#), los voltajes de entrada al comparador se encuentran en el rango de [1V,3V].

Según lo propuesto en la sección [8.2 Regulador de voltaje positivo LP2985](#), la caída de voltaje recomendada en el regulador de 1V por encima del voltaje de salida, para tener una mayor margen de error, se propone tener un voltaje de 1.5V, es decir, el voltaje mínimo visto en la batería de 9V, sería de 6.5V.

Dadas las especificaciones anteriores, mientras el voltaje de la batería se encuentre en el rango de [6.5V,9V], la salida del comparador estará referenciada al voltaje de alimentación negativo, cuando el voltaje de la batería decaea por debajo de 6.5V, la salida del comparador cambiara hacia la alimentación positiva, teniendo un voltaje de 3.5 en su salida, encendiendo el indicador led.

12.2.1 Voltaje en la terminal negativa (Divisor de voltaje a 9V)

El primer divisor de voltaje se encuentra referenciado a la batería de 9V, este divisor debe reducir el voltaje de tal forma que se tenga un valor en la entrada del comparador en el rango de [1V,3V]. De igual forma, cuando el voltaje de la batería se encuentre alrededor de 6.5V, el voltaje en el divisor debe estar contenido en este rango de valores ([Figura 103](#)).

Indicador de batería baja, terminal negativa

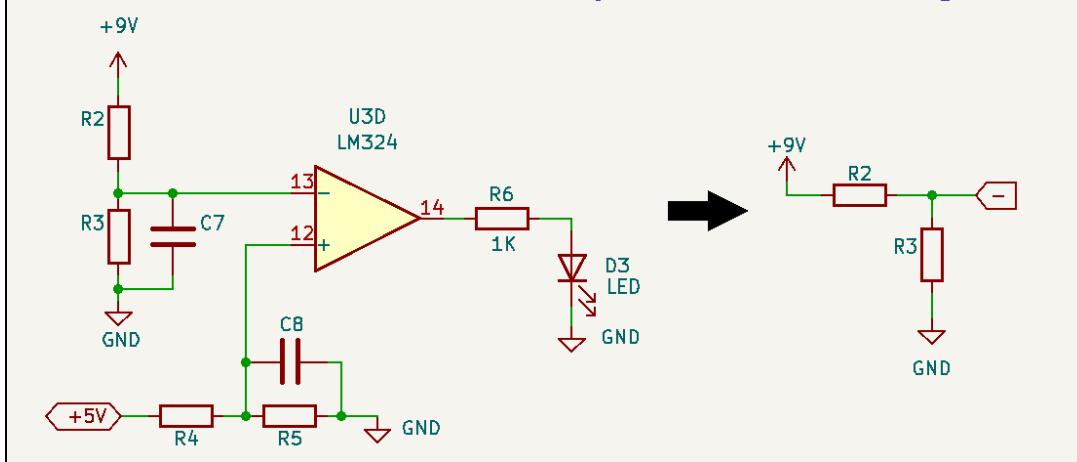


Figura 103.- Divisor de voltaje de 9V, terminal (-) del comparador, (módulo C).

Se propone tener un voltaje de 3V en el divisor cuando la batería de 9V se encuentra en su carga máxima, teniendo que:

$$V_{out} = \frac{R_3}{R_2 + R_3} V_{in} \rightarrow 3V = \frac{R_3}{R_2 + R_3} 9V \rightarrow 3R_2 + 3R_3 = 9R_3 \rightarrow R_2 = \frac{6}{2} R_3 \rightarrow R_2 = 2R_3$$

El valor de R2 debe ser 2 veces el valor de R3. Se propone usar un valor de R1=100KΩ y R2=51KΩ

Teniendo estos valores, se hace el cálculo cuando el voltaje en la batería es de 6.5V, este dato será usado para calcular el segundo divisor de voltaje.

$$V_{out} = \frac{R_3}{R_2 + R_3} V_{in} \rightarrow V_{out} = \frac{R_3}{R_2 + R_3} 6.5V \rightarrow V_{out} = \frac{51K\Omega}{100K\Omega + 51K\Omega} 6.5V = \frac{51K\Omega}{151K\Omega} 6.5V$$

$$V_{out} = 2.20V$$

Con los voltajes calculados se sabe que:

- Voltaje en la batería = 9V -> Voltaje en la terminal negativa = 3V
- Voltaje en la batería = 6.5V -> Voltaje en la terminal negativa = 2.2V

Por último, se agrega el capacitor C7, el cual filtra el ruido térmico y cualquier tipo de perturbación que pueda tener la fuente de alimentación.

El capacitor C7 y las resistencias R2 y R3 crean un filtro pasa bajas. La frecuencia de corte de este filtro deberá ser inferior a 20Hz, esto con el fin de no afectar el desempeño del circuito de audio. Proponiendo un capacitor de 470nF, se tiene:

$$f_c = \frac{1}{2\pi(100K\Omega||51K\Omega)470nf} = 10.02Hz$$

12.2.2 Voltaje en la terminal positiva (Divisor de voltaje 5V)

Para obtener un cambio en la polaridad de la salida del comparador es necesario introducir un voltaje de referencia similar al calculado en la sección [12.2.1 Voltaje en la terminal negativa \(Divisor de voltaje a 9V\)](#) cuando el voltaje en la batería de 9V decae a su valor mínimo permitido de 6.5V. Siendo el voltaje calculado en la sección anterior de 2.2V, este será el valor de voltaje a introducir en la terminal positiva del comparador, de tal forma que, cuando el voltaje en la batería disminuya, el voltaje en la terminal negativa del comparador también disminuye, pasando por debajo del voltaje de referencia de 2.2V, al tener un voltaje mayor en la terminal positiva que en la terminal negativa, el voltaje en la salida del comparador cambia a ser referenciado hacia el voltaje de alimentación, encendiendo el led ([Figura 104](#)).

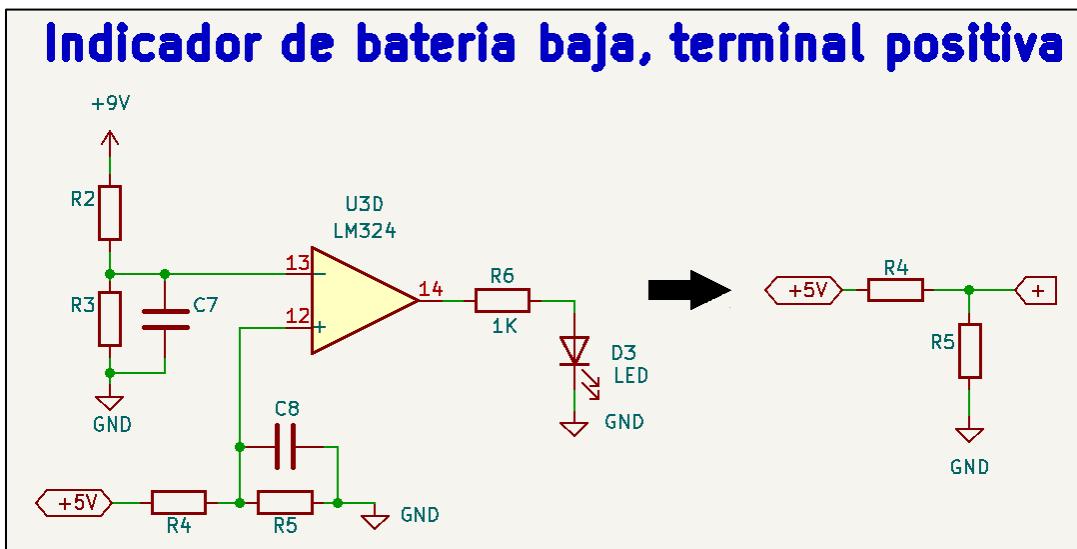


Figura 104.- Divisor de voltaje de 5V, terminal (+) del comparador, (módulo C).

Este segundo divisor necesita un voltaje fijo de operación, el cual será tomado de la salida del regulador LP2985-5:

$$V_{out} = \frac{R_5}{R_4 + R_5} V_{in} \rightarrow 2.2V = \frac{R_5}{R_4 + R_5} 5V \rightarrow 2.2R_4 + 2.2R_5 = 5R_5 \rightarrow R_4 = \frac{2.8}{2.2} R_5 \rightarrow R_4 = 1.27R_5$$

La resistencia R4 debe ser 1.27 veces mayor que R5, se propone R4=100 KΩ Y R5=75 KΩ

De igual forma que en el divisor de 9V, se agrega un capacitor C8 para filtrar el ruido térmico y cualquier tipo de perturbación que pueda tener la fuente de alimentación.

El capacitor C8 y las resistencias R4 y R5 crean un filtro pasa bajas. La frecuencia de corte de este filtro deberá ser inferior a 20Hz, esto con el fin de no afectar el desempeño del circuito de audio. Proponiendo un capacitor de 470nF, se tiene:

$$f_c = \frac{1}{2\pi(100K\Omega||75K\Omega)470nf} = 7.9Hz$$

Finalmente, el circuito queda de la siguiente forma:

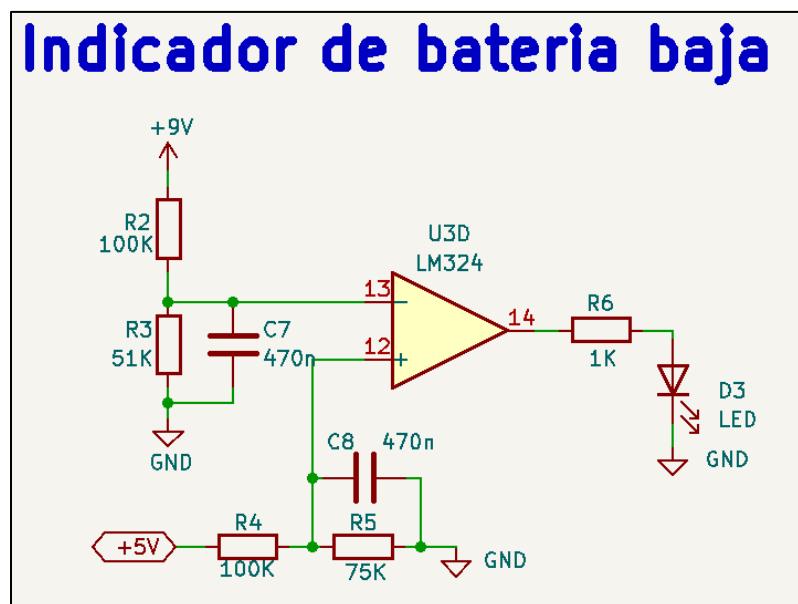


Figura 105.- Circuito indicador de batería baja (con valores), (módulo C).

12.3 Pre-amplificador del micrófono MAX9814

Según la sección [8.6 Pre-amplificador de micrófono MAX9814](#), el tipo de empaquetado para el MAX9814 es el TDFN, debido a la complejidad de integrar este componente a la placa, se decidió hacer uso de la placa de Adafruit ([8.5 Módulo MAX9814 de Adafruit](#)) la cual ya cuenta con el MAX9814 soldado. Para ajustar el comportamiento del módulo a las necesidades de la placa, se desoldarán ciertos componentes que trae el módulo de Adafruit.

12.3.1 Microphone Bias Resistor

Según la hoja de datos del MAX9814 [15], se recomienda usar una resistencia de 2.2KΩ para la polarización del micrófono, siendo común en aplicaciones de micrófonos Electret. ([Figura 106](#)).

Microphone Amplifier with AGC and Low-Noise Microphone Bias

Setting the AGC Threshold

To set the output-voltage threshold at which the microphone output is clamped, an external resistor-divider must be connected from MICBIAS to ground with the output of the resistor-divider applied to TH. The voltage V_{TH} determines the peak output-voltage threshold at which the output becomes clamped. The maximum signal swing at the output is then limited to two times V_{TH} and remains at that level until the amplitude of the input signal is reduced. To disable AGC, connect TH to MICBIAS.

Microphone Bias Resistor

MICBIAS is capable of sourcing 20mA. Select a value for RMICBIAS that provides the desired bias current for the electret microphone. A value of 2.2kΩ is usually sufficient for a microphone of typical sensitivity. Consult the microphone data sheet for the recommended bias resistor.

Choose C_{IN} such that f_{-3dB_IN} is well below the lowest frequency of interest. Setting f_{-3dB_IN} too high affects the amplifier's low-frequency response. Use capacitors with low-voltage coefficient dielectrics. Aluminum electrolytic, tantalum, or film dielectric capacitors are good choices for AC-coupling capacitors. Capacitors with high-voltage coefficients, such as ceramics (non-C0G dielectrics), can result in increased distortion at low frequencies.

Output Capacitor

The output of the MAX9814 is biased at 1.23V. To eliminate the DC offset, an AC-coupling capacitor (C_{OUT}) must be used. Depending on the input resistance (R_L) of the following stage, C_{OUT} and R_L effectively form a highpass filter. The -3dB point of the highpass filter, assuming zero output impedance, is given by:

MAX9814

Figura 106.- Microphone Bias Voltaje [15], (módulo C).

Para extraer las señales eléctricas producidas por el micrófono, se utiliza un capacitor de acople. Según el diagrama proporcionado por Adafruit ([Figura 12](#)), la resistencia utilizada en la polarización del micrófono es de 2.2KΩ, esta resistencia junto con el capacitor de acople forma un filtro pasa altas.

La señal de interés del corazón se encuentra en un rango de frecuencias de 20Hz a 1KHz ([Figura 4](#)), por lo cual, se propone una frecuencia de corte de 15Hz, teniendo la siguiente ecuación:

$$C = \frac{1}{2\pi R f_c} = \frac{1}{2\pi(2.2K\Omega)(15Hz)} = 4.8\mu F \rightarrow 4.4\mu F$$

Dicho esto, se modificará el capacitor de acople del módulo de Adafruit, pasando de un valor de 0.1uF a dos capacitores en paralelo de 2.2uF.

12.3.2 Bias Capacitor

Según la hoja de datos del MAX9814 [15], se recomienda utilizar un capacitor de polarización de 470nF conectado a tierra ([Figura 107](#)), el modelo visto en la [Figura 12](#) de Adafruit ya cuenta con este capacitor, por lo cual no se modificará.

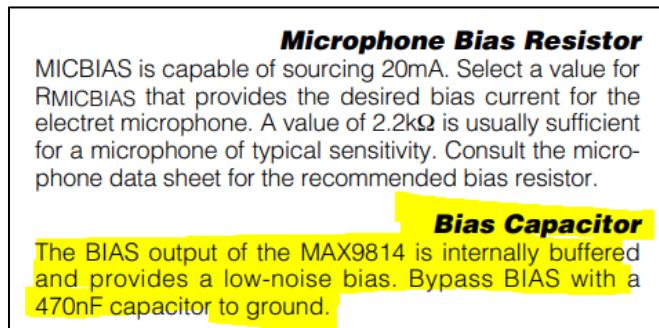


Figura 107.- Bias Capacitor [15], (módulo C).

12.3.3 Attack Time and Release Time

Para ajustar los tiempos de activación (Attack Time) y el tiempo que le toma al control de ganancia automático de regresar a su valor normal de operación (Release Time), se utiliza el capacitor CT y la terminal A/R.

En este caso, se requiere que tanto el “Attack Time” y el “Release Time” sean lo más rápido posible permitiendo disminuir las señales generadas por golpes accidentales.

Según la hoja de datos del MAX9814 [15], se propone utilizar un capacitor de 22nF y conectar el pin A/R a GND, teniendo un “Attack Time” de 0.05ms con un “Release” Time de 25ms ([Figura 108](#)).

The attack and release times can be selected by utilizing the corresponding capacitances listed in Table 2.				
CCT	tATTACK (ms)	tRELEASE (ms)		
		A/R = GND	A/R = V _{DD}	A/R = UNCONNECTED
22nF	0.05	25	100	200
47nF	0.11	55	220	440
68nF	0.16	80	320	640
100nF	0.24	120	480	960
220nF	0.53	265	1060	2120
470nF	1.1	550	2200	4400
680nF	1.63	815	3260	6520
1μF	2.4	1200	4800	9600

Figura 108.- Attack Time y Release Time [15], (módulo C).

El circuito proporcionado por Adafruit ([Figura 12](#)), cuenta con un capacitor CT de 100nF y con el pin A/R sin conectar, en este caso, se tiene que cambiar el capacitor y conectar el pin a GND.

12.3.4 Ajuste de offset

Según la hoja de datos [15], es necesario conectar un capacitor de 2.2uF entre la terminal CG y GND, esto con el fin de garantizar un offset de cero en la salida. ([Figura 109](#)).

Pin Description			
PIN		NAME	FUNCTION
TDFN	UCSP		
1	A1	CT	Timing Capacitor Connection. Connect a capacitor to CT to control the Attack and Release times of the AGC.
2	B2	SHDN	Active-Low Shutdown Control
3	A2	CG	Amplifier DC Offset Adjust. Connect a 2.2 μ F capacitor to GND to ensure zero offset at the output.
4, 11	—	N.C.	No Connection. Connect to GND.

Figura 109.- Offset Adjust [15], (módulo C).

El circuito proporcionado por Adafruit ([Figura 12](#)), ya cuenta con este capacitor, por lo que no es necesario realizar alguna modificación.

12.3.5 Power-Supply Bypassing

Según la hoja de datos del MAX9814 [15], se recomienda conectar un capacitor de bypass de 0.1uF entre la fuente de alimentación y GND ([Figura 110](#)). El circuito de Adafruit ([Figura 12](#)), cuenta con un capacitor de 2.2uF, por lo cual no se realizará alguna modificación.

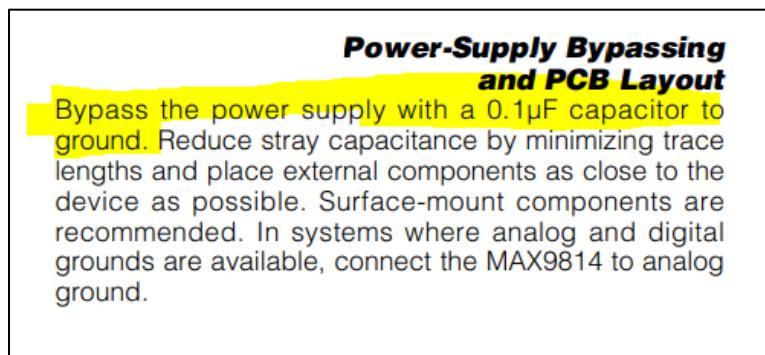


Figura 110.- Power-Supply Bypassing [15], (módulo C).

12.3.6 Threshold Voltaje

Dentro del módulo, existe la terminal “TH” a la cual se le suministra un voltaje constante, este voltaje es comparado internamente con respecto al voltaje de entrada. En caso de que el voltaje de entrada sea inferior a este voltaje de umbral, la señal será amplificada, caso contrario, la señal es atenuada ([Figura 111](#)).

Setting the AGC Threshold

To set the output-voltage threshold at which the microphone output is clamped, an external resistor-divider must be connected from MICBIAS to ground with the output of the resistor-divider applied to TH. The voltage V_{TH} determines the peak output-voltage threshold at which the output becomes clamped. The maximum signal swing at the output is then limited to two times V_{TH} and remains at that level until the amplitude of the input signal is reduced. To disable AGC, connect TH to MICBIAS.

Figura 111.- AGC Threshold Voltaje [15], (módulo C).

Para conocer este voltaje de umbral, es necesario probar el circuito de polarización del micrófono por sí solo. Se dessoldó el micrófono integrado en el módulo de Adafruit y se conectó el micrófono utilizado en el fonocardiograma (CMC-60158-42P). Una vez hecho esto, se midió la amplitud de la señal cardiaca antes de entrar al MAX9814.

Como puede observarse en la [Figura 112](#), la amplitud de la señal cardiaca es de 50mVp, se tomará este voltaje como referencia para la señal V_{TH} .

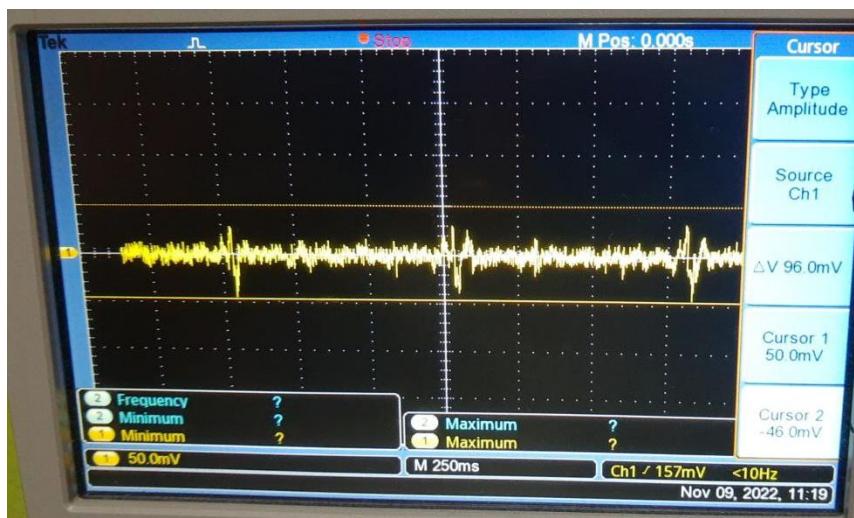


Figura 112.- Amplitud de la señal cardiaca antes del MAX9814, (módulo C).

Para tener un margen de seguridad respecto a la señal cardiaca, se propone un factor de seguridad de 2 unidades, es decir:

$$V_{th} = 2 * V_{cardiaco} = 100\text{mVp}$$

Una vez teniendo el valor de voltaje, se calcula el valor de las resistencias tomando como referencia el circuito de la [Figura 113](#).

$$V_{TH} = \frac{R_{TH2}}{R_{TH1} + R_{TH2}} V_{MICBIAS} \rightarrow 100\text{mVp} = \frac{R_{TH2}}{R_{TH1} + R_{TH2}} 2\text{V} \rightarrow 0.05(R_{TH1} + R_{TH2}) = R_{TH2}$$

$$0.05R_{TH1} = 0.95R_{TH2} \rightarrow R_{TH1} = 19R_{TH2}$$

El valor de RTH1 debe ser 19 veces mayor a RTH2, se propone RTH1 de 100K Ω y RTH2 de 5.1K Ω

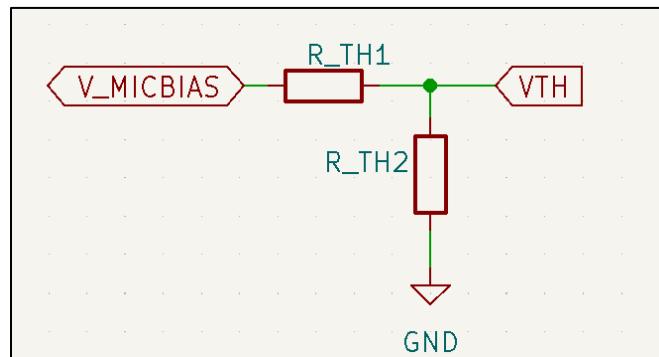


Figura 113.- Divisor de voltaje para VTH, (módulo C).

El circuito proporcionado por Adafruit ([Figura 12](#)) cuenta con una resistencia RTH1=150K Ω , RTH2=100k Ω , por lo cual, es necesario cambiar ambas resistencias.

12.3.7 Capacitor de acople a la salida

Por último, se propone usar un capacitor de 2.2uF en la salida del integrador para acoplar la señal sonora a las demás etapas del circuito. Este capacitor es externo al módulo de Adafruit.

12.3.8 Ganancia del circuito

La ganancia del circuito puede ser modificada conectando la terminal “Gain” a VDD, GND o dejándola desconectada.

VDD-> 40dB

GND->50dB

Desconectada->60dB

Realizando una prueba de funcionamiento sobre el circuito ([Figura 114](#)), al conectar la terminal de ganancia a GND, se tiene una señal con una variación de 1.56V, la cual se encuentra dentro de las cotas establecidas en la sección [8.7 Amplificador operacional LM324](#).



Figura 114.- Amplitud de la señal cardiaca, circuito de Adafruit modificado, (módulo C).

Dado esto, se conectará directamente la terminal hacia GND dentro de la placa del módulo C.

12.3.9 Modificaciones físicas al módulo de Adafruit

El diagrama modificado del módulo MAX9814 se muestra a continuación ([Figura 115](#)).

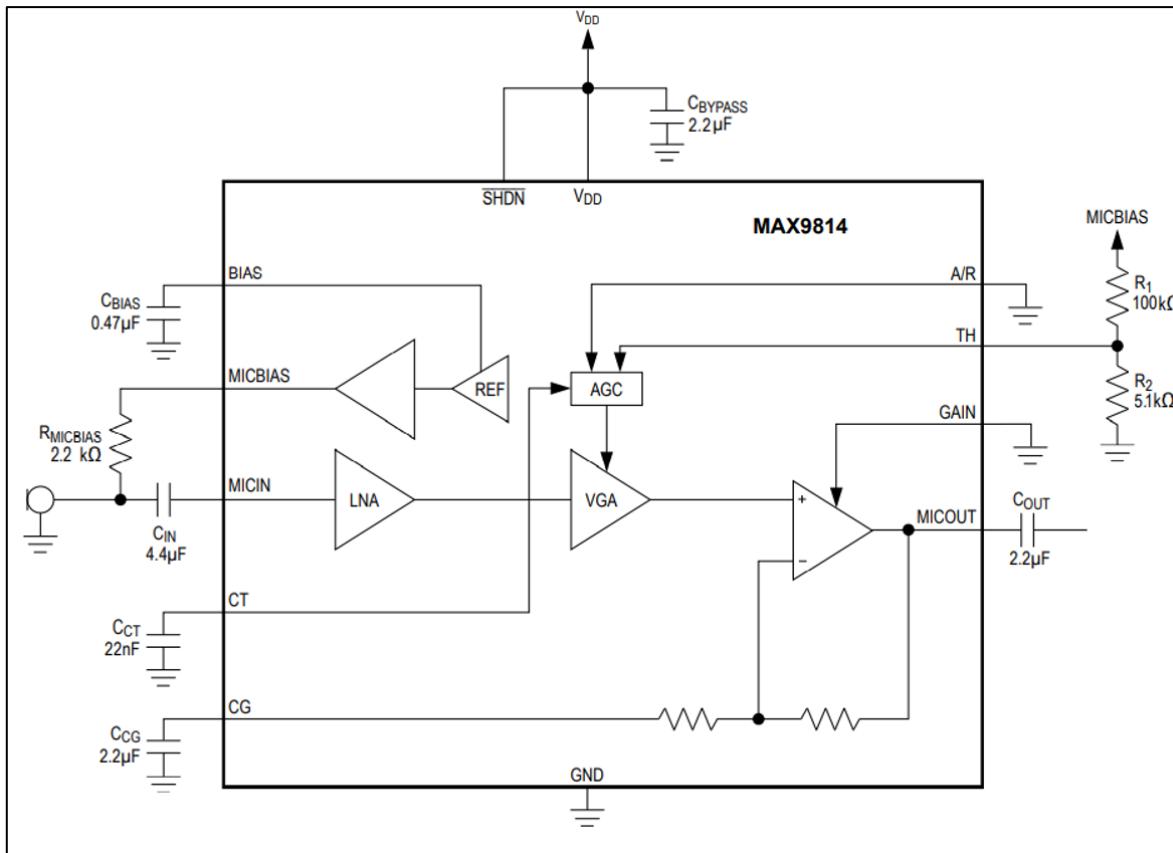


Figura 115.- Diagrama modificado del MAX9814, (módulo C).

En la [Figura 116](#), se encuentran marcados los componentes a modificar dentro del módulo de Adafruit, los cuales son:

- Resistencia 150K: La resistencia RTH1 en el divisor de voltaje visto en la terminal VTH.
- Resistencia 100K: La resistencia RTH2 en el divisor de voltaje visto en la terminal VTH.
- Capacitor 100nF: Capacitor CT para ajustar los tiempos de “Attack Time” y “Release Time”.
- Capacitor 0.1uF: Capacitor de acople en la entrada del micrófono.

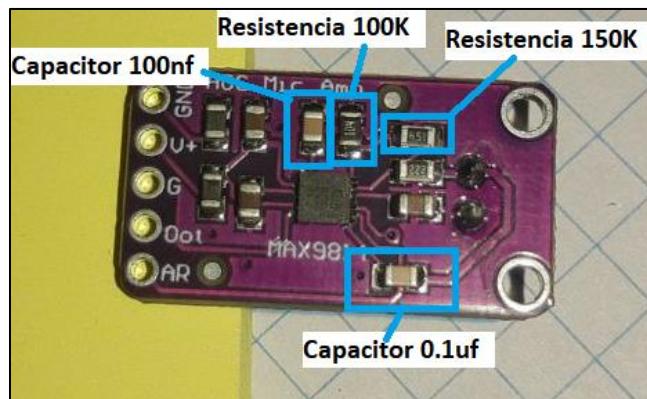


Figura 116.- Componentes a modificar en la placa de Adafruit MAX9814, (módulo C).

En la [Figura 117](#), se encuentran marcados los componentes modificados del módulo de Adafruit, los cuales son:

- Resistencia 100K: La resistencia RTH1
- Resistencia 5.1K: La resistencia RTH2
- Capacitor 22nF: Capacitor CT
- Capacitor 4.4uF: Capacitor de acople del micrófono.

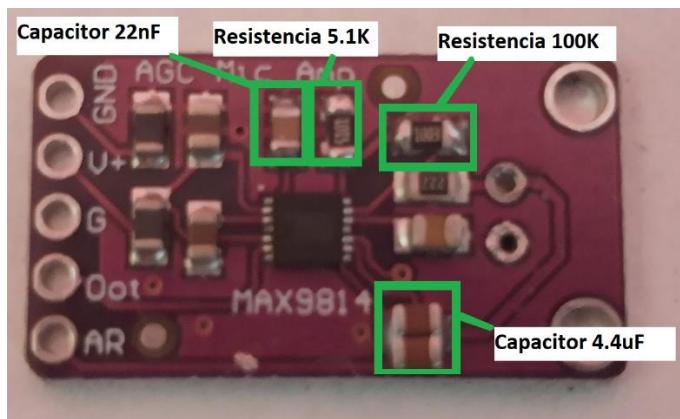


Figura 117.- Componentes modificados en la placa de Adafruit MAX9814, (módulo C).

12.4 Voltaje de referencia offset

Es necesario introducir las variaciones de voltaje AC provenientes del MAX9814 a las etapas posteriores del circuito, para ello se utiliza una componente de offset ya que el circuito está diseñado para trabajar con voltajes positivos. Esta componente puede ser agregada en la terminal no inversora de los opamps usando un divisor de tensión ([Figura 118](#)).

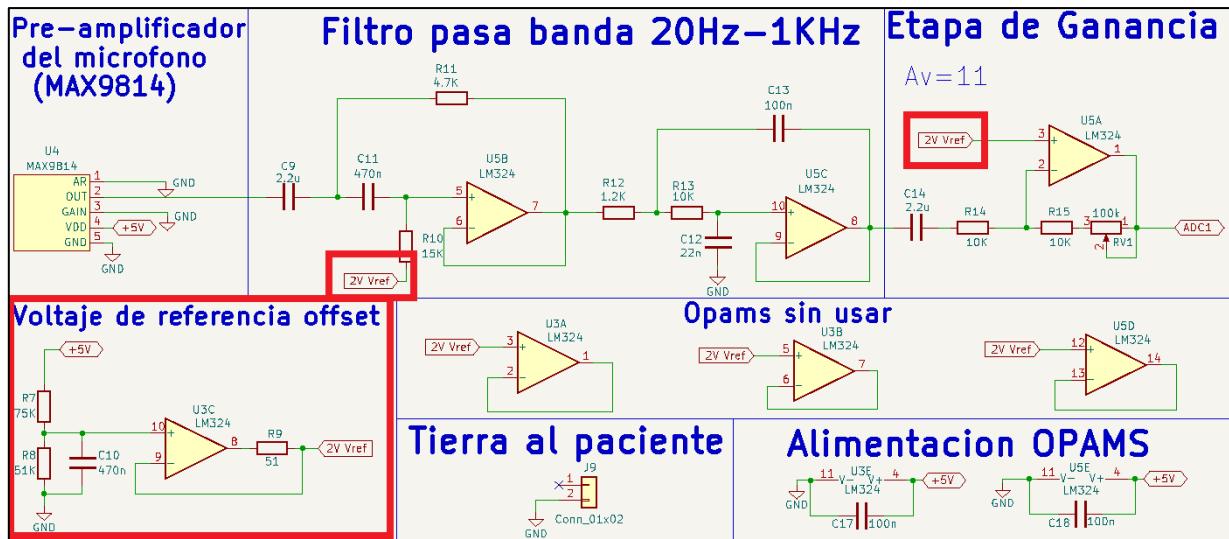


Figura 118.- Red de polarización de opams, (módulo C).

Según los voltajes de operación discutidos en la sección [8.7 Amplificador operacional LM324](#) se tiene como límite superior 3V e inferior 1V. Dado esto, la componente de offset deberá ubicarse entre estos valores de tal forma que se tenga el mismo rango de voltaje hacia los límites. Se propone una componente de offset de 2V teniendo $\pm 1V$ de variación.

Para calcular las resistencias se simplificará el circuito, teniendo un divisor de voltaje simple ([Figura 119](#)).

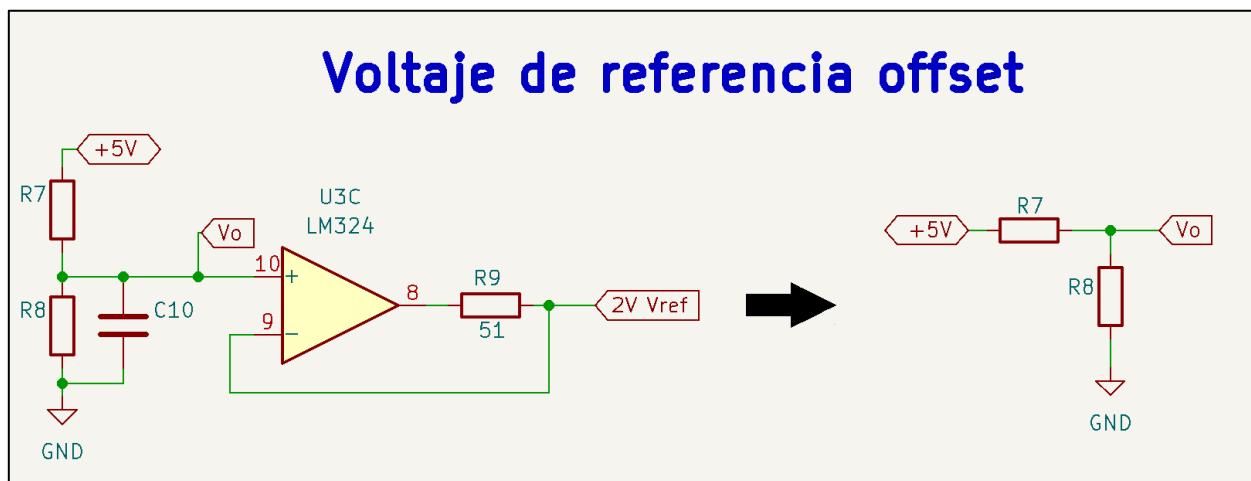


Figura 119.- Divisor de tensión, voltaje de referencia offset, (módulo C).

Conociendo que:

$$V_o = \frac{R_8}{R_7 + R_8} 5V \rightarrow 2V = \frac{R_8}{R_7 + R_8} 5V \rightarrow 2R_7 + 2R_8 = 5R_8 \rightarrow R_7 = \frac{3}{2}R_8$$

El valor de R7 debe ser 3/2 veces el valor de R8. Se propone usar un valor de R7=75KΩ y R8=51KΩ

Se agrega el capacitor C10, el cual filtra el ruido térmico y cualquier tipo de perturbación que pueda tener la fuente de alimentación.

El capacitor C10 y las resistencias R7 y R8 crean un filtro pasa bajas. La frecuencia de corte de este filtro deberá ser inferior a 20Hz para no afectar el desempeño del circuito de audio. Proponiendo un capacitor de 470nF, se tiene:

$$f_c = \frac{1}{2\pi(75K\Omega||51K\Omega)470nf} = 11.15Hz$$

Finalmente, el circuito de voltaje de referencia offset puede ser visto a continuación ([Figura 120](#)).

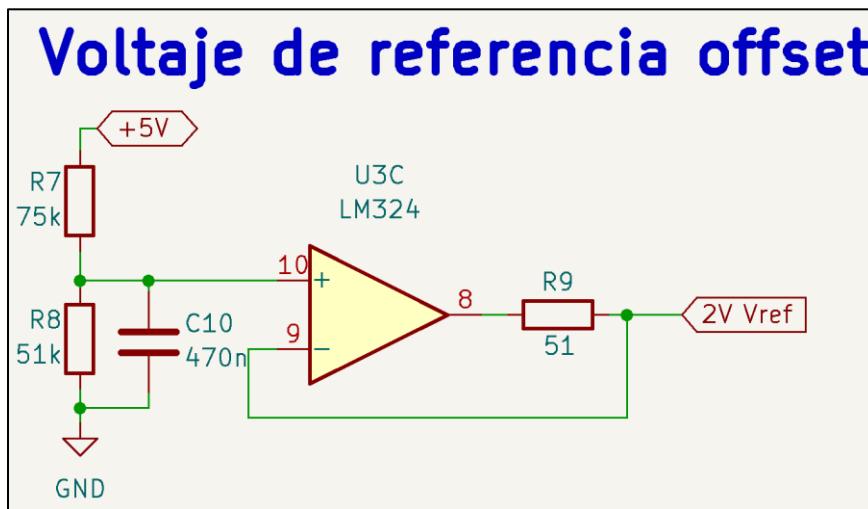


Figura 120.- Voltaje de referencia offset con valores, (módulo C).

12.5 Filtro pasa banda 20Hz-1KHz

Este filtro se divide en dos secciones:

- Filtro pasa altas 20Hz
- Filtro pasa bajas 1KHz

En ambos filtros se usan dos polos con una configuración Sallen-Key de tipo Butterworth. Los valores de los componentes fueron obtenidos gracias al software de “Filter Pro”. Los filtros obtenidos por este medio tienen como base una alimentación simétrica. Para pasar de una fuente dual a unipolar se hace uso de las notas de aplicación de Texas Instruments “*Single-supply, 2nd-order, Sallen-Key band-pass filter circuit*” [21].

Es necesario agregar una componente de offset en la entrada del filtro pasa altas, el circuito con esta modificación puede ser visto a continuación ([Figura 121](#)).

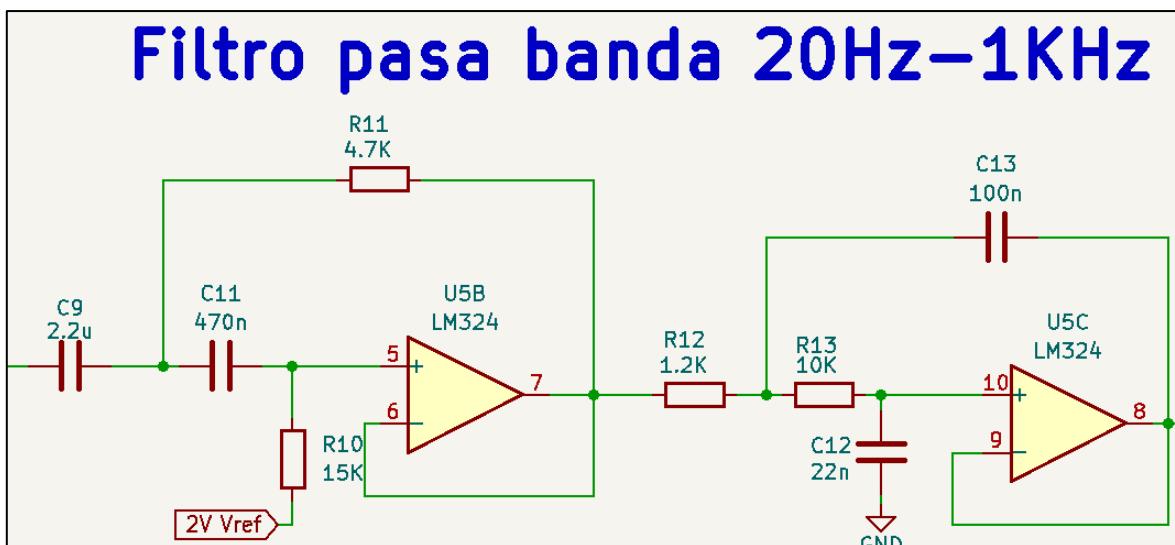


Figura 121.- Filtro pasa bandas 20Hz-1KHz, (módulo C).

De igual forma, se agrega un capacitor de acople C9 para introducir las variaciones de AC de partes anteriores del circuito en el filtro pasa bandas.

12.5.1 Filtro pasa altas 20Hz

Para el filtro pasa altas, se eligió una frecuencia de corte de 20Hz. Según los datos proporcionados por el software, se necesita un amplificador con una ganancia ancho de banda mínima de 2KHz ([Figura 122](#)). Se propone usar un LM324 ya que posee un GBP de 1.2MHz.

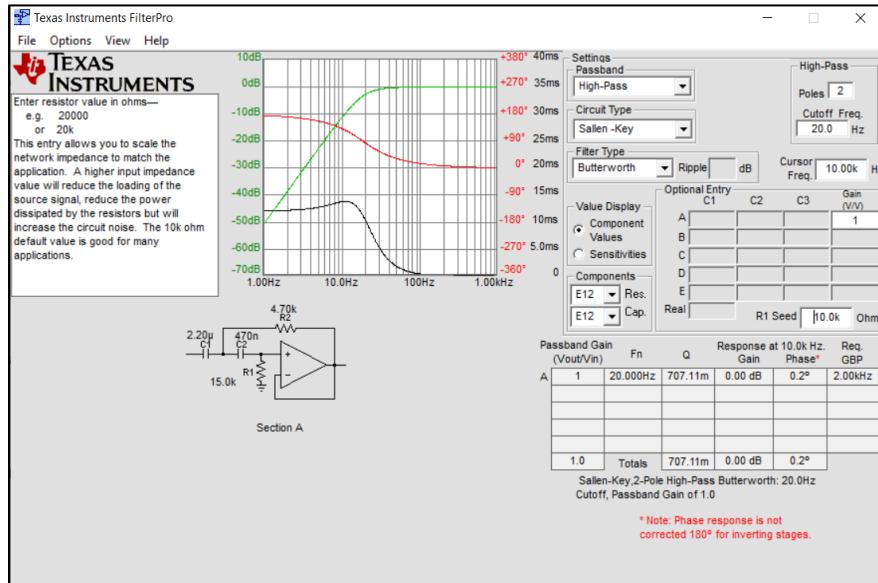


Figura 122.- Filtro pasa altas fc=20Hz, (módulo C).

12.5.2 Filtro pasa bajas 1KHz

Para el filtro pasa bajas, la frecuencia de corte escogida es de 1Khz. Para realizar este filtro se necesita una ganancia-ancho de banda de 100Khz ([Figura 123](#)). Se propone usar el LM324 ya que su GBP es de 1.2MHz.

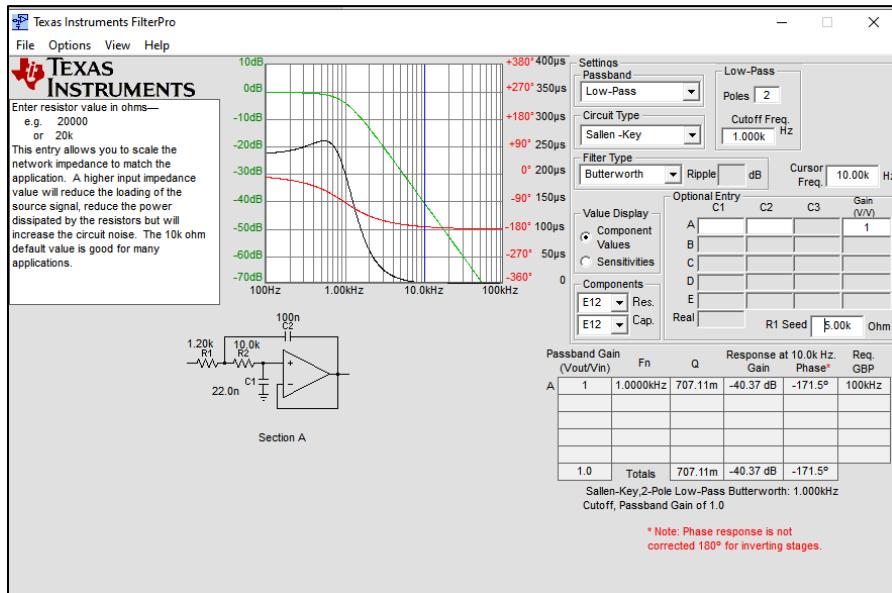


Figura 123.- Filtro pasa bajas fc=1Khz, (módulo C).

12.6 Etapa de ganancia

Por último, se amplifica la señal filtrada para hacerla coincidir con los rangos de voltaje establecidos en la sección [8.7 Amplificador operacional LM324](#). Según la sección [12.3.8 Ganancia del circuito](#), el voltaje de salida de la señal es de 1.54Vpp, ya que este voltaje es cercano a los 2Vpp propuestos en el rango de valores, se utilizará una ganancia máxima $A_v=1$ hasta $A_v=11$.

Según la hoja de datos del LM324 [16] la ganancia en lazo abierto en 1KHz es de 40dB o $A_v=100$ ([Figura 124](#)), por lo cual se utilizará una única etapa de ganancia ([Figura 125](#)).

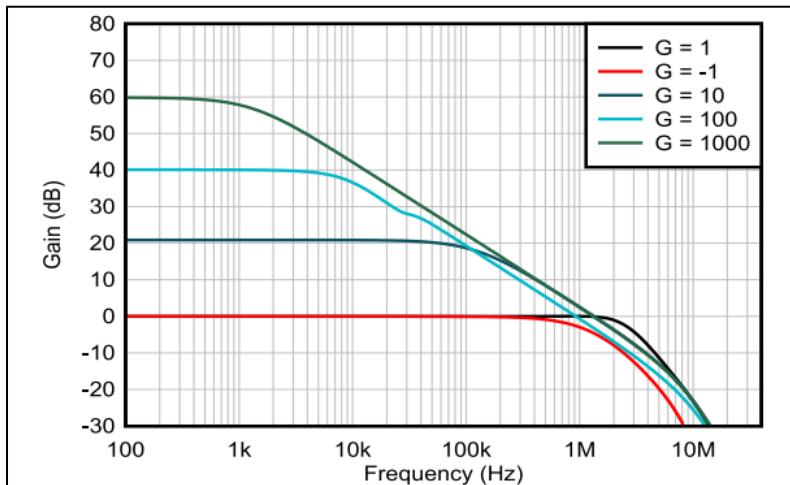


Figura 124.- Relación ganancia-ancho de banda en lazo abierto LM324, (módulo C).

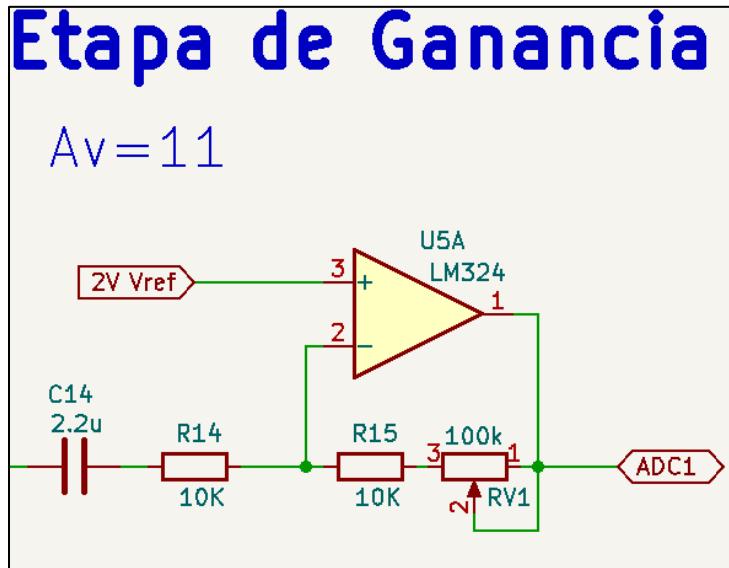


Figura 125.- Etapa de ganancia, (módulo C).

De manera similar a la sección [11.6 Etapa de ganancia](#), para obtener una ganancia unitaria al ajustar el potenciómetro a su valor mínimo (corto circuito), es necesario que el valor de la resistencia en el lazo de retroalimentación sea igual al de la resistencia de entrada de la señal, por lo cual se hace uso de la resistencia R15, la cual se mantiene fija en todo momento, teniendo que:

$$Av = -\frac{R_{15} + R_{V1}}{R_{14}} = -\frac{R_{15} + 0}{R_{14}} = -\frac{R_{15}}{R_{14}} = -1 \rightarrow R_{14} = R_{15}$$

Para calcular la ganancia variable de 11, se ajusta el potenciómetro al máximo valor, teniendo:

$$Av = -\frac{R_{15} + R_{V1}}{R_{14}} \rightarrow R_{15} + R_{V1} = (Av)R_{14} \rightarrow R_{V1} = (Av)R_{14} - R_{15}$$

Sabiendo que $R_{14}=R_{15}$ y que $Av=11$

$$R_{V1} = (11)R_{15} - R_{15} \rightarrow R_{V1} = 10R_{15}$$

La resistencia R_{V1} debe ser 10 veces más grande que R_{15} , se propone una resistencia $R_{14}=R_{15}$ de $10K\Omega$ y R_{V1} de $100K\Omega$.

Por último, los capacitores C_{14} se encarga de acoplar la señal de entrada la etapa anterior.

12.7 Módulo Bluetooth EGBT-045MS

Según la sección [8.12 Módulo EGBT-045MS](#), el voltaje de trabajo del módulo EGBT-045MS es de 3.3V. Según la sección [8.11 Módulo bluetooth HC-05 \(ZS-040\)](#), el ZS-040 implementa cierta electrónica en sus terminales para configurar el módulo.

Para acoplar el bluetooth con la tarjeta del módulo C, se usará la documentación “*Bluetooth Modules Wireless UART Cable Replacement EGBT-045MS EGBT-046S*” [19]

12.7.1 Pin de transmisión de datos TXD

Sabiendo que los voltajes de trabajo de este pin son de 0V como voltaje bajo y 3.3V como voltaje alto y según la sección [8.9 Microcontrolador Attiny 85](#), los voltajes de entrada del microcontrolador son:

- Entrada baja -> [-0.5 , 1.5]V
- Entrada alta -> [3.0 , 5.5]V

Los voltajes de salida del módulo Bluetooth entran en el rango de voltajes de entrada del attiny85, por lo cual es posible conectar estos pines de forma directa ([Figura 126](#)).

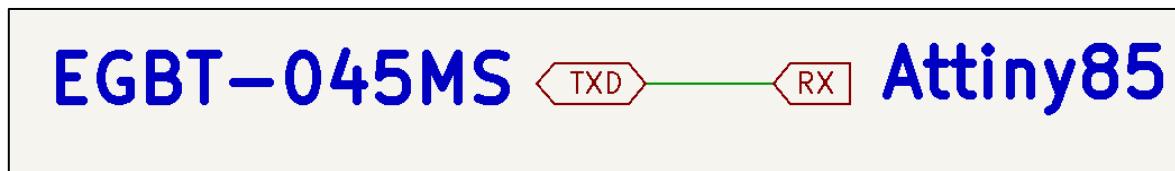


Figura 126.- Acoplamiento del pin TXD del EGBT-045MS, (módulo C).

12.7.2 Pin de recepción de datos RXD

Según la sección [8.9 Microcontrolador Attiny 85](#), los voltajes de salida del attiny85 son:

- Salida baja -> [0.0 , 0.6]V
- Salida alta -> [4.3 , 5.0]V

El voltaje de salida alta se encuentra por arriba del voltaje de entrada alta del EGBT-045MS, lo cual podría dañar al circuito. Para acoplar esta terminal se hará uso de un divisor de tensión ([Figura 127](#)).

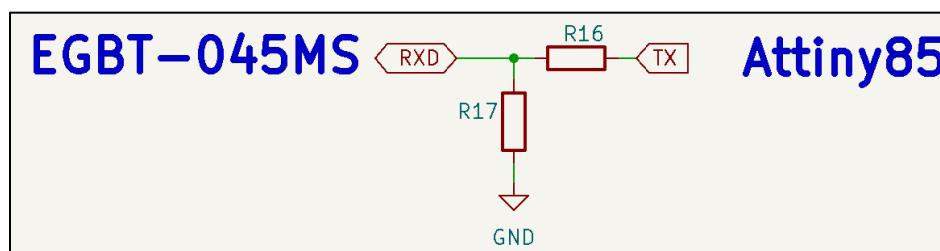


Figura 127.- Divisor de tensión para acoplar el pin RXD del EGBT-045MS, (módulo C).

Se sabe que el pin TX del attiny 85 tendrá un voltaje de 5V y que el voltaje de trabajo del HC-05 en el pin RX es de 3.3V. Con base en el divisor de la [Figura 127](#), se tiene:

$$R_{XD} = \frac{R_{17}}{R_{16} + R_{17}} T_X \rightarrow 3.3V = \frac{R_{17}}{R_{16} + R_{17}} 5V \rightarrow 3.3R_{16} + 3.3R_{17} = 5R_{17}$$

$$R_{16} = \frac{1.7}{3.3} R_{17} \rightarrow R_{16} \approx \frac{1}{2} R_{17}$$

El valor de R16 debe ser ≈ 0.5 el valor de R17. Se propone R16=100KΩ y R17=200KΩ.

12.7.3 Pin indicador led de estatus

De manera similar al módulo visto en la sección [8.11 Módulo bluetooth HC-05 \(ZS-040\)](#), se incluirá un led en el pin 31 para indicar los diferentes estados del módulo Bluetooth. Este indicador consistirá en una resistencia de 1K y un led ([Figura 128](#)).

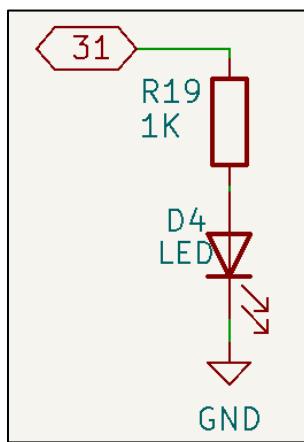


Figura 128.- Indicador de status Bluetooth, (módulo C).

12.7.4 Pulsador de modos AT

De manera similar al módulo visto en la sección [8.11 Módulo bluetooth HC-05 \(ZS-040\)](#), se incluirá un pulsador para cambiar entre el modo AT1 y AT2. El pin 34 estará conectado con una resistencia pull-down hacia el pulsador ([Figura 129](#)).

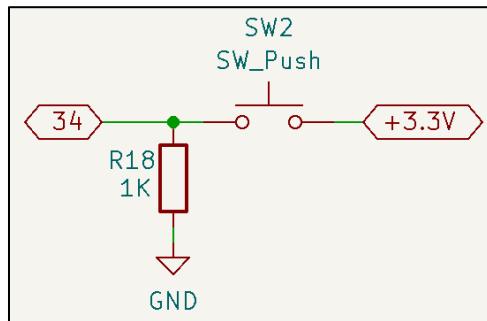


Figura 129.- Pulsador de modos AT, (módulo C).

12.7.5 Conexiones del módulo EGBT-045MS

Finalmente, el módulo EGBT-045MS con sus respectivas conexiones puede verse a continuación ([Figura 130](#)).

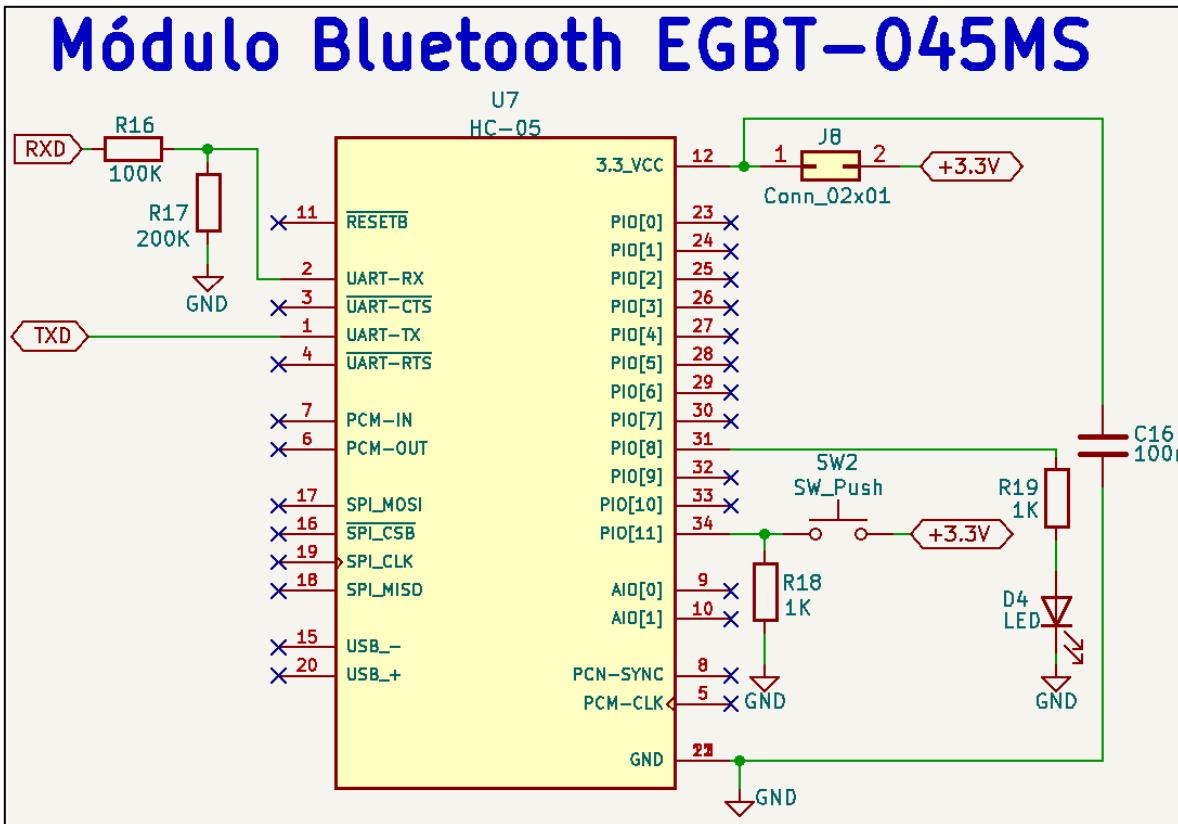


Figura 130.- Conexiones del módulo Bluetooth EGBT-045MS, (módulo C).

Se agregan 2 terminales en el pin de entrada de voltaje para poder encender y apagar el módulo según sea necesario, esto con el fin de probar por separado cada parte de la placa al momento de soldarla.

12.8 Microcontrolador attiny 85

El microcontrolador utilizado en este proyecto es un attiny85 de la marca Atmel, se eligió este microcontrolador ya que cuenta con 6 puertos de entrada/salida y convertidores ADC, además de permitir el ser programado bajo el IDLE de Arduino.

El trabajo del microcontrolador es el que transformar la señal analógica del circuito de acondicionamiento en un valor digital y mandarlo a través del puerto serial al módulo Bluetooth.

Para programar el attiny85 es necesario contar con una placa Arduino externa ya que ésta realizará el trabajo de ISP para programar el microcontrolador.

12.8.1 Voltaje de referencia ADC

Para no perder resolución en la conversión analógica-digital de la señal del corazón se propone modificar el voltaje de referencia del ADC a través del pin AREF del microcontrolador, (para más información ver la sección [8.9 Microcontrolador Attiny 85](#)).

Según los límites establecidos en la sección [8.7 Amplificador operacional LM324](#), el voltaje máximo de trabajo será de 3V, por lo cual se propone dar un margen de error referenciando el ADC a 3.3V.

Gracias al regulador LP2985-33 usado para alimentar al módulo bluetooth, se tiene un voltaje de 3.3V estable, este voltaje será conectado directamente al pin AREF del microcontrolador.

12.9 Conexiones del attiny 85

Debido a que el attiny85 cuenta únicamente con 6 pines de entrada/salida, se tienen dos modos de configuración, el modo programador y el modo de funcionamiento normal. Para elegir entre estos modos, se conectan terminales dobles en cada uno de sus pines, esto con el fin de conectar o desconectar el microcontrolador del circuito a través de jumpers conectados en sus terminales, en la [Figura 131](#) se muestran de color azul las terminales a puentear.

De esta forma, es posible conectar el microcontrolador al circuito de acondicionamiento puenteando sus terminales con jumpers (modo de funcionamiento normal), o dejar las terminales libres para ser programado de manera externa (modo programador).

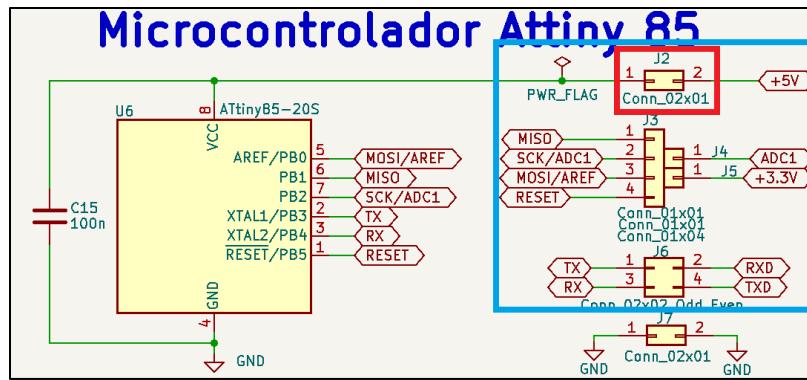


Figura 131.- Terminales de conexión del microcontrolador attiny85, (módulo C).

En la [Figura 131](#), de color rojo se muestra una terminal doble entre la conexión Vcc a +5V del attiny85, esta conexión sirve para encender o apagar el microcontrolador independientemente del circuito de acondicionamiento, esto con fin de probar por separado diferentes partes del circuito al momento de ensamblarlo.

12.9.1 Conexión al programador

Para conectar el attiny85 al programador, es necesario quitar los jumpers del microcontrolador, dejando las terminales al aire. Se conectan de manera externa cables hacia las terminales SCK/AD1, MISO, MOSI/AREF y RESET ([Figura 89](#)). El diagrama de conexiones será discutido en la sección [14 Programación del attiny85](#).

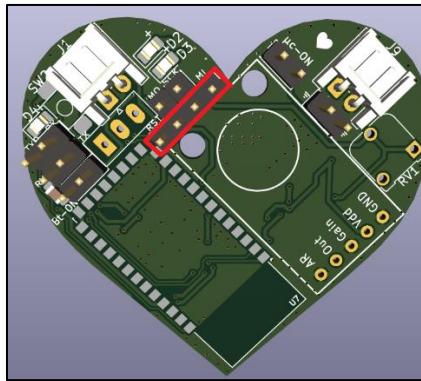


Figura 132.- Terminales de programación, (módulo C).

12.9.2 Conexión al circuito de acondicionamiento

La terminal MOSI/AREF se conecta a la salida del voltaje de referencia ADC visto en la sección [12.8.1 Voltaje de referencia ADC](#) siendo esta terminal en donde se suministra el voltaje de referencia para el convertidor analógico-digital. Por otra parte, la terminal SCK/AD1 se conecta a la salida del circuito de acondicionamiento, siendo el pin entrada de la señal analógica al microcontrolador ([Figura 133](#)).

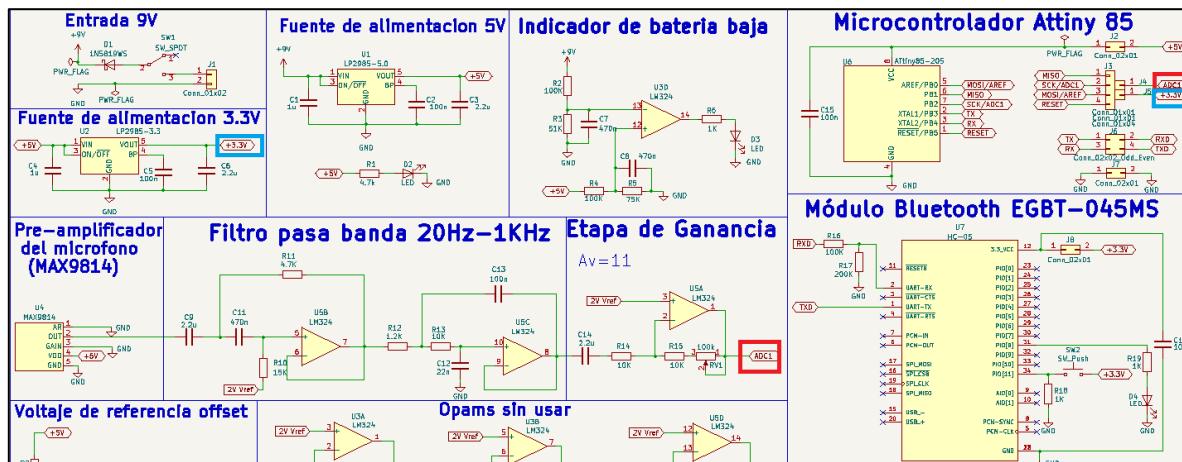


Figura 133.- Conexiones del microcontrolador al circuito de acondicionamiento, (módulo C).

En la [Figura 133](#) se tiene:

- De color rojo la conexión en entre el circuito de acondicionamiento y el microcontrolador.
- De color azul la conexión entre el voltaje de referencia y el microcontrolador.

12.9.3 Conexión al módulo Bluetooth

Las conexiones al módulo Bluetooth fueron discutidas en la sección [12.7 Módulo Bluetooth EGBT-045MS](#). Las conexiones para configurar el módulo Bluetooth por separado, así como los comandos AT para programarlo serán discutidos en la sección [15 Programación del módulo Bluetooth HC-05 / EGBT-045MS](#).

12.10 Conexión de tierra al paciente

Para eliminar cualquier tipo de perturbación que pueda ser inducida por el paciente, se propone conectar un cable hacia GND, logrando referenciar la señal del circuito.

Según la hoja de datos del amplificador de instrumentación AD620 [12], para un circuito de aplicación biomédica ECG, se recomienda conectar una tierra virtual y una etapa de protección entre el circuito y los electrodos del paciente ([Figura 134](#)).

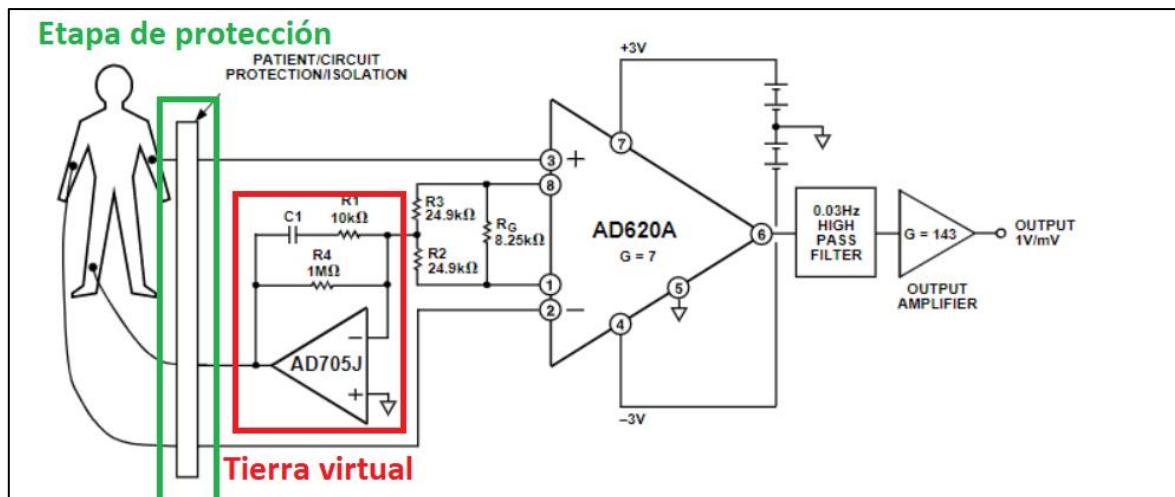


Figura 134.- Circuito de monitor ECG, (módulo C).

Debido a las limitaciones de los amplificadores LM324 utilizados en el circuito de acondicionamiento, no es posible conseguir una salida hacia GND con una alimentación unipolar. Para generar una tierra virtual sería necesario integrar una nueva etapa al circuito que permita alimentaciones bipolares o modificar los valores previamente calculados para agregar esta nueva referencia. Dadas las complicaciones no se integrará la etapa de tierra virtual, sin embargo, la etapa de protección entre GND y el paciente puede ser generada.

Se propone utilizar una pulsera antiestática conectada desde GND hasta alguna extremidad del paciente. La pulsera cuenta con una impedancia de aproximadamente $1M\Omega$, permitiendo una protección de corto circuito y a su vez, referenciar el circuito con el paciente, eliminando cualquier tipo de ruido que este pueda generar.

Para conectar la pulsera se hace uso de un conector JST XH2.54 el cual está conectado directamente a tierra ([Figura 135](#)).

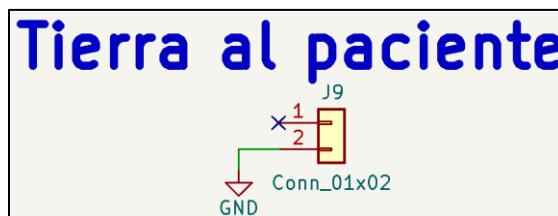


Figura 135.- Tierra al paciente, (módulo C).

13 Arduino como ISP

Para programar al microcontrolador attiny 85 es necesario contar con una placa de Arduino externa la cual realizará el trabajo de ISP. Para configurar la placa, se conecta el Arduino a utilizar al puerto USB de la computadora, se abre el IDE de Arduino y se da clic en “Archivo>Ejemplos>ArduinoISP” ([Figura 136](#)).

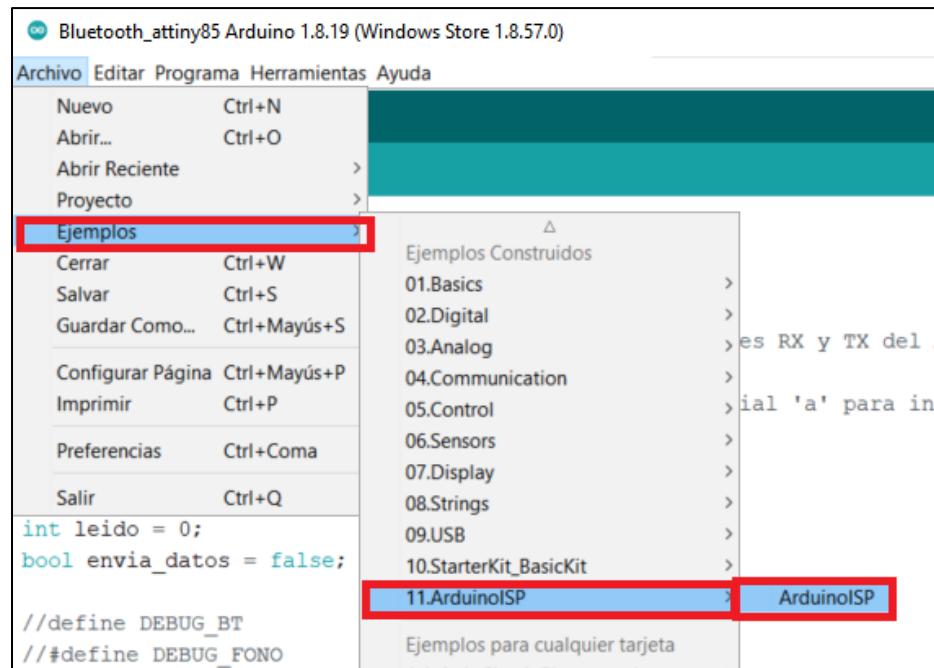


Figura 136.- Ruta del ejemplo ArduinoISP

Se elije el modelo de la placa Arduino dentro de la sección de “Herramientas>Placas”, así como el puerto serial donde se encuentra conectado el Arduino. Por último, se verifica que dentro de “Herramientas>Programador” se tenga seleccionado AVRISP mkII, siendo esta la configuración por defecto ([Figura 137](#)). Una vez hecho esto, se da clic en compilar y subir el programa.

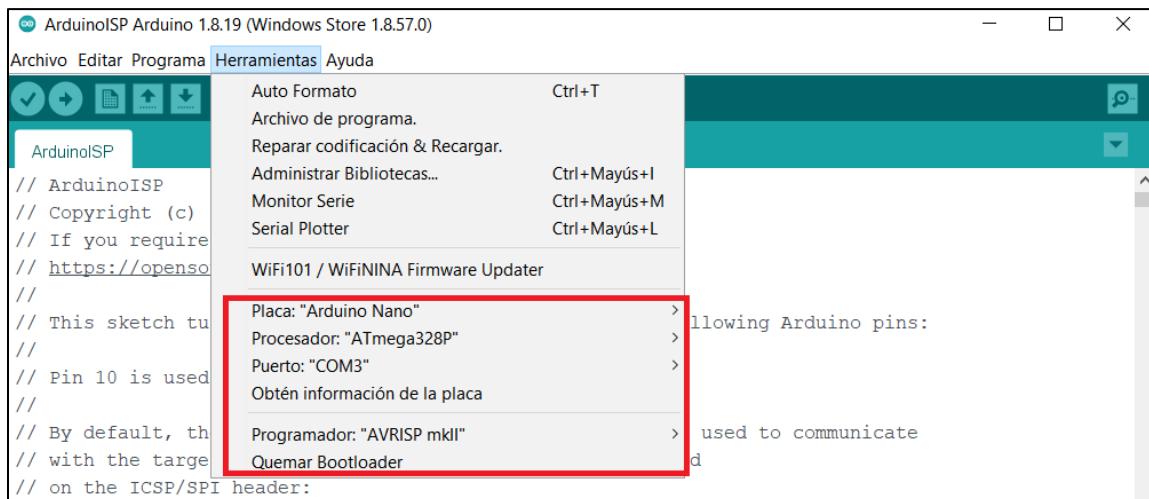


Figura 137.- Configuración del IDE del Arduino para Arduino como ISP

14 Programación del attiny85

El attiny85 es un microcontrolador que no es soportado de manera nativa en el software de Arduino, por lo cual, es necesario descargar librerías externas que permitan programarlo.

Para descarga la librería se da clic en “Herramientas>Placa>Gestor de tarjetas” ([Figura 138](#)).

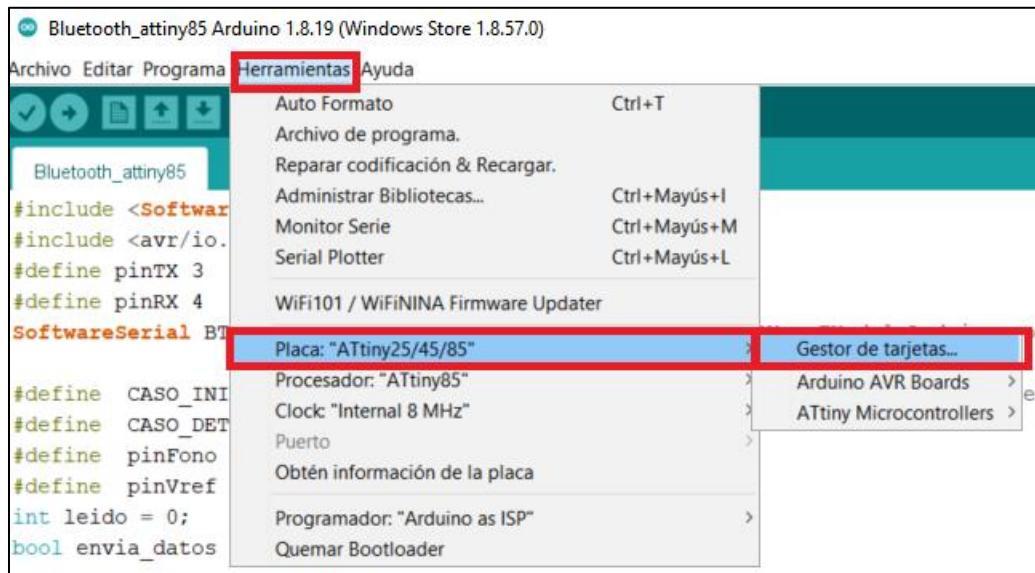


Figura 138.- Ruta del gestor de librerías, Arduino

Al acceder al gestor de librerías, se debe escribir “attiny85” en el buscador. La librería utilizada en el proyecto es la creada por “David A. Mellis”. Se selecciona esta librería y se da clic en Instalar ([Figura 139](#)).



Figura 139.- Librería del attiny 85, Arduino

14.1 Conexiones del attiny85

El attiny85 cuenta con dos modos de operación: Puede ser programado dentro de la placa o conectado para funcionar con el circuito de acondicionamiento. Estos dos modos de operación pueden ser cambiados al puentear las terminales del módulo con jumpers ([Figura 140](#), [Figura 141](#) y [Figura 142](#)).

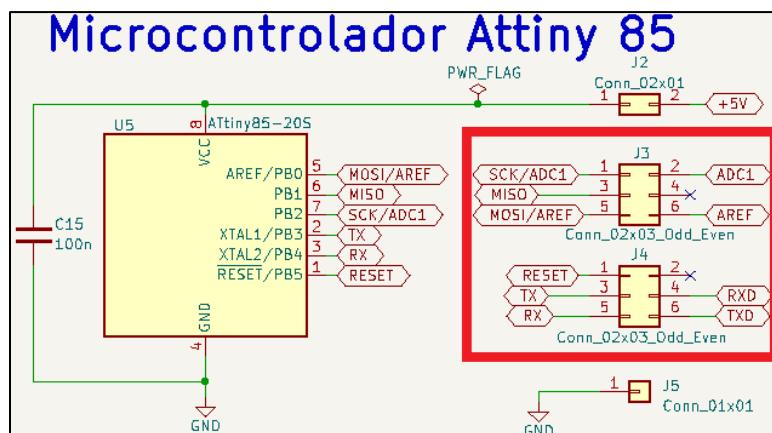


Figura 140.- Terminales de conexión del attiny85, (módulo A).

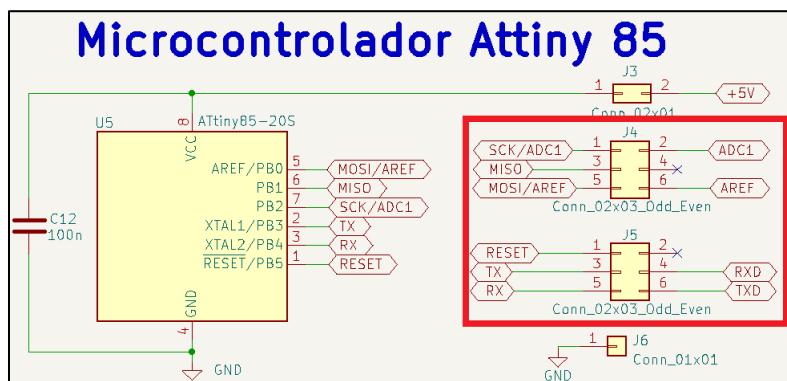


Figura 141.- Terminales de conexión del attiny85, (módulo B).

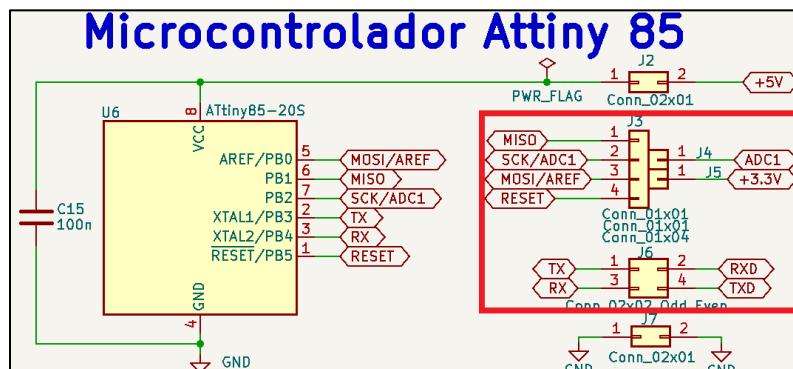


Figura 142.- Terminales de conexión del attiny85, (módulo C).

Nota: Para habilitar el attiny85 es necesario puentejar la terminal Vcc a +5V del microcontrolador o alimentar directamente con 5V la terminal Vcc.

14.2 Conexiones entre el attiny 85 y el Arduino Nano

Dentro del código “ArduinoISP” de la sección [13 Arduino como ISP](#), se encuentran comentarios sobre las conexiones que deben hacerse desde el Arduino hacia el microcontrolador. El diagrama de conexiones es mostrado a continuación ([Figura 143](#)).

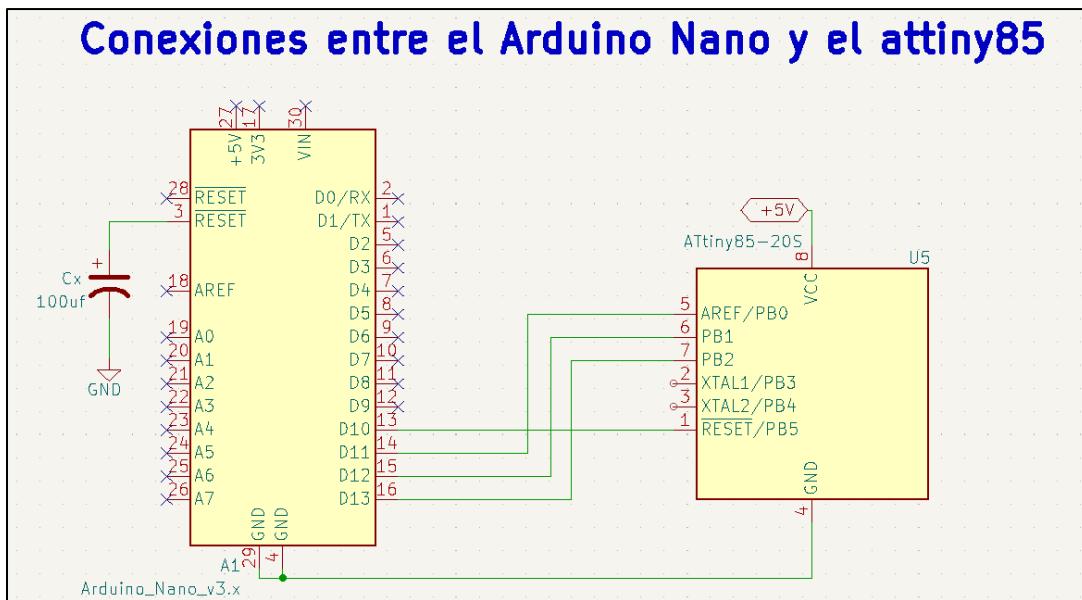


Figura 143.- Diagrama de conexiones para programación, attiny85

Las conexiones entre el Arduino y el microcontrolador pueden realizarse sobre la misma placa del circuito quitando los jumpers que unen las terminales:

- SCK/AD1 <-> ADC1
- MISO
- MOSI/AREF <-> AREF
- RESET

A estas terminales se conectan cables que van desde el Arduino Nano hasta el attiny85. Con base en el diagrama de la [Figura 143](#), se tienen las siguientes conexiones:

Tabla 2.- Conexiones entre el attiny85 y el Arduino Nano

Pin del attiny85	Pin del Arduino Nano
RESET	10
MOSI/AREF	11
MISO	12
SCK/ACD1	13
Vcc	Vcc *
GND	GND

Nota: Se debe conectar un capacitor electrolítico externo entre la terminal de RESET y GND del Arduino ya que el microcontrolador puede presentar problemas de ruido al ser programado, se recomienda un valor entre 10uf-220uf.

* La terminal Vcc del Arduino se conecta en caso de que la terminal Vcc<->+5V del attiny85 se encuentre desconectada.

Para conocer la distribución de las terminales del attiny85 en cada placa (módulo A,B,C), **se recomienda ver los archivos del circuito impreso en Kicad**.

14.3 Carga del programa (attiny85)

Una vez configurado el Arduino y teniendo las conexiones correspondientes entre el Arduino y el microcontrolador, es necesario cambiar dentro del entorno el tipo de placa a programar, así como el programador usado ([Figura 144](#)):

- En “Herramientas>Placas”, se selecciona la placa a programar, siendo “ATtiny25/45/85”
- En “Herramientas>Procesador”, se elige “Attiny85”
- En “Herramientas>Puerto Serial”, se debe elegir el puerto donde se encuentra el Arduino conectado.
- En “Herramientas>Clock”, se puede elegir cualquier opción que permita tener un oscilador interno.
- En “Herramientas>Programador” se elige la opción de “Arduino as ISP”.

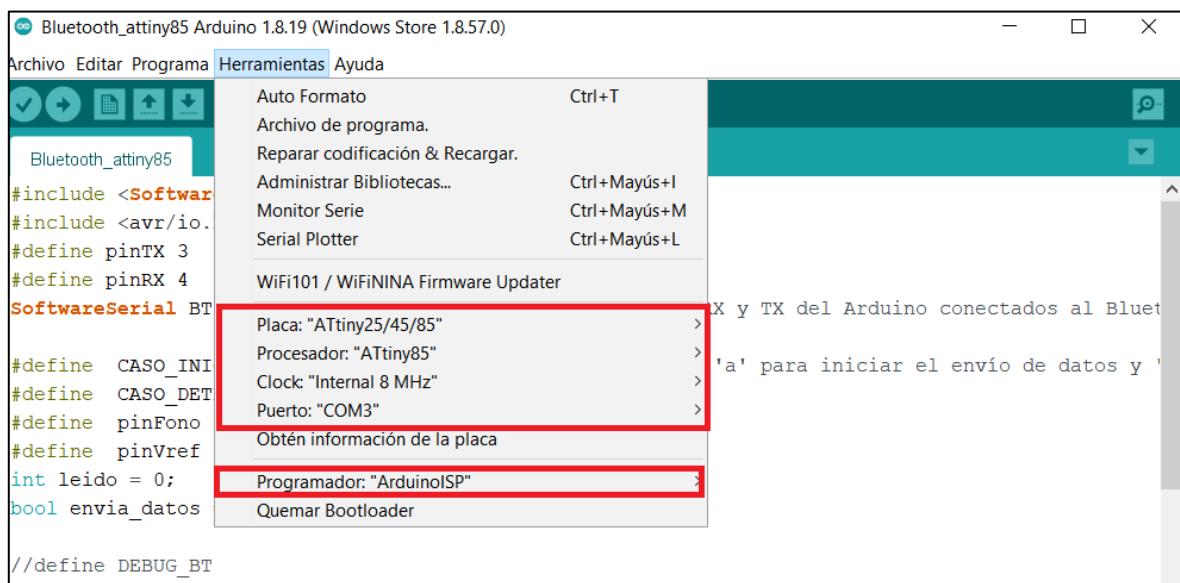


Figura 144.- Configuración del IDLE de Arduino para el attiny85.

En caso de ser la primera vez que se carga un sketch de Arduino al microcontrolador, **es necesario cargar el Bootloader**, de esta forma se tiene configurado al attiny85 para trabajar con el IDE de Arduino. Para esto se da clic en “Herramientas>Quemar Bootloader” ([Figura 145](#)).

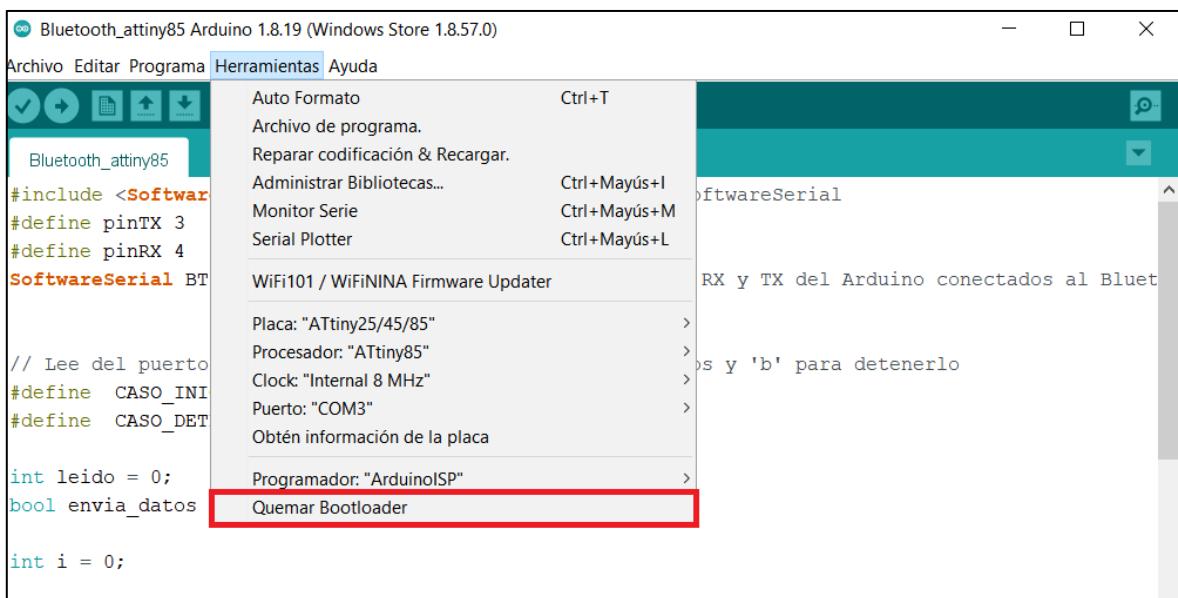


Figura 145.- Carga del Bootloader

Por último, se manda el siguiente script para programar el attiny85:

<pre>#include <SoftwareSerial.h> #define pinTX 3 #define pinRX 4 #define PIN_VOUT A1 SoftwareSerial BT(pinRX,pinTX); // Lee del puerto serial 'a' para iniciar el envío de datos y 'b' // para detenerlo #define CASO_INICIO 'a' #define CASO_DETENER 'b' int leido = 0; bool envia_datos = false; void setup() { BT.begin(38400); analogReference(EXTERNAL); }</pre>	<pre>void loop() { if(BT.available()){ leido = BT.read(); if(leido == CASO_INICIO) envia_datos = true; if(leido == CASO_DETENER) envia_datos = false; } if(envia_datos){ i = analogRead(PIN_VOUT); BT.write(highByte(i)); BT.write(lowByte(i)); } }</pre>
------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

15 Programación del módulo Bluetooth HC-05 / EGBT-045MS

El módulo Bluetooth HC-05 viene configurado por defecto como:

- Modo: Esclavo
- Nombre del dispositivo: HC-05
- Código de emparejamiento: 1234
- Baud Rate: 9600

Es posible modificar estos parámetros a través de comandos AT, para fines prácticos, se hace uso de un Arduino Nano conectado a la computadora, gracias a la cual se mandarán los comandos AT a través del puerto serie.

15.1 Conexiones del módulo Bluetooth

El módulo Bluetooth cuenta con dos modos de operación: Puede ser programado dentro de la placa o conectado para funcionar con el attiny85. Estos dos modos de operación pueden ser cambiados al puenteear las terminales del módulo con jumpers ([Figura 146](#), [Figura 147](#) y [Figura 148](#)).

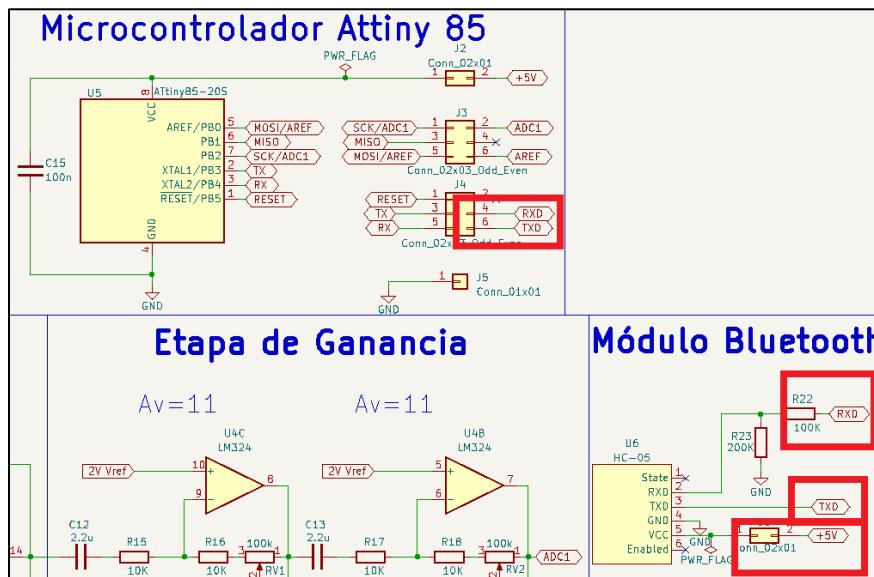


Figura 146.- Terminales de conexión del módulo Bluetooth, (módulo A).

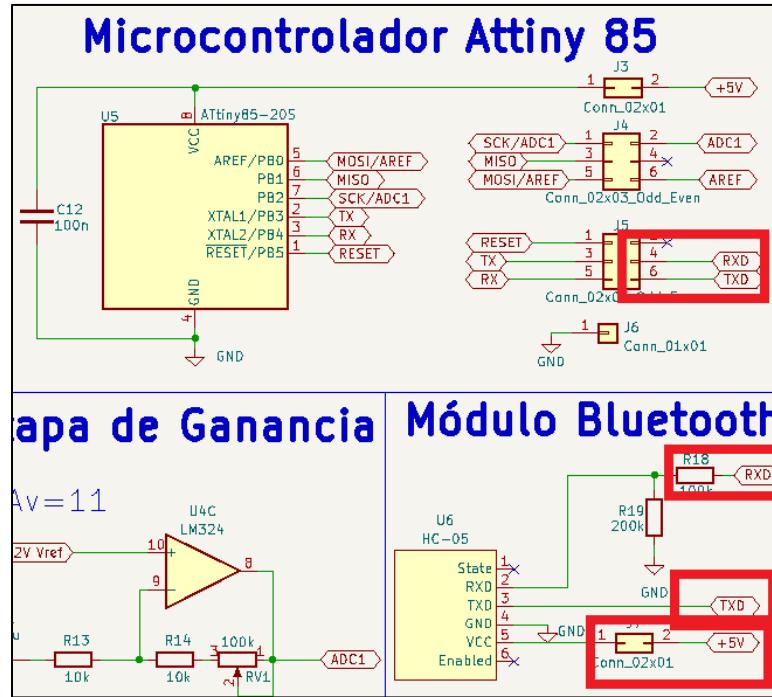


Figura 147.- Terminales de conexión del módulo Bluetooth, (módulo B).

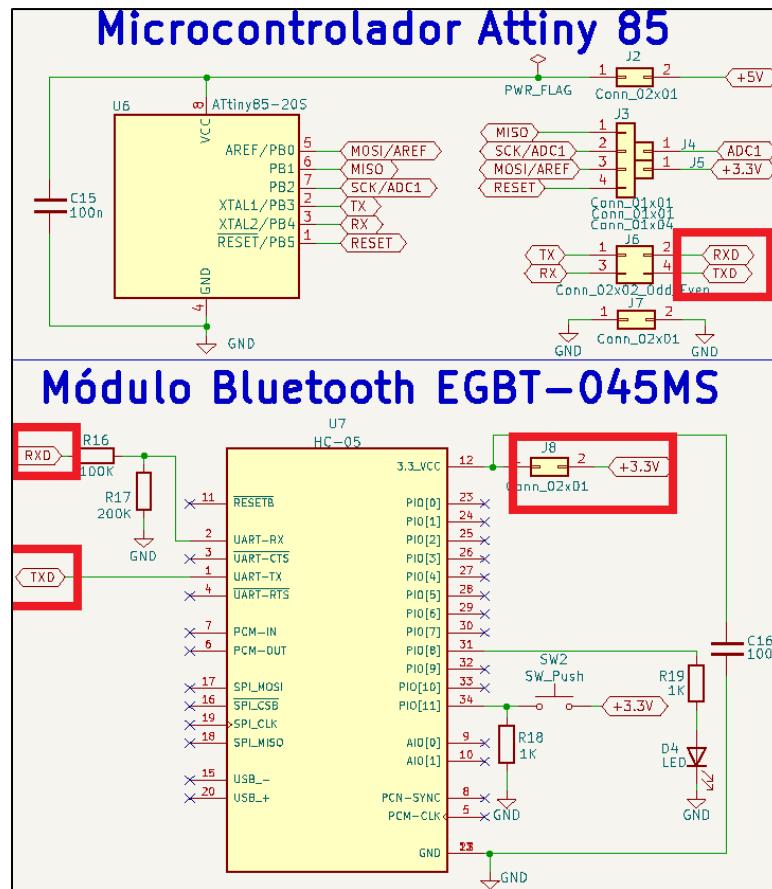


Figura 148.- Terminales de conexión del módulo Bluetooth, (módulo C).

Nota: Para habilitar el módulo Bluetooth es necesario puentejar la terminal Vcc a +5V del HC-05 o alimentar directamente con 5V la terminal Vcc del módulo Bluetooth.

15.2 Conexiones entre el módulo Bluetooth y el Arduino Nano

Para conectar el Arduino Nano al módulo Bluetooth, se desconectan los jumpers de las terminales:

- TX/RXD
- RX/TXD

Las terminales de transmisión y recepción del módulo Bluetooth se encuentran desconectadas del attiny85, permitiendo conectar cables externos del Arduino Nano hacia el módulo Bluetooth (Modo programador).

La tabla de conexiones entre el módulo Bluetooth y el Arduino Nano, se muestra a continuación ([Tabla 3](#)).

Tabla 3.- Conexiones entre el HC-05 y el Arduino Nano

<i>Pin del HC-05</i>	<i>Pin del Arduino Nano</i>
TXD	10
RXD	11
Vcc	Vcc *
GND	GND

* La terminal Vcc del Arduino se conecta en caso de que la terminal Vcc<->+5V del módulo Bluetooth se encuentre desconectada.

15.3 Carga del programa (HC-05/EGBT-045MS)

Para evitar problemas en cuanto a la velocidad serial con la que se comunica el módulo Bluetooth y el Arduino, se propone entrar en el “Modo AT 2”, donde la velocidad es de 38400 baudios. Para entrar en este modo basta con tener presionado el botón del módulo al momento de alimentarlo, una vez encendido se puede soltar. Para verificar que eventualmente se entró en este modo, se comprueba que el led indicador parpadee lentamente.

A continuación, se muestra el código usado para enviar los comandos AT, este código se carga en el Arduino Nano a través del IDLE de Arduino seleccionando el tipo de placa, procesador, programador y el puerto donde éste se encuentre. ([Figura 149](#)).

```

#include <SoftwareSerial.h>
SoftwareSerial BT(10,11); // Pin RX y TX del Arduino

void setup(){
    BT.begin(38400); // Inicializacion del puerto Serie BT
    Serial.begin(38400); // Inicializacion del puerto Serie
}

void loop(){
    if(BT.available()){ // Si llega un dato por el puerto BT se envía al
        monitor serial
        Serial.write(BT.read());
    }

    if(Serial.available()){ // Si llega un dato por el monitor serial se
        envía al puerto BT
        BT.write(Serial.read());
    }
}

```

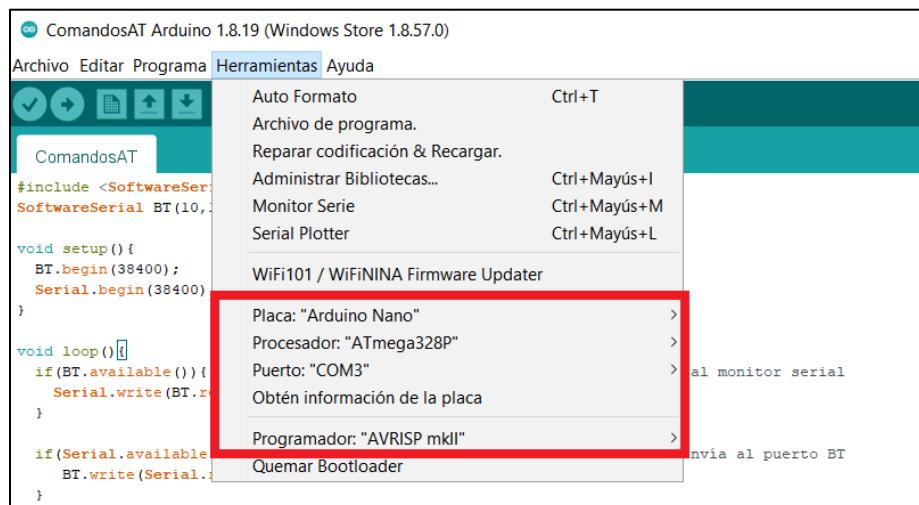


Figura 149.- Configuración del IDLE de Arduino para el Arduino Nano

Una vez cargado el programa, se abre el monitor serial configurándolo con terminación “NL & CR” a 38400 baudios. ([Figura 150](#)).

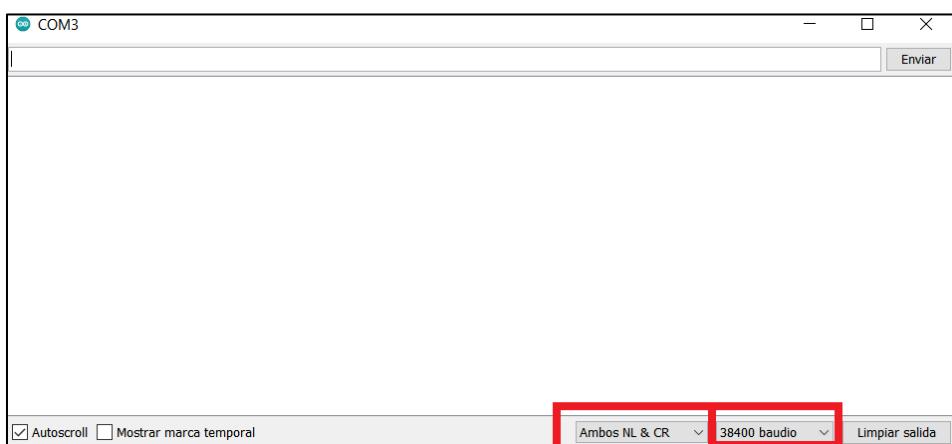


Figura 150.- Monitor serial para comandos AT

15.3.1 Test de comunicación

Para comprobar el estado del dispositivo Bluetooth se realiza un test de comunicación con los comandos AT:

- Enviar: AT
- Recibe: OK

Si al enviar el comando AT se recibe un OK como respuesta, se puede proseguir con los demás comandos.

15.3.2 Cambiar el nombre del módulo HC-05

Para cambiar el nombre, se tiene el comando “AT+NAME=<Nombre>,” donde “<Nombre>” se sustituye por el nombre a cambiar del módulo.

En este caso el nombre a asignar es “Fonocardiograma”, por lo cual el comando queda de la siguiente forma:

- Enviar: AT+NAME=Fonocardiograma
- Recibe: OK

15.3.3 Código de vinculación

Por defecto, el código de vinculación es “1234”. Para cambiar el código se usa el comando “AT+PSWD=<Pin>”, donde “<Pin>” se sustituye por la secuencia numérica a asignar.

En este caso se asigna el pin “0000”, por lo cual el comando queda de la siguiente forma:

- Enviar: AT+PSWD=”0000”
- Recibe: OK

15.3.4 Velocidad de comunicación

Por defecto, la velocidad es de 9600 baudios con Stop bit = 0 (un bit de parada) y sin Paridad. El comando para cambiar esta configuración viene dado por “AT+UART=<Baud>,<StopBit>,<Parity>”, donde “<Baud>” corresponde a los baudios, “<StopBit>” al bit de parada y “<Parity>” al bit de paridad.

En este caso, se busca una velocidad de transmisión de 38400, con 1 bit de parada y sin paridad, quedando el comando de la siguiente forma:

- Enviar: AT+UART=38400,0,0
- Recibe: OK

Finalmente se resetea el módulo con el comando “AT+RESET”, recibiendo un “OK” como respuesta.

16 Conexiones entre el módulo Bluetooth el attiny85

Para conectar el módulo Bluetooth al microcontrolador attiny85, se conectan jumpers en las terminales mostradas de color rojo de las secciones [14.1 Conexiones del attiny85](#) y [15.1 Conexiones del módulo Bluetooth](#) (Modo de funcionamiento normal).

17 Caracterización del hule espuma e implementación del arnés.

Para montar el circuito de acondicionamiento al paciente es necesario contar con un arnés el cual ejerza presión sobre el pecho con la campana del estetoscopio. El arnés implementado se encuentra compuesto de dos correas ajustables ([Figura 151](#)), una vertical y otra horizontal. De manera interna, apuntando hacia el corazón, cuenta con una zona de plástico endeble con un pequeño orificio, dentro de este, se inserta la campana de estetoscopio. ([Figura 152](#)).



Figura 151.- Sujetador para el circuito del fonocardiograma.



Figura 152.- Sujetador interno.

En la [Figura 153](#), se puede observar cómo va montado el circuito sobre el paciente, de color azul se encuentra la campana del fonocardiograma apuntando al pecho, de color rojo el circuito y de color verde la batería de 9V.

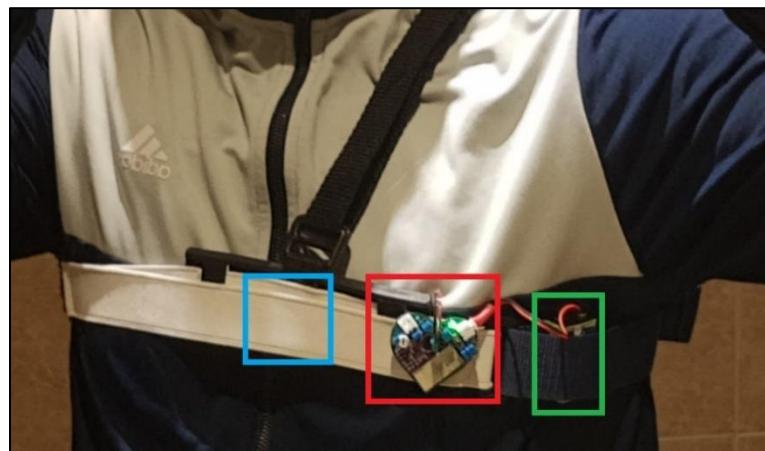


Figura 153.- Circuito del fonocardiograma montado sobre el pecho.

En las pruebas de funcionamiento, las tres versiones del circuito son susceptibles a golpes externos, el “Módulo A”, presenta una mayor presencia de señal basura al ser golpeada accidentalmente, mientras que el “Módulo B” y “Módulo C” siguen presentando este ruido, pero en menor medida.

Para tratar de mitigar la presencia de golpes externos se hace uso de un hule espuma. Para conocer si las frecuencias de interés no serán atenuadas por el hule espuma, éste será caracterizado acústicamente mediante un tubo de impedancias para determinar sus coeficientes acústicos. La espuma para caracterizar será una espuma comercial ([Figura 154](#)).



Figura 154.- Hule espuma

Al caracterizar el hule espuma, la gráfica de coeficientes de transmisión es la siguiente ([Figura 155](#)).

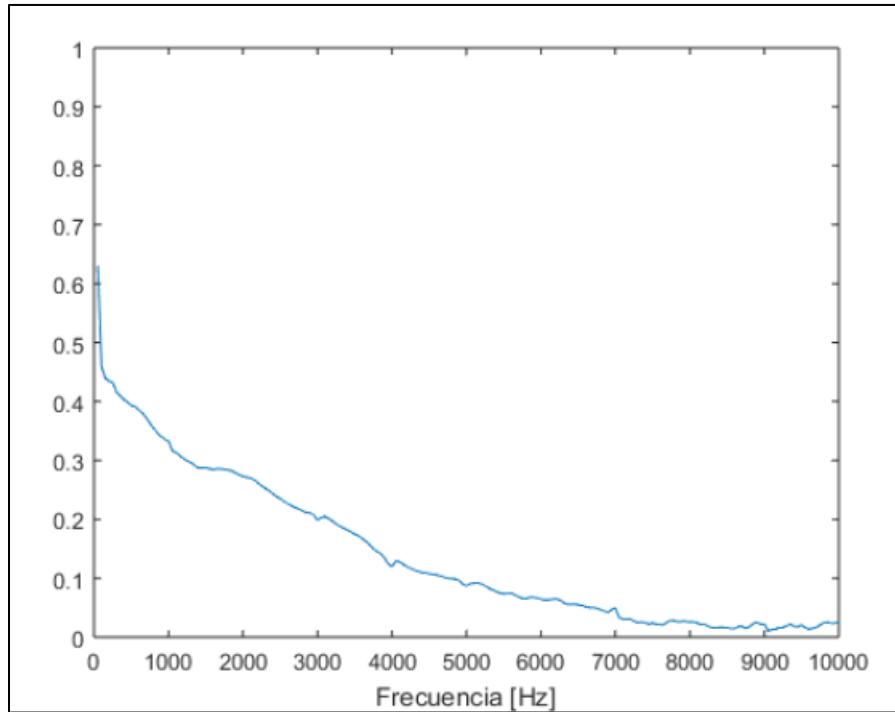


Figura 155.- Coeficientes de transmisión del hule espuma.

Como se puede ver en la [Figura 155](#), el material permite la transmisión para frecuencias bajas mientras que, a frecuencias altas, menor será su amplitud. Según lo visto en la [Figura 4](#), la señal de interés del corazón se encuentra en el rango de 20Hz a 1KHz, por lo cual el hule espuma podría servir como filtro mecánico en el dispositivo.

18 Aplicación Android (Estructura)

La aplicación móvil está estructurada en clases. Se tienen clases tipo “Plugins” que funcionan como herramientas para las clases principales, además se tienen dos clases globales que son importadas por todas las demás, las cuales sirven de control y como variables globales. Por otro lado, se tienen las clases tipo “Thread” que se encargan de procesos en segundo plano. Por último, se encuentran las pantallas principales.

La estructura de la aplicación será explicada en bloques sin entrar en detalles con el código, debido a su complejidad. En el repositorio mencionado de la sección [9.2 Aclaraciones del proyecto](#), se encuentra el código fuente comentado, en caso de requerirlo.

18.1 Hilos de programación

Dentro de los hilos de programación se encuentran ([Figura 156](#)):

- MainActivity
- ThreadData
- ThreadDataProcess
- ThreadHttpServer
- WebSocketHttp

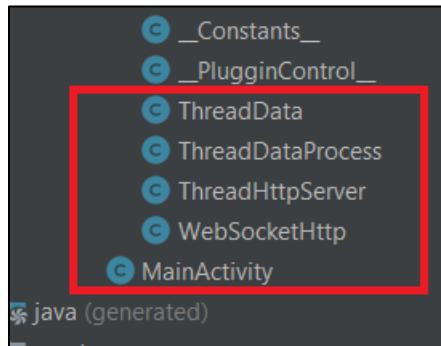


Figura 156.- Hilos de programación

18.1.1 Main Activity

Siendo este el código ligado a la Activity principal de la aplicación. Cuenta con un gran número de métodos para configurar los objetos visuales, iniciar los hilos de programación, pasar y recuperar información entre diferentes instancias de la misma clase y un Handler para recibir mensajes de otros hilos de programación ([Figura 157](#)). Los métodos implementados son:

- `onCreate(savedInstanceState)`: Al ser inicializada la “MainActivity”, el método “`onCreate()`” es llamado. Este método establece las referencias con los objetos visuales de la UI; llama a diferentes funciones para tratar de inicializar los demás hilos de programación en caso de no estar activos; configura algunas referencias encontradas en la sección [18.4 __PluginControl__](#) y [18.5.2 myPlotter](#) con respecto al objeto instanciado de la “MainActivity” y en caso de haber información guardada de una instancia anterior de la “MainActivity” (si la aplicación se ha rotado o cambiado el tema del dispositivo, destruyendo la Activity anterior y generando una nueva) se recuperan la información.
- `onDestroy()`: Al ser destruida la “MainActivity”, el método “`onDestroy()`” es llamado. Este método desreferencia algunas variables de la instancia actual de la “MainActivity” para generar la menor cantidad de basura posible.
- `startThreadData()`: En caso de no estar corriendo el hilo de programación de ThreadData, inicia el proceso.
- `startThreadDataProcess()`: En caso de no estar corriendo el hilo de programación de ThreadDataProcess, inicia el proceso.
- `startThreadHttpServer()`: En caso de no estar corriendo el hilo de programación de ThreadHttpServer, inicia el proceso.
- `startmyTimmer()`: En caso de no estar corriendo el timer, inicia el proceso.

- onSaveInstanceState(outState): Método sobreescrito de la Activity, este método es llamado justo antes de eliminar una Activity. Es usado para guardar la información importante y poder ser recuperada en una nueva instancia.
- mainHandler(getMainLooper()): Handler de la “MainActivity”, al llegar un mensaje ejecuta alguna de las siguientes opciones
 - HANDLER_ICON_BLINK_TIMER: Cambia el icono bluetooth según su estado.
 - HANDLER_CHRONOMETER_TIMER: Actualiza el estado del cronómetro.
 - HANDLER_BUFFERS_BAUDRATE: Actualiza la velocidad de muestreo recibida en el Handler en la pantalla principal.
 - HANDLER_ERROR: En caso de error, despliega un mensaje en la UI.
 - HANDLER_DEBUG: Usado para enviar mensajes de prueba.
- buttonMain(): Método para el botón de Iniciar/Detener/Reiniciar el muestreo. Según el estado del nombre del archivo, conectividad bluetooth y del cronometro, toma una acción.
- buttonMain_Start(filename): Inicia el muestreo y cambia sus parámetros para avisarle al sistema su estado actual. Se manda un mensaje al Handler de “ThreadData” para iniciar el muestro, mientras que “filename” es mandado al Handler de “ThreadDataProcess” para darle un nombre al archivo que se usará para almacenar la información.
- buttonMain_Stop(): Detiene el muestreo y cambia sus parámetros para avisarle al sistema su estado actual. Se manda un mensaje a “ThreadData” para detener el muestreo, además de mandar un mensaje a “ThreadDataProcess” para vaciar el buffer.
- onCreateOptionsMenu(): Despliega el menú de la barra superior.
- onOptionsItemSelected(): Funciona como onClickListener de todos los iconos del menú, según el ícono presionado se toma una acción:
 - info: Despliega un mensaje en la pantalla principal con la información de la aplicación.
 - settings: Muestra la ventana emergente de configuración.
 - bluetooth: Inicia una nueva Activity donde serán desplegados los dispositivos bluetooth previamente emparejados.
 - path: Muestra la ruta de los archivos.
 - Ip_address: Muestra la dirección IP del dispositivo.
- configurationWindow(): Ventana emergente para la configuración de los segundos mostrados en la pantalla.
- setConfiguration(view): Cambia el tamaño del buffer a mostrar según el valor ingresado por el usuario y activa o desactiva la opción de graficar en el dispositivo (La gráfica en red local sigue activa).

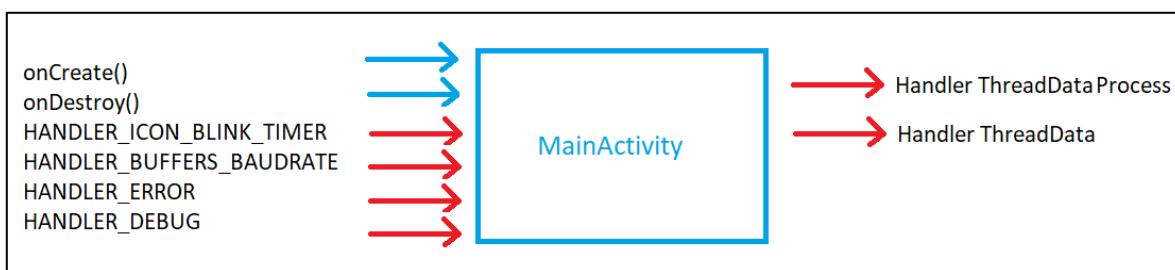


Figura 157.- Diagrama a bloques, Main Activity.

18.1.2 Thread Data

Hilo de programación bluetooth. Este hilo cuenta con métodos para generar un socket Bluetooth, escribir y leer información del socket, así como mandar los paquetes de información en un buffer llamado “bufferRawData” hacia los demás hilos ([Figura 158](#)). Los métodos de la clase son:

- `createBluetoothSocket(device)`: Crea y devuelve el bluetooth socket, “null” en caso contrario.
- `linkBuffersBluetooth(socket)`: Liga los buffers de entrada y salida del socket entrante.
- `writeBluetooth(str)`: Manda el string “str” vía bluetooth.
- `errorToMain_Bluetooth()`: Manda un mensaje de error al “MainActivity”
- `connectBluetooth(macAddress)`: Conecta el bluetooth a la “macAddress” dada.
- `resetBluetooth()`: Reinicia el bluetooth cerrando los sockets y configurando todos sus parámetros como “null”.
- `isBluetoothSocketReady()`: Verifica si todos los parámetros bluetooth están configurados, devuelve “True” si todo es correcto, “False” en caso contrario.
- `isThreadDataRunning()`: Devuelve el estado del hilo de programación, “True” si está corriendo, “False” en caso contrario.
- `threadDataHandler(myLooper())`: Handler de “threadData”, al llegar un mensaje ejecuta alguna de las siguientes opciones
 - `HANDLER_BLUETOOTH_RESET`: Manda el carácter para tener el envío de datos del attiny85 y reinicia la conexión bluetooth.
 - `HANDLER_BLUETOOTH_CONNECT`: Trata de iniciar la conexión bluetooth con la dirección MAC dada.
 - `HANDLER_BLUETOOTH_START`: Carga en “startStopAttiny” el carácter para iniciar el envío de datos del attiny85
 - `HANDLER_BLETOOTH_STOP`: Carga en “startStopAttiny” el carácter para detener el envío de datos del attiny85.
 - `HANDLER_BLUETOOTH_TIMER`: Manda el carácter de control “startStopAttiny85” y en caso de tener datos almacenados en buffer, manda los datos a través de “bufferRawData” a los demás hilos de programación.
- `run()`: Si hay información en el buffer de entrada de datos bluetooth, lee el buffer y almacénalo en un arreglo temporal.



Figura 158.- Diagrama a bloques, Thread Data.

18.1.3 Thread Data Process

La información enviada en el “bufferRawData” por el hilo de programación de la sección [18.1.2 Thread Data](#), es recibida por este proceso. Dentro de este hilo se encuentran los métodos para guardar la información en un archivo de texto, de igual forma cuenta con una sección para procesamiento digital dentro del método “run()”, sin embargo, debido a cuestiones de tiempo, esta sección no pudo ser implementada. La información procesada es enviada en un nuevo buffer hacia los demás hilos llamado “bufferData” ([Figura 159](#)). Cuenta con los siguientes métodos:

- `isExternalStorageWritable()`: Devuelve “True” si el almacenamiento del dispositivo es de lectura y escritura, “False” en caso contrario.
- `checkWriters()`: Verifica y genera la ruta para el almacenamiento del archivo.
- `linkBuffer()`: Crea el archivo para almacenar la información y liga los buffers de entrada y salida con el archivo.
- `destroyBuffers()`: Restablece las variables para el almacenamiento bluetooth.
- `saveOnFile(data)`: Guarda la información “data” dándole formato dentro del archivo.
- `saveOnFileNoFormat(data)`: Guarda la información “data” dentro del archivo.
- `erroToMain_Files(msgText)`: Manda un mensaje de error a la “MainActivity”.
- `flushData()`: Escribe la velocidad de muestreo al final y cierra el archivo.
- `getPath()`: Regresa la ruta de almacenamiento del archivo.
- `isThreadDataProcessRunning()`: Devuelve el estado del hilo de programación, “True” si está corriendo, “False” en caso contrario.
- `threadDataProcessHandler(myLooper())`: Handler de la “threadDataProcess”, al llegar un mensaje ejecuta alguna de las siguientes opciones:
 - `HANDLER_BUFFERS_PATH`: Verifica y genera la ruta para el almacenamiento del archivo.
 - `HANDLER_BUFFERS_LINK`: Verifica y genera la ruta para el almacenamiento, y liga los buffers de entrada y salida al nuevo archivo generado.
 - `HANDLER_BUFFERS_FLUSH`: Escribe la velocidad de muestreo y cierra el archivo.
 - `HANDLER_BUFFERS_CLEAR`: Escribe la velocidad de muestreo, cierra el archivo y restablece las variables de almacenamiento.
- `run()`: Si hay información en el buffer “bufferRawData” toma la información, agrega procesamiento digital (sección no implementada por cuestiones de tiempo), y regresa la información procesada a través de “bufferData”. Además, guarda esta misma información en el archivo de texto y en caso de haber un WebSocket, manda la información a la LAN a través del método `broadcast()`.

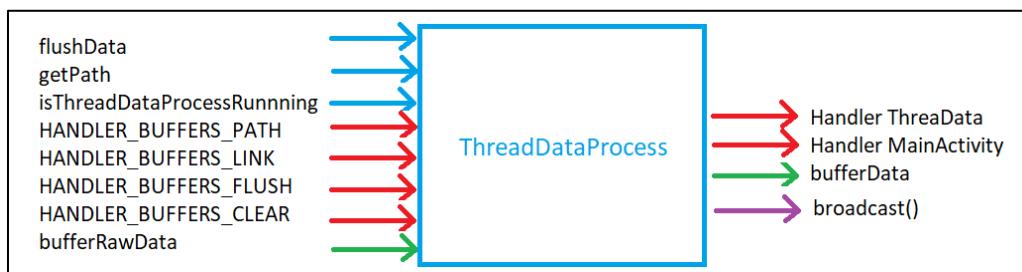


Figura 159.- Diagrama a bloques, ThreadDataProcess

18.1.4 ThreadHttpServer

Hilo de programación para recibir las peticiones HTTP, en caso de haber una petición, manda el archivo “index.html” hacia el cliente ([Figura 160](#)). Cuenta con las siguientes funciones:

- `htmlToString(fileName)`: Lee el archivo HTML cuyo nombre se encuentra en la variable “`fileName`” y convierte el archivo a una cadena de texto legible.
- `toMainThreadHandlerMsg(info)`: Manda un mensaje cuyo contenido se encuentra en la variable “`info`” a la “`MainActivity`”.
- `getIpAddress()`: Regresa la primera dirección IP del dispositivo.
- `isThreadHttpServerRunning()`: Devuelve el estado del hilo de programación, “True” si está corriendo, “False” en caso contrario.
- `run()`: Crea el WebSocket con el puerto asignado en `_Constants_`, en caso de haber una petición HTTP, acepta la conexión y manda el archivo “index.html” al cliente. Dentro del “index.html” se encuentra el código del WebSocket del cliente.

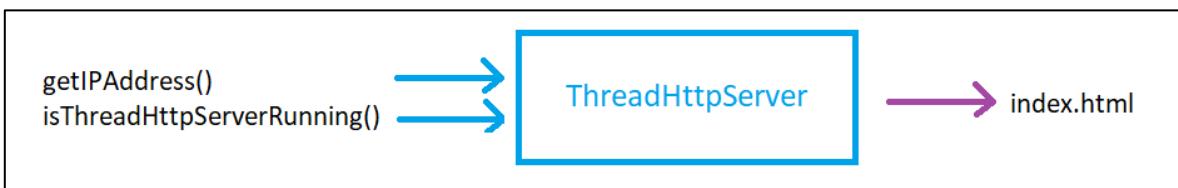


Figura 160.- Diagrama a bloques, ThreadHttpServer

18.2 WebSocketHttp

Esta clase contiene toda la información del WebSocket para la red local. Dentro de esta se encuentran las siguientes funciones:

- `WebSocketHttp(address, port)`: Constructor de la clase, recibe la dirección IP y el puerto para configurar el WebSocket.
- `onStart()`: Método que se ejecuta al iniciar el servidor del WebSocket.
- `onOpen()`: Al recibir una nueva petición, se ejecuta este método.
- `onMessage()`: Método que se ejecuta al recibir un mensaje a través del WebSocket.
- `onClose()`: Al cerrar una conexión con el WebSocket, este método es llamado.
- `onError()`: En caso de error con el WebSocket, se ejecuta este método.

De los métodos previamente mencionados, el único utilizado es el método `onStart()` en el cual se configura un tiempo de espera para aceptar las conexiones con los clientes. Para realizar el envío de la información se usará un método heredado de la clase `WebSocket` siendo este el método “`broadcast`” ([Figura 161](#)).

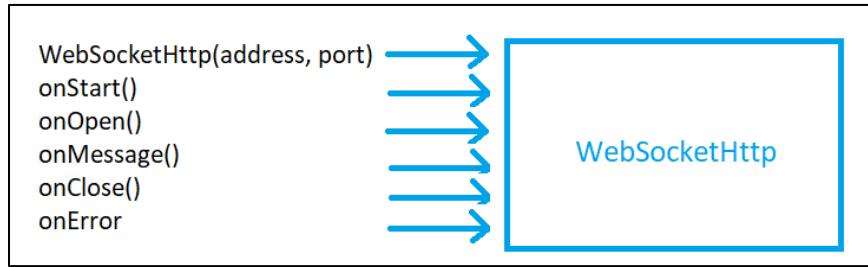


Figura 161.- Diagrama a bloques WebSocketHttp.

18.3 Constants

Esta clase contiene las variables de control globales numéricas o de tipo carácter utilizadas en todas las demás clases. Se tienen variables para discriminar los mensajes mandados a través de Handlers, tiempos de muestreo, estados del cronómetro, puertos para el WebSocket y la petición HTTP, mensajes para iniciar/detener el muestreo de información, entre otras.

18.4 PluginControl

De manera similar a la clase Constants, plugin control contiene variables globales utilizadas por los mismos Plugins o por los diferentes hilos de programación, con la diferencia de que las variables contenidas en este archivo son de tipo objeto.

18.5 Plugins

Dentro de esta carpeta se encuentran algunas herramientas utilizadas por los hilos de programación. Se tiene un plugin para el cronómetro, la gráfica, un temporizador, un mensaje emergente en la pantalla principal y para hacer vibrar el teléfono ([Figura 162](#)).

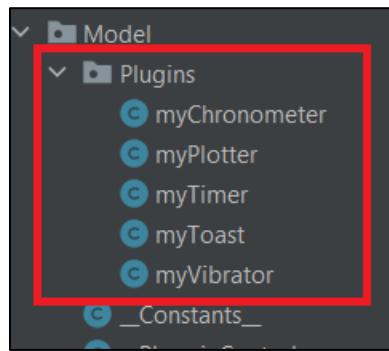


Figura 162.- Plugins.

18.5.1 myChronometer

Este plugin contiene un cronómetro interno y tres métodos de entrada ([Figura 163](#)):

- startChronometer(): Inicia la cuenta del cronómetro.

- `stopChronometer()`: Detiene la cuenta actual del cronómetro.
- `setBaseTime(time)`: Cambia el tiempo base del cronómetro con base en el parámetro pasado de “time”.

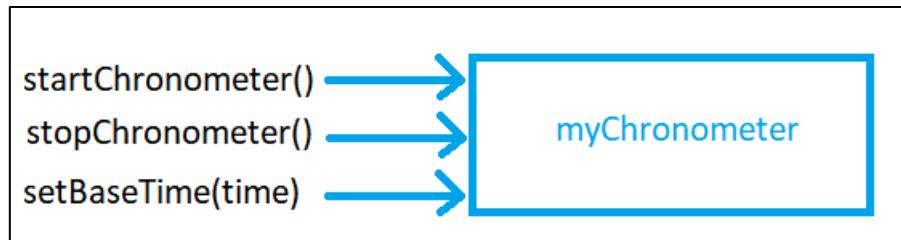


Figura 163.- Diagrama a bloques de myChronometer.

18.5.2 myPlotter

Internamente cuenta con las variables necesarias para crear una nueva gráfica. Los métodos de la clase son los siguientes ([Figura 164](#)):

- `sendBaudRateToMain(baudRate)`: Manda el BaudRate hacia la “MainActivity” a través de un mensaje en su Handler.
- `setPlotter(plot)`: Liga la gráfica de la clase con la gráfica mandada como argumento “plot”.
- `setPlotterAuxLayout(auxLayout)`: Por cuestiones estéticas, se tiene un layout que acompaña a la gráfica, para ligar este layout, con una vista en la “MainActivity”, se hace uso de esta función.
- `setPlotterConfiguration()`: Configura la vista de la gráfica, definiendo valores máximos, mínimos, colores, etiquetas y aspecto de la gráfica.
- `cleanPlotter()`: Limpia la gráfica.
- `getBaudRate()`: Retorna la velocidad de muestreo actual.
- `resetBaudRate()`: Reinicia las variables relacionadas con la velocidad de muestreo.
- `addEntry()`: Toma el valor almacenado en el “bufferData” y lo agrega como una nueva entrada en la gráfica, además calcula la velocidad de muestreo.

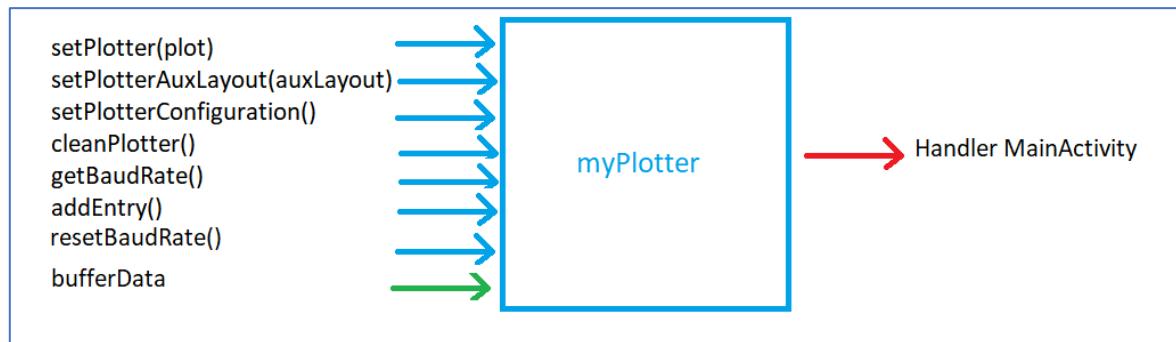


Figura 164.- Diagrama a bloques de myPlotter

18.5.3 myTimmer

Esta clase contiene un contador que manda mensajes a diferentes hilos cada segundo. El contador es utilizado para sincronizar la aplicación. Este contador fue implementado por cuestiones de

tiempo, se recomienda utilizar una metodología diferente con funciones asíncronas. Cuenta con los siguientes métodos:

- `isMyTimmerRunning()`: Regresa el estado del temporizador, “True” si el temporizador está activo, “False” en caso contrario.
- `startTimer()`: Inicia la cuenta del temporizador.
- `stopTimer()`: Detiene la cuenta.

Dentro de la función `startTimer()` se encuentra el Timer Task del temporizador, este manda mensajes al Handler de la UI y al Handler del “ThreadData” ([Figura 165](#)).



Figura 165.- Diagrama a bloques de myTimmer.

18.5.4 myToast

Esta clase es utilizada para mandar mensajes tipo Toast a la Main UI, cuenta con un método público en el cual se carga el mensaje a desplegar, este método utiliza la clase `myVibrator` para vibrar el dispositivo al mandar el Toast ([Figura 166](#)).

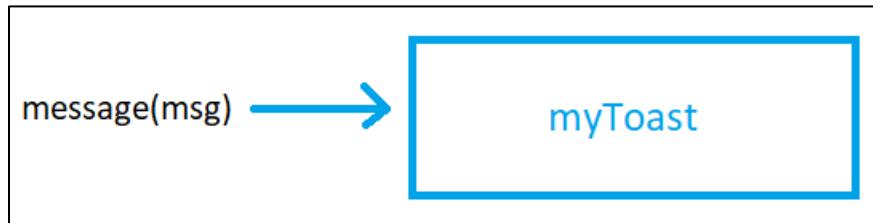


Figura 166.- Diagrama a bloques myToast.

18.5.5 myVibrator

Esta clase cuenta con un solo método público el cual se encarga de hacer vibrar el dispositivo por cierta cantidad de tiempo, el tiempo de vibración puede ser configurado a través de la variable “time”. La vibración se realiza en `time*ms`.

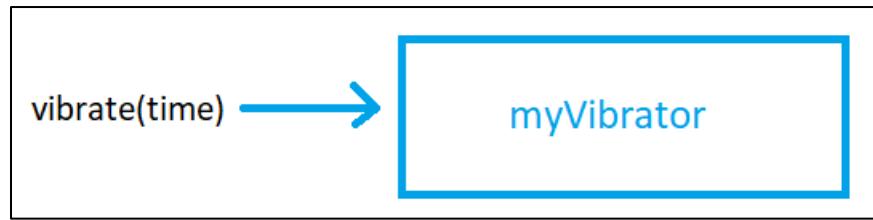
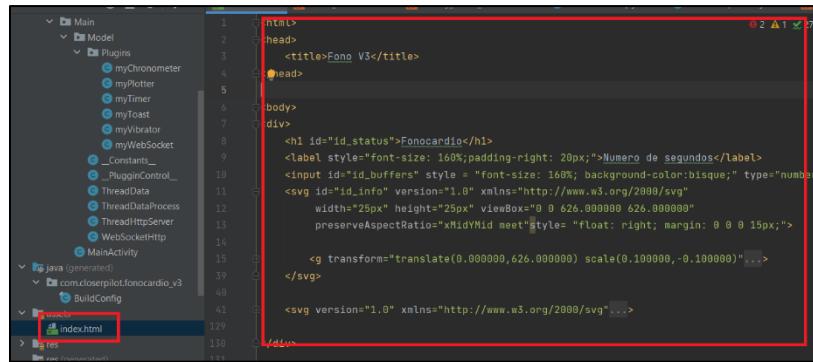


Figura 167.- Diagrama a bloques de myVibrator

18.6 index.html

Al mandar una petición HTTP al servidor, se responde con el archivo “index.html” el cual contiene toda la configuración para desplegar la gráfica en el navegador, así como guardar la información en un archivo de texto. De igual forma, internamente cuenta con el código del WebSocket del cliente para conectarse al servidor y recibir los paquetes de información en tiempo real ([Figura 168](#)).



```
<html>
  <head>
    <title>Fono V3</title>
  </head>
  <body>
    <div>
      <h1 id="id_status">Fonocardio</h1>
      <label style="font-size: 16px; padding-right: 20px;">Número de segundos</label>
      <input id="id_buffers" style="font-size: 16px; background-color: bisque;" type="number">
      <svg id="id_info" version="1.0" xmlns="http://www.w3.org/2000/svg"
        width="250px" height="250px" viewBox="0 0 626.000000 626.000000"
        preserveAspectRatio="xMidYMid meet" style="float: right; margin: 0 0 0 15px;">
        <g transform="translate(0.000000,626.000000) scale(0.100000,-0.100000)" ...>
      </svg>
    </div>
  </body>
</html>
```

Figura 168.- index.html

18.7 Layout

Dentro de esta carpeta se encuentran todas las pantallas y ventanas utilizadas en la aplicación ([Figura 169](#)).

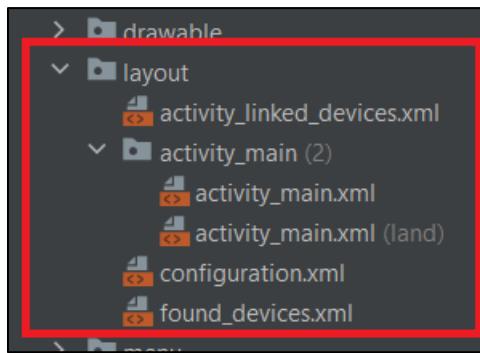


Figura 169.- Layouts

18.7.1 activity_main

Siendo la estructura de la pantalla principal cuando se encuentra en vista vertical. Cuenta con el indicador de Conectado/Desconectado, el cronómetro, un recuadro para ingresar el nombre del archivo, el botón de Inicio/Detener para el muestreo, así como la gráfica para los datos ([Figura 170](#)). Este archivo guarda las referencias hacia los elementos de la pantalla que serán utilizados por las clases en Java.

```

<?xml version="1.0" encoding="UTF-8"?
<RelativeLayout
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:id="@+id/id_relative"
    xmlns:android="http://schemas.android.com/apk/res/android"

<androidx.constraintlayout.widget.ConstraintLayout
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".Main.MainActivity"

<androidx.constraintlayout.widget.ConstraintLayout
    android:id="@+id/constraintLayout"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent">

```

Figura 170.- *activity_main.xml (Portrait)*.

18.7.2 *activity_main (Land)*

Siendo la estructura de la pantalla principal cuando se encuentra en vista horizontal. Cuenta con los mismo elementos y referencias vistas en la sección [18.7.1 *activity_main*](#) con la única diferencia de redistribuir sus elementos para adaptarse a la forma de la pantalla ([Figura 171](#)).

```

<?xml version="1.0" encoding="UTF-8"?
<RelativeLayout
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:id="@+id/id_relative"
    xmlns:android="http://schemas.android.com/apk/res/android"

<androidx.constraintlayout.widget.ConstraintLayout
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".Main.MainActivity"

<androidx.constraintlayout.widget.ConstraintLayout
    android:id="@+id/constraintLayout"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent">

```

Figura 171.- *activity_main.xml (Land)*.

18.7.3 *configuration*

La aplicación cuenta con una ventana emergente la cual sirve como configuración para asignar un valor al número de segundos a desplegar en la gráfica, y activar o desactivar la gráfica en el dispositivo. Esta ventana emergente tiene su propio layout siendo el archivo configuration.xml, cuenta con un recuadrado de selección numérico y un botón de aceptar ([Figura 172](#)).

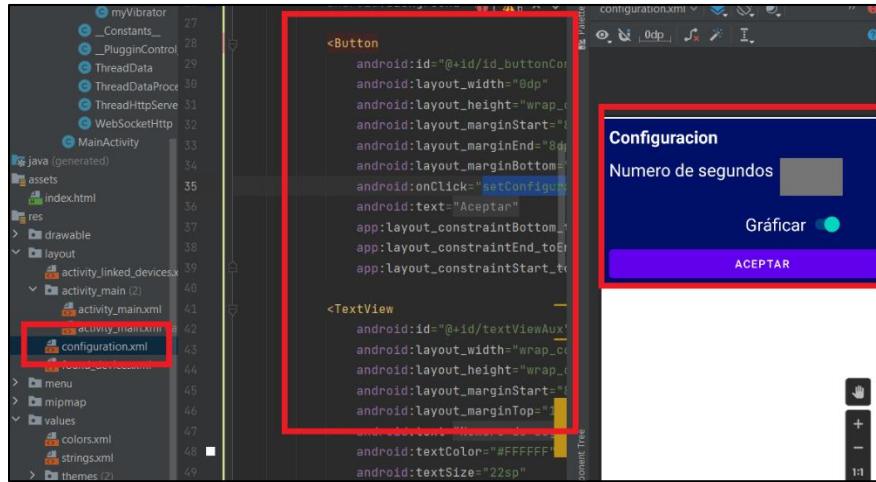


Figura 172.- configuration.xml

18.7.4 found_devices

Este layout se encuentra vacío en primera instancia, sirve para ser llenado con la información de los dispositivos vinculados a través de bluetooth. Cuenta con un TextView que abarca toda la pantalla ([Figura 173](#)).

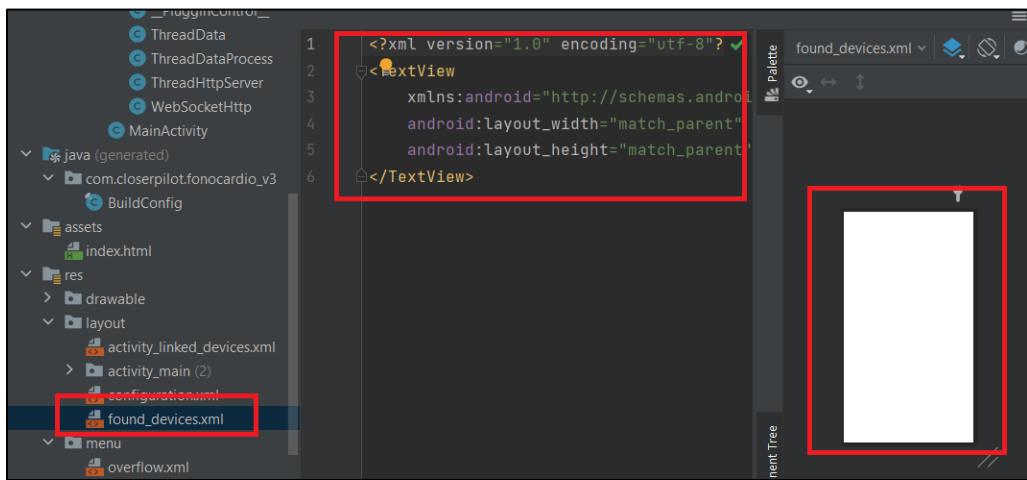


Figura 173.- founs_devices.xml

18.7.5 activity_linked_devices

Esta layout es utilizada como el padre del xml de la sección [18.7.4 found_devices](#), sirve como una nueva Activity cuando se ejecuta el servicio bluetooth ([Figura 174](#)). Al ser ejecutada se abre una nueva pantalla donde aparecen todos los dispositivos previamente emparejados en layout de found_devices.xml.

The screenshot shows the Android Studio interface with the XML code for `activity_linked_devices.xml` highlighted. The code defines a `ConstraintLayout` with a `ListView` and various styling attributes. To the right, the preview pane shows a white screen, indicating that the layout is currently empty or not yet populated.

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".LinkedDevices.LinkedDevicesActivity">

    <ListView
        android:id="@+id/PairedDevices"
        android:layout_width="0dp"
        android:layout_height="0dp"
        android:padding="20dp"
        android:paddingStart="20dp"
        android:paddingEnd="20dp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent" />

```

Figura 174.- `activity_linked_devices.xml`

18.7.6 overflow

Este archivo contiene las vistas de la barra de menú de la parte superior de la aplicación. Dentro de este menú se encuentran los iconos para desplegar la ruta de almacenamiento de los archivos, la dirección IP del dispositivo, un cuadro de diálogo con información de la aplicación, una ventana emergente para cambiar el número de segundos a mostrar en la gráfica, así como el ícono bluetooth para abrir la Activity de la sección [18.7.5 `activity_linked_devices`](#).

The screenshot shows the Android Studio interface with the XML code for `overflow.xml` highlighted. The code defines a `Menu` with two items: one for folder paths and another for device IP address. To the right, the preview pane shows a purple-themed action bar with icons for file, network, and Bluetooth, along with a dropdown menu containing "Configuracion" and "Informacion".

```
<menu xmlns:app="http://schemas.android.com/apk/res-auto"
      xmlns:android="http://schemas.android.com/apk/res/android">

    <item
        android:id="@+id/id_path"
        android:checkable="false"
        android:icon="@drawable/baseline_folder_24"
        android:title="Ubicacion de las carpetas"
        android:visible="true"
        app:showAsAction="always" />

    <item
        android:id="@+id/id_ip_address"
        android:checkable="false"
        android:icon="@drawable/baseline_wifi_24"
        android:title="Direccion IP de dispositivo"
        android:visible="true" />

```

Figura 175.- `overflow.xml`

18.8 Drawable

Dentro de la carpeta Drawable se encuentran las imágenes vectorizadas usadas en el proyecto. ([Figura 176](#)).

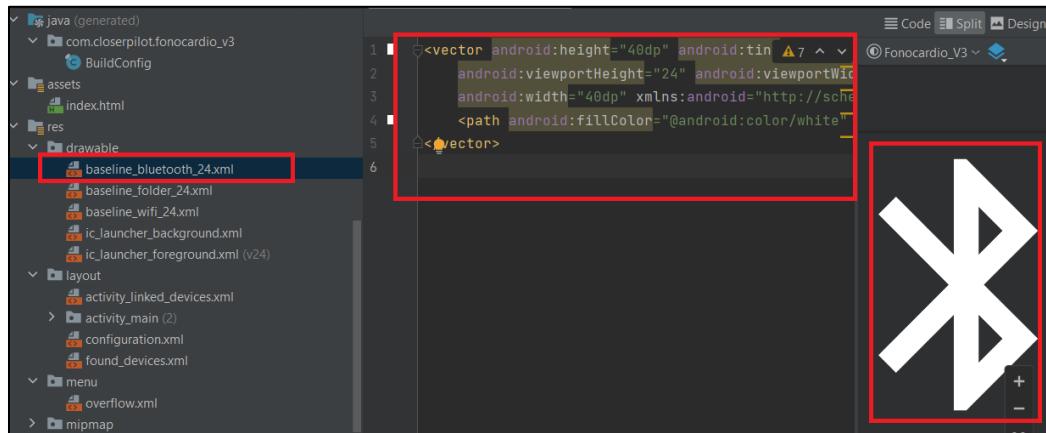


Figura 176.- Ejemplo de ícono Bluetooth, Drawable

18.9 Values

Dentro de la carpeta Values se encuentran dos archivos xml llamados “strings.xml” y “colors.xml” estos archivos contienen valores que pueden ser usados tanto en los layouts como en las clases de java.

18.9.1 Strings

Dentro de este archivo se escriben los nombres de los objetos a utilizar en los layouts. ([Figura 177](#)).

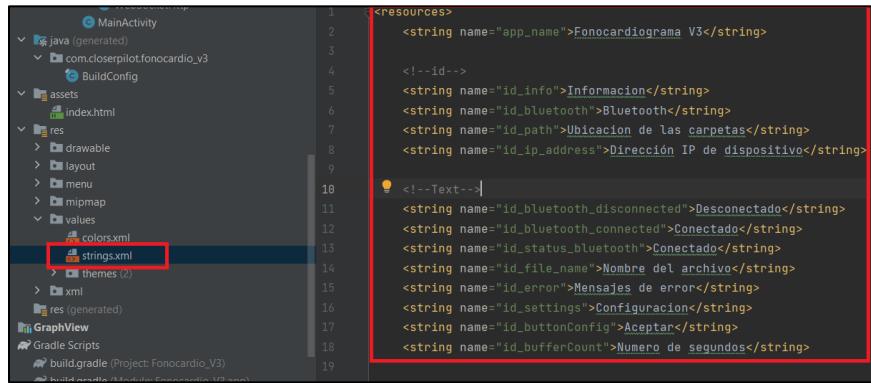


Figura 177.- Strings.xml

18.9.2 Colors

Dentro de este archivo se encuentran las constantes de los colores utilizados en el proyecto([Figura 178](#)).

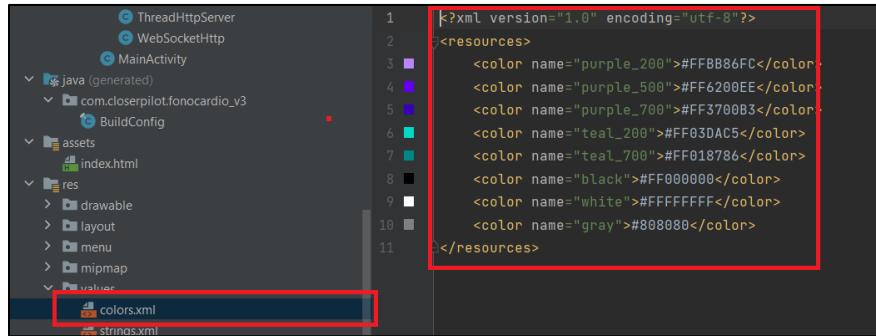


Figura 178.- Colors.xml

18.10 Gradle Scripts

Dentro del proyecto se encuentra la sección de “Gradle Scripts”, estos agregan las dependencias externas requeridas. La aplicación móvil cuenta con dos librerías externas ([Figura 179](#)):

- GraphView
- Java-WebSocket:1.5.3



Figura 179.- Dependencias externas en Gradle.

18.10.1 GraphView

Librería para graficar creada por jjoe64 [22], es una librería creada para Android que permite graficar diagramas de manera sencilla. Se eligió esta librería para graficar los datos del corazón en la pantalla del celular, sin embargo, la implementación hecha en la aplicación móvil presenta algunos problemas de rendimiento. Esta librería fue descargada desde GitHub y se encuentran dentro de los archivos de la aplicación en la carpeta “repositoriosExternos” ([Figura 180](#)).

Nombre	Fecha de modificación	Tipo	Tamaño
.gradle	25/06/2023 07:26 p. m.	Carpeta de archivos	
.idea	25/06/2023 07:26 p. m.	Carpeta de archivos	
app	25/06/2023 07:26 p. m.	Carpeta de archivos	
gradle	25/06/2023 07:26 p. m.	Carpeta de archivos	
repositoriosExternos	25/06/2023 07:26 p. m.	Carpeta de archivos	
.ignore	19/06/2023 09:38 p. m.	Documento de texto	1 KB
build.gradle	19/06/2023 09:38 p. m.	Archivo GRADLE	1 KB
gradle.properties	19/06/2023 09:38 p. m.	Archivo PROPERTI...	2 KB
gradlew	19/06/2023 09:38 p. m.	Archivo	6 KB
gradlew.bat	19/06/2023 09:38 p. m.	Archivo por lotes ...	3 KB
local.properties	22/06/2023 11:07 p. m.	Archivo PROPERTI...	1 KB
settings.gradle	19/06/2023 09:38 p. m.	Archivo GRADLE	1 KB

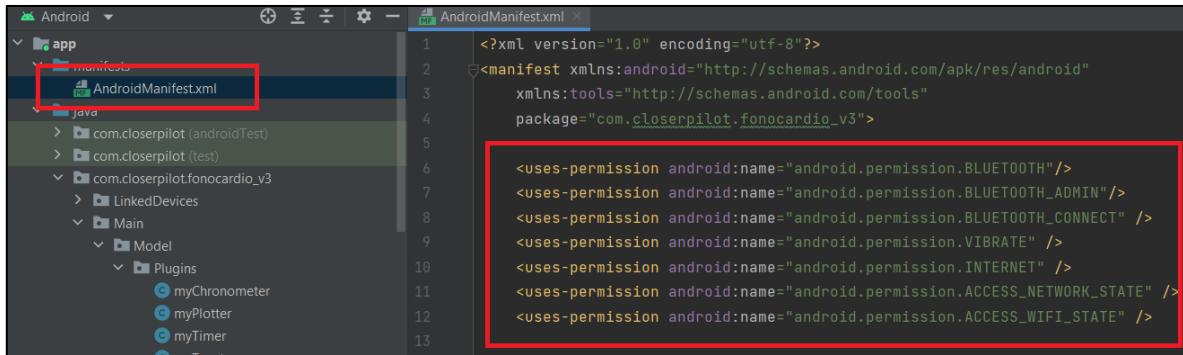
Figura 180.- Ruta de la librería GraphView

18.10.2 Java-WebSocket:1.5.3

Librería para la creación de WebSockets creada por Nathan Rajlich y Marcel Prestel de org.java-websocket [23] siendo su versión más reciente hasta el momento, esta librería esta utilizada para construir el WebSocket usado en la conectividad LAN.

18.11 Android Manifest

Debido al funcionamiento de la aplicación, es necesario agregar permisos de uso. Estos permisos se encuentran dentro del Manifest. Para la aplicación móvil, se usan los permisos para vibrar el dispositivo, así como permisos de conectividad Bluetooth, Wifi e Internet ([Figura 181](#)).



```

<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    package="com.closerpilot.fonocardio_v3">

    <uses-permission android:name="android.permission.BLUETOOTH"/>
    <uses-permission android:name="android.permission.BLUETOOTH_ADMIN"/>
    <uses-permission android:name="android.permission.BLUETOOTH_CONNECT" />
    <uses-permission android:name="android.permission.VIBRATE" />
    <uses-permission android:name="android.permission.INTERNET" />
    <uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />
    <uses-permission android:name="android.permission.ACCESS_WIFI_STATE" />

```

Figura 181.- Android Manifest

19 Aplicación Android

19.1 Funcionamiento de la aplicación móvil

19.1.1 Sincronización del circuito del fonocardiograma.

Siendo la primera vez que se usa la aplicación móvil con el circuito del fonocardiograma, es necesario emparejar el circuito con el celular. Para ello se va a “Opciones Bluetooth” y se trata de emparejar un nuevo dispositivo, según la configuración de la placa debería aparecer de la siguiente manera ([Figura 182](#)):



Figura 182.- Vinculación del dispositivo por primera vez

Una vez vinculado el dispositivo, se abre la aplicación del celular ([Figura 183](#)):



Figura 183.- Secciones de la aplicación móvil.

Inicialmente, la aplicación se encuentra desconectada del circuito del fonocardiograma, para conectarlo es necesario dar clic dentro de la sección de “Barra de menú” en el ícono Bluetooth.

Al dar clic, se abre la siguiente ventana (*Figura 184*):

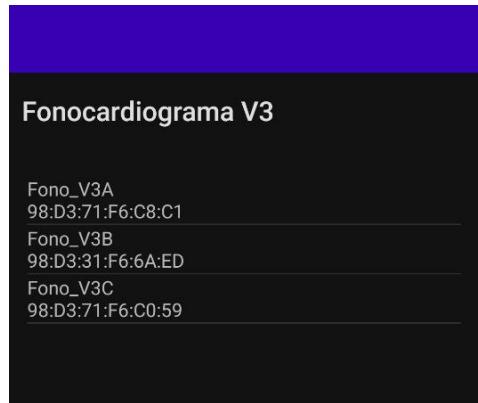


Figura 184.- Opciones de emparejamiento bluetooth

Para conectarse, es necesario seleccionar el dispositivo del fonocardiograma y esperar que la conexión quede establecida.

Nota: *En caso de no ver el dispositivo del fonocardiograma, se debe de tratar de volver a emparejar el celular con el circuito del fonocardiograma.*

En caso de haber una conexión exitosa, el ícono de estado parpadeará de color verde y el letrero cambiará a “Conectado” (*Figura 185*).

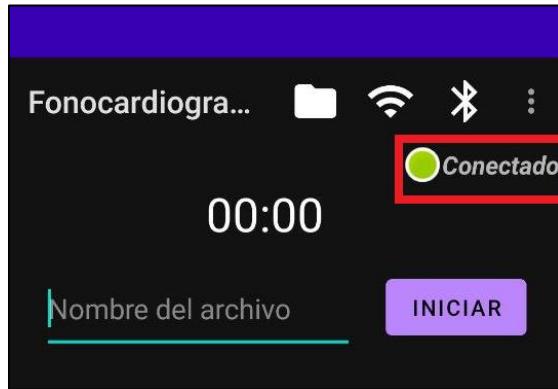


Figura 185.- Conexión bluetooth exitosa.

En caso de haber algún problema con la conexión bluetooth, aparecerá el siguiente cuadro de diálogo (*Figura 186*):

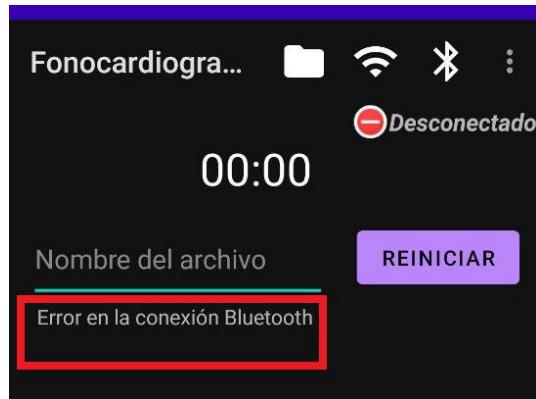


Figura 186.- Error en la conexión bluetooth.

19.1.2 Creación del archivo

Una vez vinculado el celular con el circuito del fonocardiograma, el siguiente paso es nombrar al archivo. En el cuadro de dialogo de la [Figura 187](#), se agrega el nombre que tendrá el archivo.

Nota: Si existe un archivo con el mismo nombre, este será SOBREESCRITO.

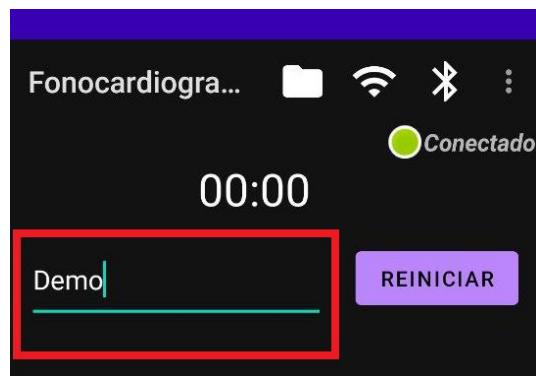


Figura 187.- Insertar el nombre del archivo

En caso de haber un error al tratar de crear el archivo, se mostrará un texto en la parte inferior del nombre del archivo ([Figura 188](#))

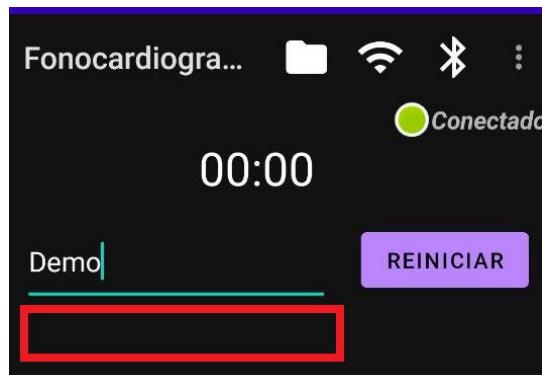


Figura 188.- Ubicación del cuadro de error en la creación del archivo.

Para conocer la ruta donde será almacenado el archivo, se da clic en el ícono en forma de carpeta, de esta forma aparecerá la ruta del archivo debajo de su nombre ([Figura 189](#)).

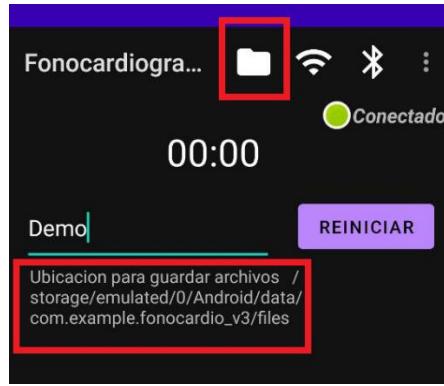


Figura 189.- Ruta para guardar el archivo.

19.1.3 Muestreo de la señal

Una vez vinculado el bluetooth y elegido el nombre del archivo, se da clic en el botón "Iniciar/Reiniciar", con esto se manda la señal de inicio del muestreo al circuito físico y se empieza a capturar la información.

Al iniciar, el cronómetro empezará a correr y el botón de "Iniciar/Reiniciar", cambiará a "Detener". De igual forma, en la parte inferior de la aplicación se empezarán a graficar los datos del fonocardiograma ([Figura 190](#)).

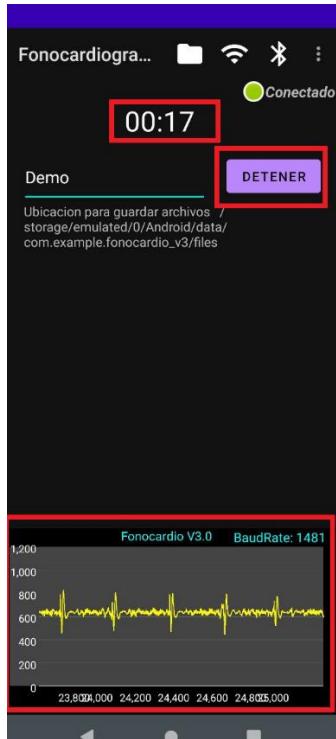


Figura 190.- Muestreo de la señal.

Para detener el muestreo, basta con dar clic en el botón de “Detener”. Al detener la cuenta, la información es guardada en el archivo creado y se espera la instrucción para “Reiniciar”.

Nota: *Al parar la cuenta y volverla a correr, se trata de una nueva instancia, es decir, LOS NUEVOS DATOS SOBREESCRIBIRÁN A LOS VIEJOS. La opción de “Detener” está pensada para cerrar los datos del paciente actual y esperar los datos de uno DIFERENTE al dar clic en “Reiniciar”.*

19.1.4 Configuración

Es posible configurar algunas funciones de la aplicación. Para acceder a la configuración basta con dar clic en los tres puntos del lado superior derecho y en “Configuración” (*Figura 191*).

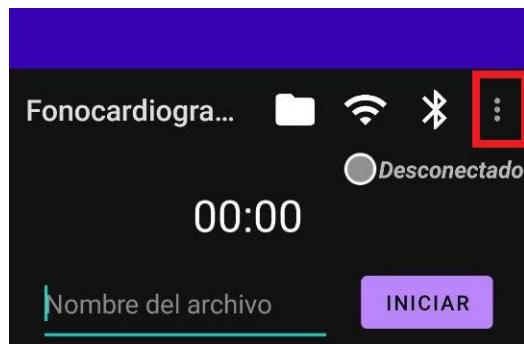


Figura 191.- Más opciones, configuración.

Al dar clic, se abre la siguiente ventana emergente (*Figura 192*):



Figura 192.- Configuración.

Dentro de esta ventana, es posible cambiar el número de segundos a mostrar en la gráfica, así como habilitar o deshabilitar la gráfica del celular. Una vez configuradas las opciones se da clic en “Aceptar” para aplicar los cambios.

Nota: La opción de “Graficar” fue implementada debido a que la aplicación móvil presenta problemas de rendimiento con respecto a la gráfica, una mejor versión de la gráfica puede ser vista a través del navegador dentro de la red local.

19.1.5 Red local

Para acceder a la aplicación dentro de la red local es necesario saber la IP del celular, para esto se da clic en el ícono wifi de la barra de menú. La dirección IP será desplegada debajo del nombre del archivo ([Figura 193](#)).



Figura 193.- Dirección IP del dispositivo.

19.2 Funcionamiento de la aplicación en red local

Para acceder a la aplicación dentro de la red local es necesario seguir los pasos de la sección [19.1 Funcionamiento de la aplicación móvil](#). Una vez configurada la conexión bluetooth, el nombre del archivo y la dirección IP del dispositivo, se abre cualquier navegador **DENTRO DE LA RED LOCAL**, y se escribe la IP junto con el puerto mostrado en la aplicación (por ejemplo 192.168.68.100:8888).

Al tener una conexión exitosa, se muestra la siguiente pantalla ([Figura 194](#)):



Figura 194.- Aplicación de la red local.

Nota: La gráfica de color negro, aparece únicamente si el muestreo se ha iniciado, caso contrario, al conectarse por primera vez sin datos recibidos, la gráfica no aparecerá, ésta aparece automáticamente una vez que inicie el muestreo.

Nota: En caso de que la página tarde mucho en cargar, se recomienda actualizarla desde el navegador la veces que sea necesario. A veces el WebSocket presenta problemas de conexión y es necesario actualizarlo manualmente.

19.2.1 Muestreo de la señal

La aplicación en red local graficará de manera automática los datos mandados por el teléfono, si el teléfono no recibe datos del circuito del fonocardiograma, el navegador no tendrá datos para actualizar. El teléfono funciona como controlador para iniciar y detener el ingreso de datos en el navegador.

19.2.2 Configuración

En la parte superior del navegador se encuentra una sección para configurar algunos parámetros de la aplicación:

- Número de segundos: Se elige cuantos segundos se quiere mostrar en la gráfica del navegador (de manera similar que, en el celular, sección [19.1.4 Configuración](#))
- Nombre del archivo: Se elige un nombre para el archivo a descarga con la información actual.
- Limpiar: El funcionamiento de este botón es el de borrar toda la información guarda hasta el momento.

19.2.3 Creación del archivo

De manera similar a la aplicación en el celular, es posible crear un archivo con toda la información recibida hasta el momento y descargarlo desde el navegador hacia el dispositivo conectado en la red local. Para ello existe un cuadro de dialogo que permite cambiar el nombre del archivo ([Figura 195](#)). Una vez cambiado, se da clic en el botón de “Guardar” y comienza la descarga del archivo.

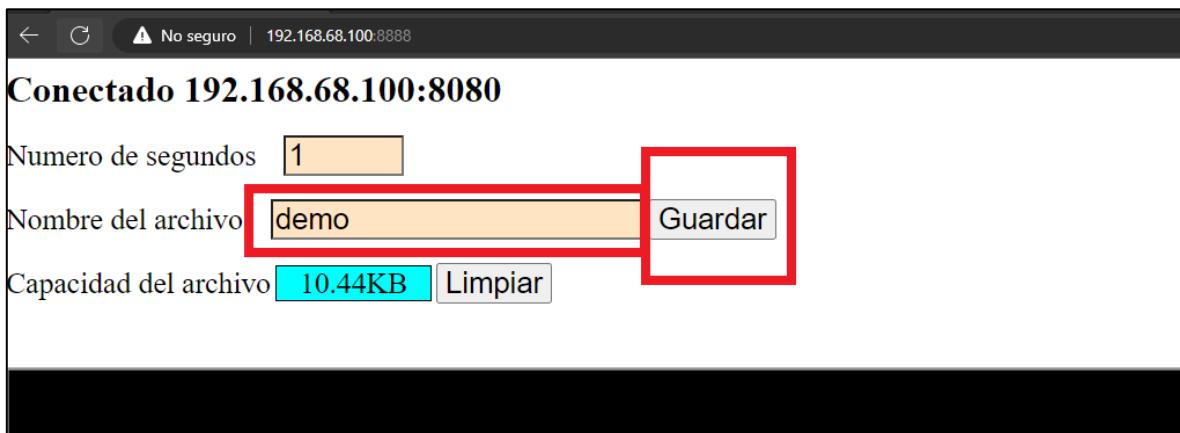


Figura 195.- Nombre del archivo a descargar.

Para borrar toda la información guardada hasta el momento y poder generar un archivo nuevo, se da clic en el botón “Limpiar”. El tamaño aproximado del archivo puede verse en el recuadro de color azul ([Figura 196](#)).

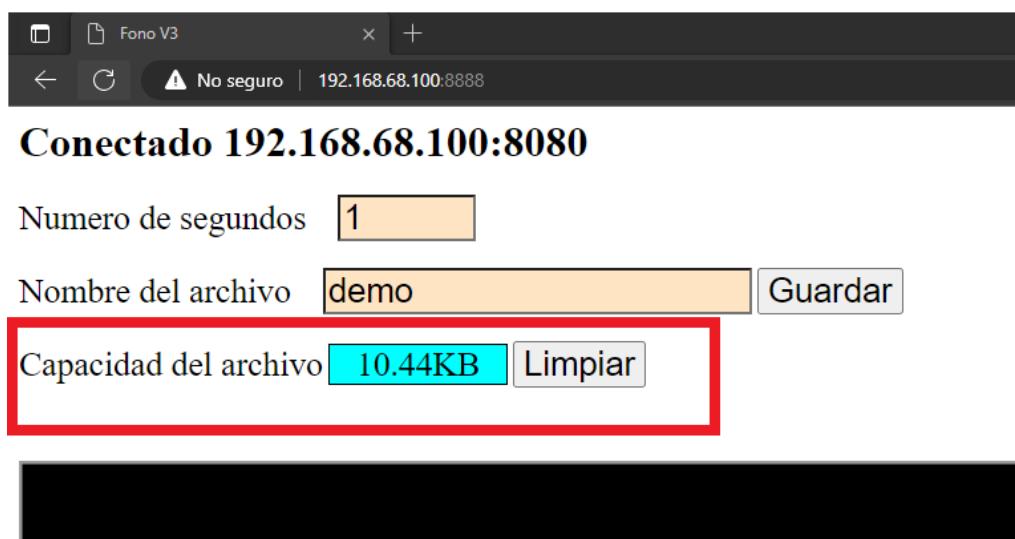


Figura 196.- Limpiar

19.3 Partes de la aplicación móvil

La pantalla principal de la aplicación móvil puede ser dividida en las siguientes secciones ([Figura 197](#)):

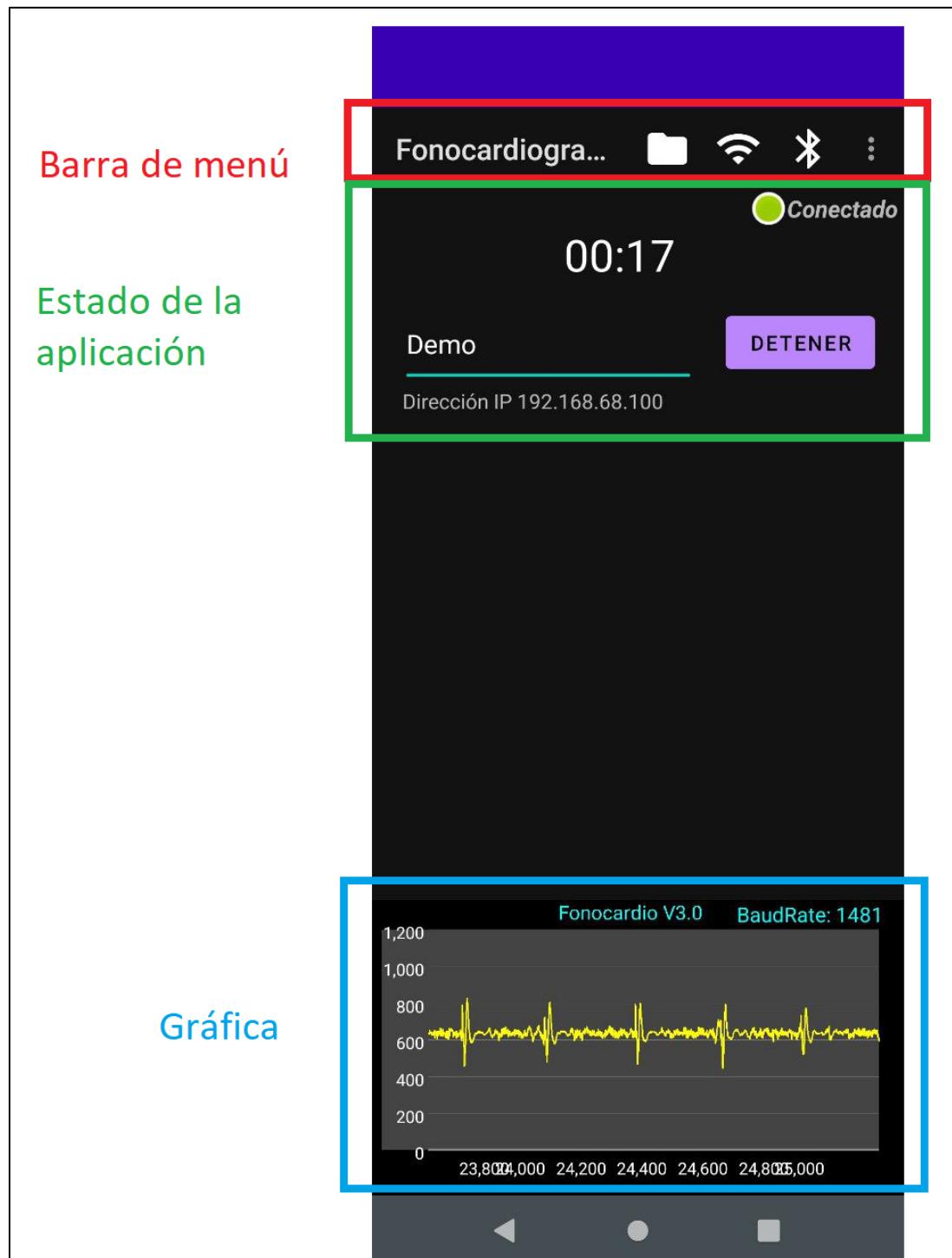


Figura 197.- Secciones de la aplicación móvil.

19.3.1 Barra de menú

Dentro de esta barra se encuentran los siguientes íconos (*Figura 198*) :

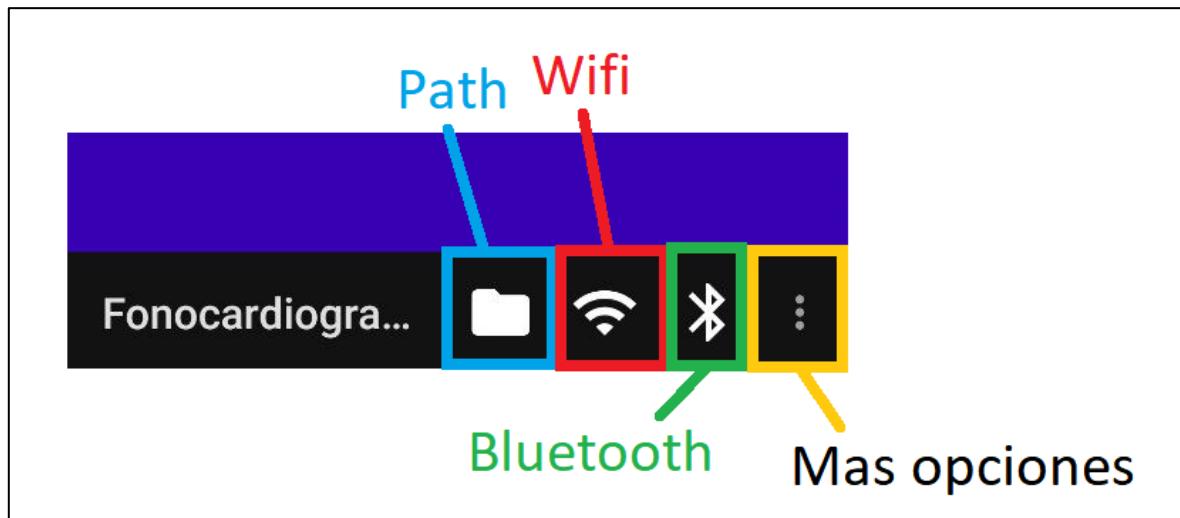


Figura 198.- Barra de menú.

- Path: Botón para mostrar el directorio donde se guardará el archivo con la información recopilada del fonocardiograma.
- Wifi: Botón para mostrar la dirección IP del celular, esta dirección será usada para conectarse a través del navegador al celular dentro de la red local.
- Bluetooth: Botón para seleccionar el dispositivo a conectar (circuito del fonocardiograma) con la aplicación del celular.
- Mas opciones: Al presionar el botón, aparecen dos opciones más (*Figura 199*):
 - Configuración: Menú desplegable para configurar ciertos parámetros de la aplicación
 - Información: Información sobre la aplicación.

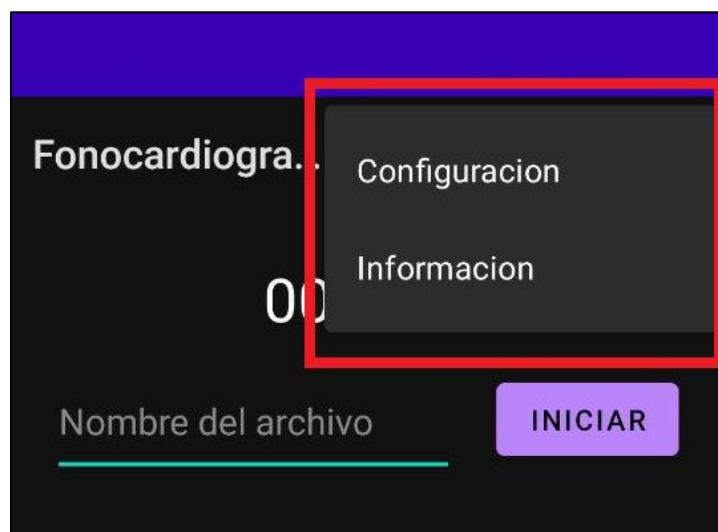


Figura 199.- Mas opciones

Al dar clic en “Configuración” se abre la siguiente ventana (*Figura 200*). Dentro de esta ventana se encuentra la opción de “Número de segundos” el cual permite modificar el número de segundos a visualizar en la gráfica del celular. Otra opción es la de habilitar o deshabilitar la gráfica.



Figura 200.- Configuración, aplicación móvil.

Por último, la opción “Información” del menú desplegable, muestra un mensaje con la información de la aplicación (*Figura 201*).

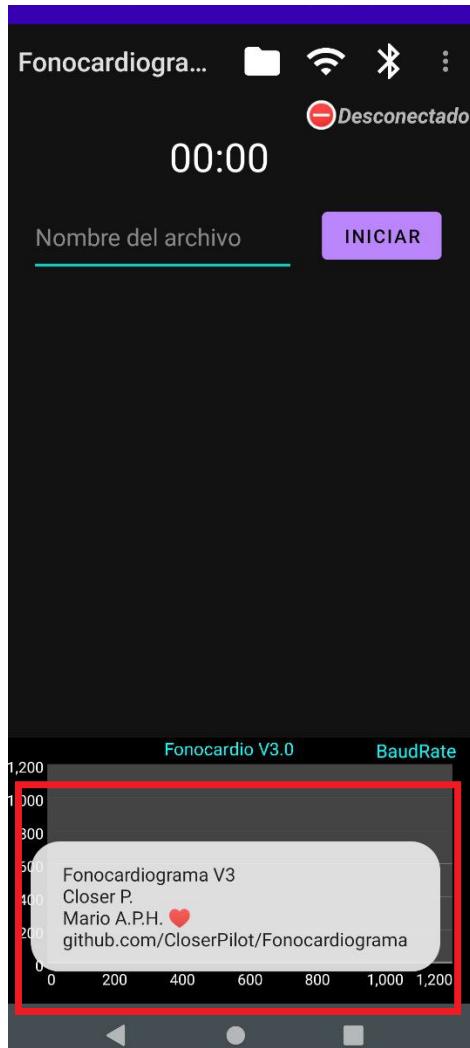


Figura 201.- Información, aplicación móvil.

19.3.2 Estado de la aplicación

Dentro de esta sección se encuentran los siguientes elementos ([Figura 202](#)):



Figura 202.- Estado de la aplicación.

- Indicador de estado: Este ícono indica el estado actual del bluetooth, en caso de que la conexión se exitosa, el ícono parpadeará de color verde junto con el mensaje “Conectado”, en caso contrario, el ícono parpadeará de color rojo con la leyenda “Desconectado”.
- Botón principal: Botón de control para el muestreo de la aplicación. Este botón es el que controla el inicio y final del muestreo.
- Cronómetro: Indicador del tiempo total de muestreo.
- Nombre del archivo: Entrada para el nombre del archivo de los datos a guardar.
- Mensaje: Mensajes importantes a comunicar con el usuario.

19.3.3 Gráfica

Dentro de esta sección se distinguen dos elementos principales (*Figura 203*):

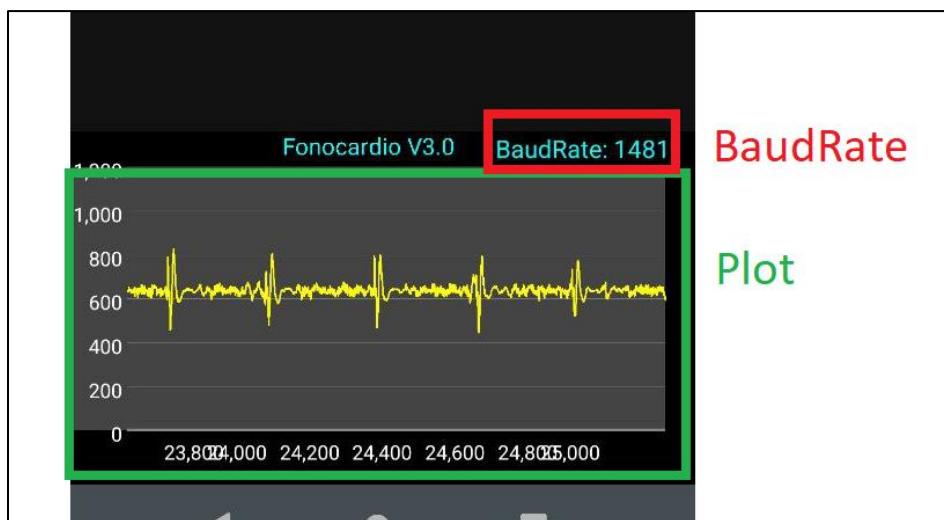


Figura 203.- Gráfica, aplicación móvil.

- Plot: Gráfica con los datos en tiempo real del fonocardiograma.
- BaudRate: Velocidad de muestreo de los datos.

19.4 Partes de la aplicación en red local

La aplicación en red local puede ser dividida en las siguientes partes (*Figura 204*):

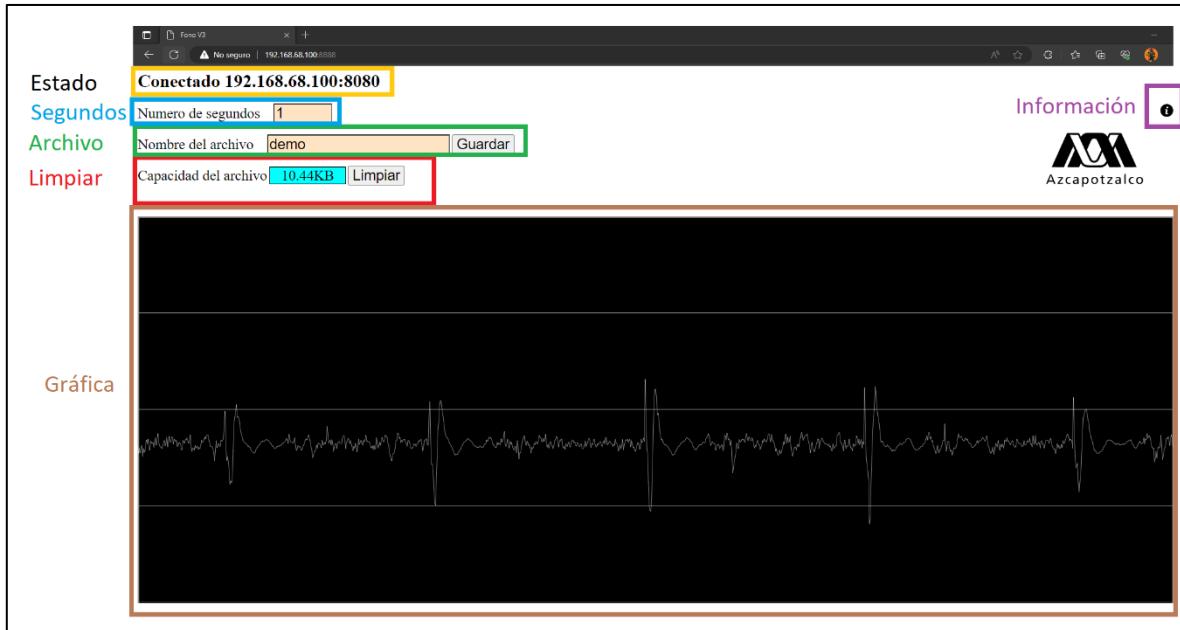


Figura 204.- Partes de la aplicación en red local.

- Estado: Muestra el estado actual de la aplicación en red local.
- Segundos: Variable para controlar el número de segundos a mostrar en la gráfica.
- Archivo: Nombre del archivo para descargar los datos desde el navegador.
- Limpiar: En caso de requerirlo, se pueden limpiar todos los datos almacenados hasta el momento y comenzar a guardar datos nuevos.
- Gráfica: Gráfica de los datos en tiempo real enviados desde el celular.
- Información: Abre una ventana emergente con información sobre la aplicación.

20 Resultados

Por último, para realizar las pruebas de funcionamiento, el circuito del fonocardiograma fue montado sobre un sujetador apuntando al pecho.

En la *Error! No se encuentra el origen de la referencia.*, se puede observar cómo va montado el circuito sobre el paciente, de color azul se encuentra la campana del fonocardiograma apuntando al pecho, de color rojo el circuito y de color verde la batería de 9V.

20.1 Módulo A: Amplificador de instrumentación AD623

A continuación, se puede ver el prototipo del módulo A ensamblado ([Figura 205](#) y [Figura 206](#)).

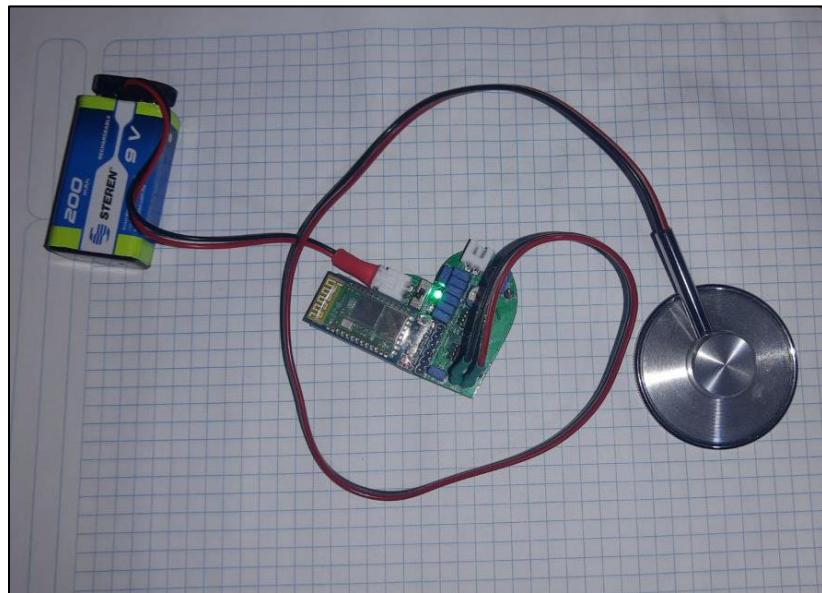


Figura 205.- Módulo A, vista frontal.



Figura 206.- Módulo A, vista trasera.

A continuación, se muestra la señal cardiaca a través de la aplicación del celular ([Figura 207](#)).

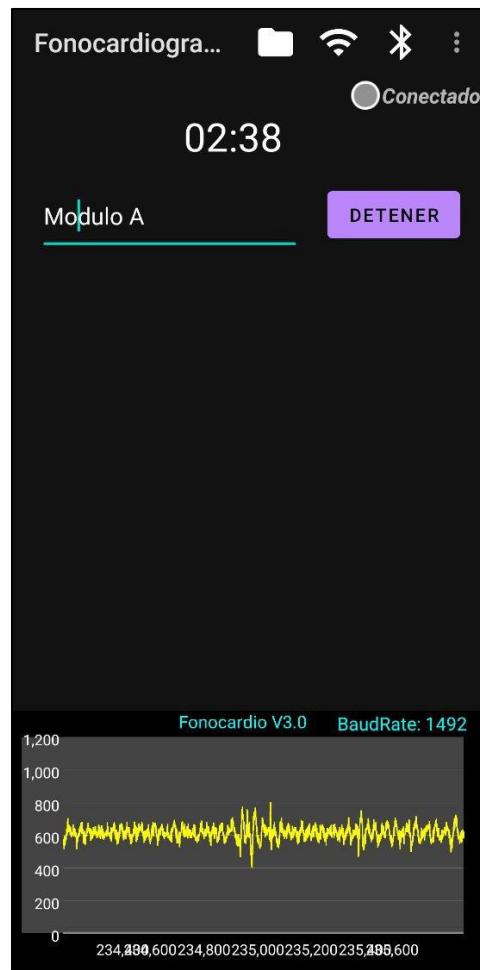


Figura 207.- Módulo A, visto desde el celular.

Por último, se presentan los resultados vistos en la red local ([Figura 208](#)).

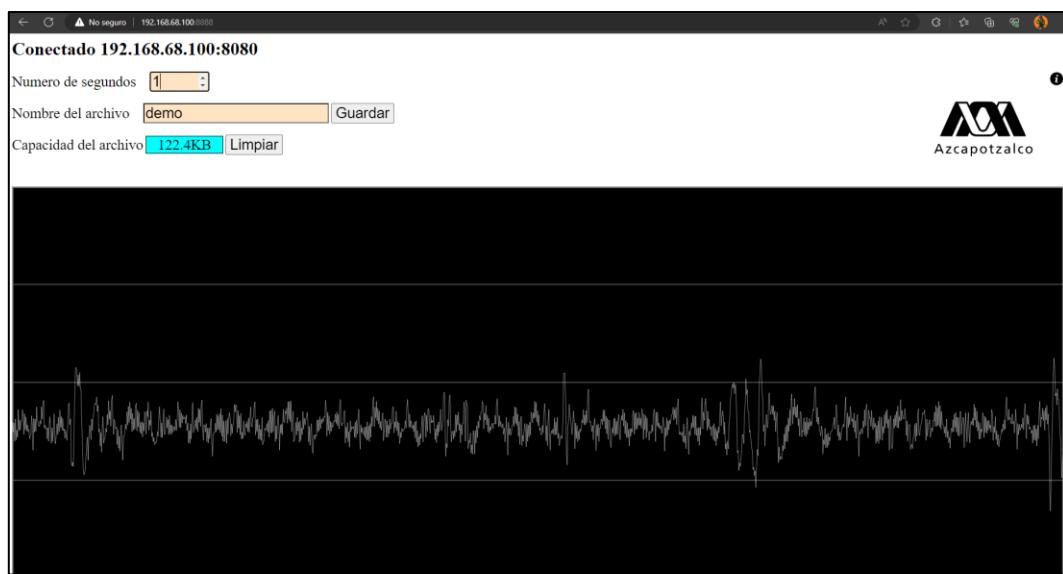


Figura 208.- Módulo A, visto desde el navegador.

20.2 Módulo B: Control de ganancia automático MAX9814

A continuación, se puede ver el prototipo del módulo B ensamblado ([Figura 205](#) y [Figura 206](#)).

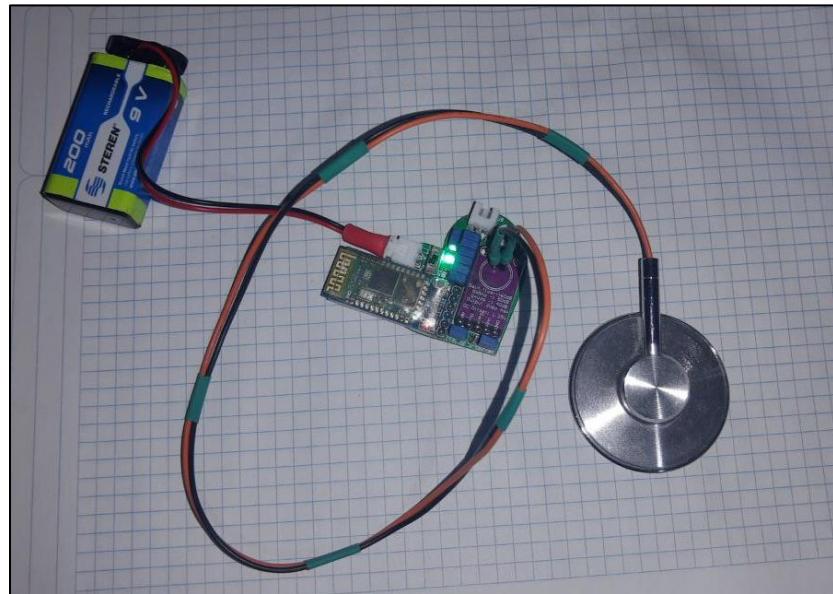


Figura 209.- Módulo B, vista frontal.

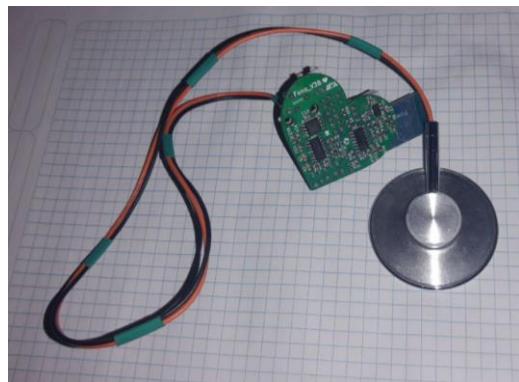


Figura 210.- Módulo B, vista trasera.

A continuación, se muestra la señal cardiaca a través de la aplicación del celular ([Figura 211](#)).

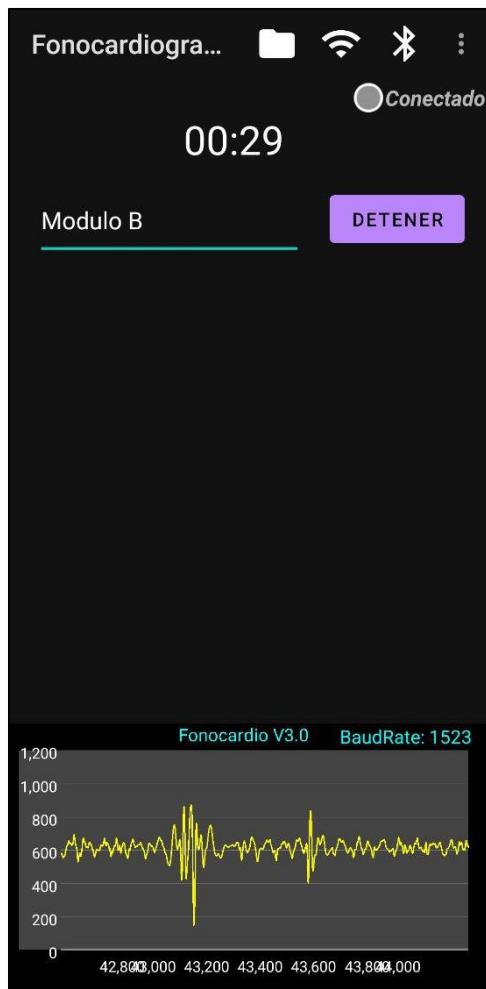


Figura 211.- Módulo B, visto desde el celular.

Por último, se presentan los resultados vistos en la red local ([Figura 212](#)).

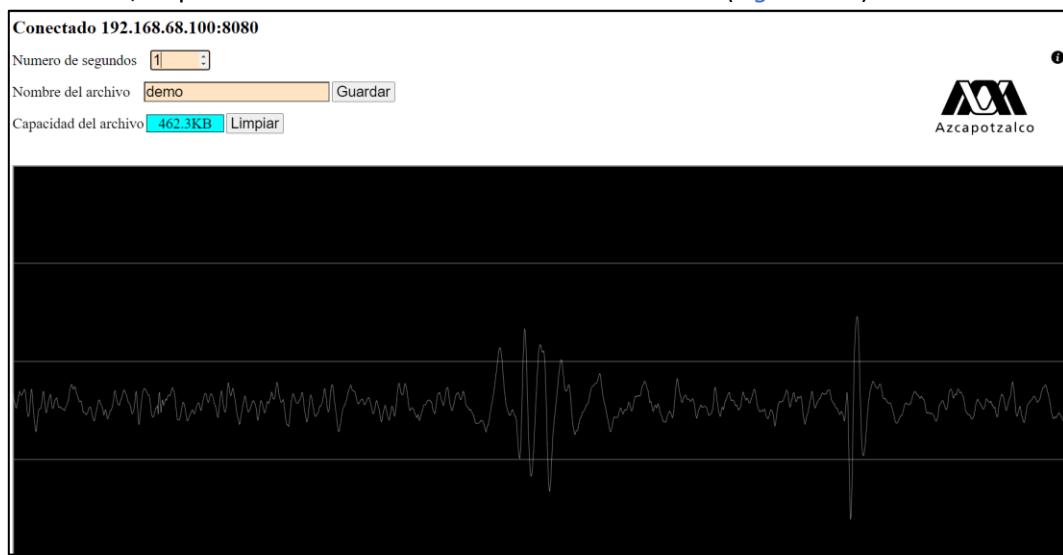


Figura 212.- Módulo B, visto desde el navegador.

20.3 Módulo C: Control de ganancia automático MAX9814 y módulo Bluetooth EGBT-045MS

A continuación, se puede ver el prototipo del módulo C ensamblado ([Figura 205](#) y [Figura 206](#)):

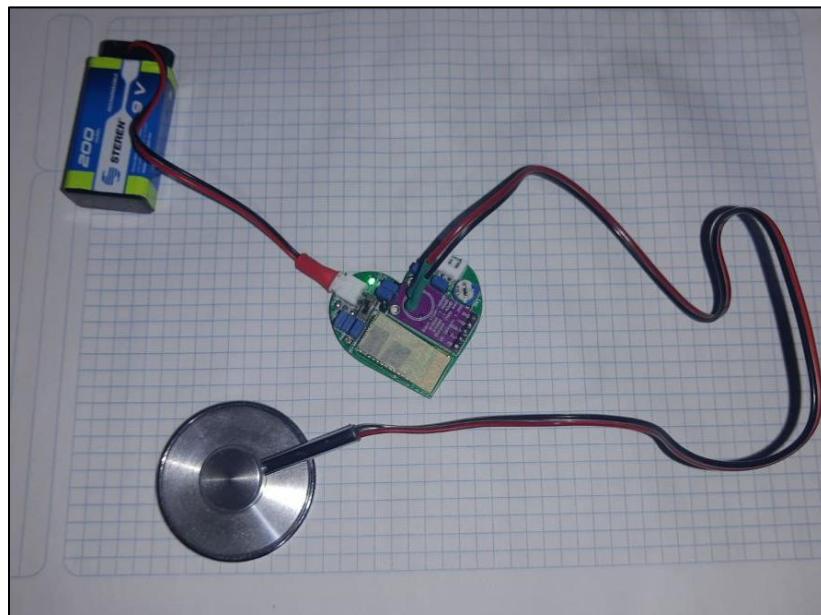


Figura 213.- Módulo C, vista frontal.

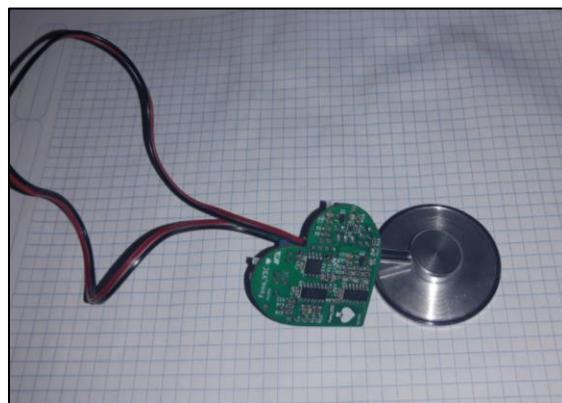


Figura 214.- Módulo C, vista trasera.

A continuación, se muestra la señal cardiaca a través de la aplicación del celular ([Figura 215](#)).

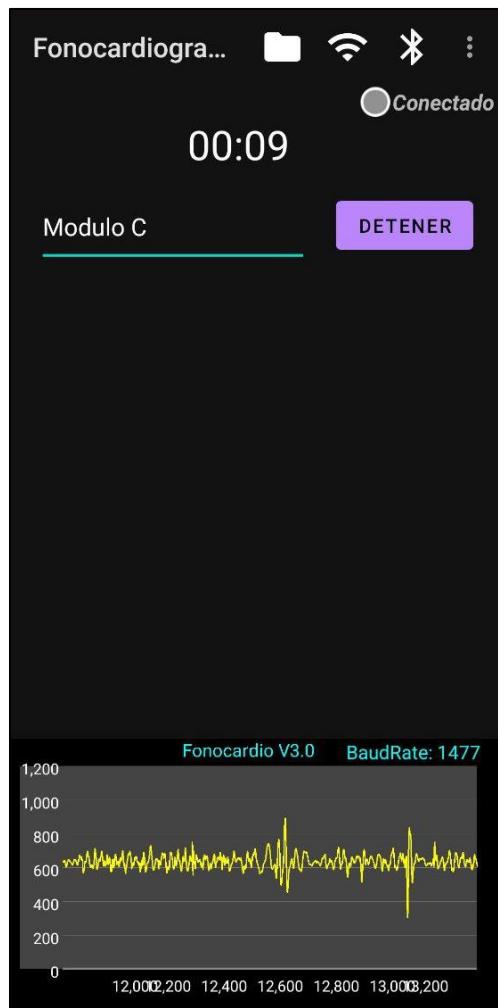


Figura 215.- Módulo C, visto desde el celular.

Por último, se presentan los resultados vistos en la red local ([Figura 216](#)).



Figura 216.- Módulo C, visto desde el navegador.

21 Análisis de resultados

21.1 Módulo A: Amplificador de instrumentación

AD623

Según los resultados obtenidos en la sección [20.1 Módulo A: Amplificador de instrumentación AD623](#), la señal recibida en la aplicación del celular y en el navegador, presentan una amplitud cercana a la señal de ruido, esto es debido a su amplificación lineal. Aumentar la amplificación no ayudaría ya que el mismo ruido sería amplificado, de igual forma la implementación de filtros analógicos no sería viable ya que la señal ruidosa se encuentra en una frecuencia similar a la señal cardiaca. Para tener una señal más limpia es necesario implementar una metodología diferente.

21.2 Módulo B: Control de ganancia automático

MAX9814

Según los resultados obtenidos en la sección [20.2 Módulo B: Control de ganancia automático MAX9814](#), la señal recibida en la aplicación del celular y en el navegador, presentan cierta mejora con respecto al [20.1 Módulo A: Amplificador de instrumentación AD623](#). Al implementar el control de ganancia automático con el MAX9814, se logra atenuar las señales ruidosas luego de cierto umbral de voltaje.

21.3 Módulo C: Control de ganancia automático

MAX9814 y módulo Bluetooth EGBT-045MS

Según los resultados obtenidos en la sección [20.3 Módulo C: Control de ganancia automático MAX9814 y módulo Bluetooth EGBT-045MS](#), los resultados obtenidos son similares a los vistos en la sección [20.2 Módulo B: Control de ganancia automático MAX9814](#), lo cual era de esperarse ya que ambas versiones del circuito comparten la misma etapa de acondicionamiento .Su única diferencia radica en cuestiones de desempeño, teniendo reguladores de voltaje más eficientes y un diodo de protección.

22 Conclusiones

22.1 Circuitos de acondicionamiento

Se procuró dar un acondicionamiento a la señal cardiaca lo más simple y eficiente posible, evitando el uso de sobre ingeniería. Las placas mostradas en el reporte son el resultado de varias pruebas y diseños fallidos, al tener un circuito que fuera considerado lo suficientemente bueno, este era implementado en una placa física y reportado.

Gracias a las modificaciones implementadas en las tres placas del circuito, se logró tener una señal más limpia con respecto al ruido. El MAX9814 es de gran ayuda para amplificar los voltajes por debajo de cierto umbral y atenuar aquellos que lo sobrepasan, ayudando a reducir la influencia de picos de voltaje generados por golpes accidentales. De igual forma, no solamente importa el funcionamiento del dispositivo, sino que la estética está involucrada, es debido a esto que se realizó el “Módulo C”, teniendo una placa más limpia y agregando pequeñas mejoras en los reguladores de voltaje y un diodo de protección.

Los componentes usados en el dispositivo no son componentes especializados, parte del ruido presente en la señal de salida puede ser debido a esto, se recomendaría realizar nuevas pruebas con base en el circuito presentado en el reporte, cambiando a componentes de grado médico.

Las modificaciones al MAX9814 fueron hechas con base en pruebas que se realizaron sobre el dispositivo (valores prácticos). No se pudo experimentar con más valores debido a que las pruebas se realizaron sobre la misma placa de Adafruit y al ser componentes SMD, había que soldarlos y desoldarlos. Se recomendaría realizar pruebas más exhaustivas para corroborar que los valores propuestos son los más eficientes.

Los filtros encontrados en el circuito de acondicionamiento son un pasa bandas de segundo orden, se podría implementar una nueva etapa de procesamiento de la señal con base en los resultados obtenidos (el ruido presente en la señal de salida de las tres placas y los picos de voltaje inducidos por golpes externos).

El programa cargado en el attiny85 es muy sencillo, únicamente se lee el valor de la señal con el ADC interno y se envía por puerto serial hacia el módulo Bluetooth. Al tener un microcontrolador es posible cargar un programa más complejo que permita procesamiento digital o añadir más funciones de comunicación con el celular.

La estructura del arnés con el hule espuma logra cierta estabilidad al montar el dispositivo, sin embargo, el hecho de ejercer presión sobre la campana del estetoscopio para recibir la señal sonora de los latidos cardiacos genera cierta molestia al ser un material rígido. Para futuras pruebas se recomienda elegir un mejor micrófono con base en sus parámetros acústicos y de ser posible, cambiar la campana del estetoscopio por otro material más endeble que permita amplificar los latidos cardiacos.

22.2 Aplicación móvil

La aplicación del celular logra recibir, graficar y guardar los datos del fonocardiograma, sin embargo, presenta ciertos errores en su diseño. Durante el desarrollo del proyecto, se estuvieron corrigiendo errores que se detectaban conforme a la marcha, pero seguramente hay más errores no detectados sin resolver. Un gran fallo conforme a la aplicación fue el incluir la librería externa para graficar los datos en el celular; el cómo está codificada dentro la aplicación la vuelve muy lenta, dejando el dispositivo congelado por ciertos lapsos de tiempo o haciendo que sus respuestas tengan cierto retraso, por cuestiones de tiempo no se pudo corregir este problema, pero se agregó la opción de deshabilitar la gráfica para evitarlo. Se recomienda deshabilitar la gráfica y ver los datos en la red local en caso de no poder visualizarlos en la aplicación del celular. Para corregir este problema se podría cambiar la implementación para graficar con la librería externa dentro del código o implementar un canvas propio.

Una etapa que no se pudo implementar fue el procesamiento digital. Se trató de dejar todo el código listo para poder implementarlo, sin embargo, esta etapa no pudo ser trabajada debido a que se tuvo que dar prioridad a otras etapas del proyecto. La implementación se encuentra comentada dentro de la clase “ThreadDataProcess”. De igual manera, una posible mejora para la aplicación es implementar procesamiento digital más avanzado; una posible idea es agregar una “Transformada de Wavelet”, sin embargo, debido a falta de tiempo y conocimiento, no pudo ser implementada. De igual forma, se tenía pensado calcular la frecuencia cardiaca y mostrarla en la pantalla del celular, pero por cuestiones de tiempo, no se implementó.

Otra posible mejora es el implementar un buscador de archivos para elegir la carpeta donde serán creados los archivos con la información del paciente. De igual forma se podría agregar una base de datos para tener un registro médico.

Teniendo una pequeña computadora como lo es el celular, se podría agregar otra nueva funcionalidad que lea la información del paciente y por medio de inteligencia artificial, discriminar si algún paciente puede tener un padecimiento y notificarle al médico.

22.3 Aplicación en red local

Teniendo la opción para visualizar la información en cualquier dispositivo dentro de la red local, permite un gran dinamismo para el usuario. La implementación mostrada en el reporte un muy sencilla, únicamente se recibe una petición HTTP y se manda al frontEnd una respuesta con un WebSocket apuntando hacia el backEnd. No se implementaron servidores complejos como “Apache” ya que lo que se trataba de demostrar era (con un ejemplo simple) que es posible integrar IoT a los dispositivos.

Teniendo un dispositivo IoT la información puede ser tratada con mayor libertad en cualquier dispositivo, no tiene que ser un celular o solo una computadora, cualquier dispositivo que tenga navegador puede acceder a la información.

La implementación mostrada contaba únicamente con una gráfica para la información entrante, un botón para guardar la información en un archivo de texto y descargarlo en el ordenador, un botón para limpiar y otro para modificar la cantidad de segundos a visualizar en la gráfica. Teniendo un sistema más complejo de parte del servidor, es posible integrar una mayor funcionalidad a esta

sección. Se podría integrar una consola que permita elegir algún tipo de procesamiento digital de la información, indicadores para la frecuencia cardiaca, una interfaz intuitiva para el usuario, conexión a base de datos, inteligencia artificial, poder reproducir la señal de manera sonora en el dispositivo, conectar alguna aplicación externa de mensajería (por ejemplo WhatsApp) y mandar notificaciones al celular, correos, alertas, recordatorios, o incluso conseguir un dominio y conectar la aplicación a Internet.

23 Bibliografía

- [1] DeltaPyramax, «eKuore One Wireless User Manual,» [En línea]. Available: <https://www.deltapyramax.com/pdf/pdf-k/eKuore%20One%20Wireless%20User%20Manual.pdf>. [Último acceso: 2022].
- [2] A. Guyton, «Tratado de fisiología médica,» Mc Graw Hill, 2000, pp. 290-304.
- [3] Widex, «El rango auditivo humano ¿Cuánto puedes oír?,» Widex, 2019. [En línea]. Available: <https://www.widex.cl/es-cl/blog/rango-auditivo-humano#:~:text=Mientras%20que%20a%20partir%20de%200%20dB>. [Último acceso: 2022].
- [4] G. Mejía, *Dispositivo para la auscultación de sonidos corporales*, Ciudad de México: Universidad Autónoma Metropolitana, Proyecto fin de carrera, Dpto. de Ingeniería Física, 2022.
- [5] J. Osorio, C. Luida y G. Felipe, «Diseño y construcción de un fonocardiógrafo digital con visualización en LabVIEW,» 2007. [En línea]. Available: https://www.researchgate.net/publication/306011663_Diseno_y_construccion_de_un_fonocardiografo_digital_con_visualizacion_en_LabVIEW. [Último acceso: 2022].
- [6] B. Cruz, L. Rizalde, L. Ayala y Catzín, «Diseño e implementación de un sistema para el monitoreo de los sonidos cardíacos,» Universidad Autónoma de Yucatán, 2011. [En línea]. Available: <http://www.amca.mx/memorias/amca2011/papers/amca2011-82-tema2.pdf>. [Último acceso: 2022].
- [7] P. Palacio y D. Barrios, «Diseño e implementación de un fonocardiógrafo digital para el laboratorio de bioelectrónica,» Institución Universitaria Pascual Bravo, 2013. [En línea]. Available: http://repositorio.pascualbravo.edu.co:8080/jspui/bitstream/pascualbravo/315/1/Rep_UPB_Tec_Electro_Fonocardi%C3%B3grafo.pdf. [Último acceso: 2022].
- [8] J. Martí y E. Berjano, «Diseño y desarrollo de un prototipo de estetoscopio digital,» Universidad Politécnica de Valencia, 2020. [En línea]. Available: <https://riunet.upv.es/handle/10251/148816>. [Último acceso: 2022].

- [9] R. González, «Estetoscopio electrónico inalámbrico para el registro de los sonidos cardiacos mediante un sensor piezoelectrónico,» Universidad Autónoma de Ciudad Juárez, [En línea]. Available: http://cathi.uacj.mx/bitstream/handle/20.500.11961/10838/Reporte%20t%C3%A1cnico_Estetoscopio%20inal%C3%A1mbrico%5b2%5d.pdf?sequence=1&isAllowed=y. [Último acceso: 2022].
- [10] STMicroelectronics, «78L05 Datasheet (PDF) - STMicroelectronics,» [En línea]. Available: <https://pdf1.alldatasheet.es/datasheet-pdf/view/22687/STMICROELECTRONICS/78L05.html>. [Último acceso: 2022].
- [11] Texas Instruments, «LP2985 150-mA, Low-Noise, Low-Dropout Regulator With Shutdown,» 2022. [En línea]. Available: https://www.ti.com/lit/ds/symlink/lp2985.pdf?ts=1685551322617&ref_url=https%253A%252F%252Fwww.ti.com%252Fproduct%252FLP2985%253Futm_source%253Dgoogle%252526utm_medium%253Dcpc%2526utm_campaign%253Dapp-null-null-GPN_EN-cpc-pf-google-wwe%2526utm_content%253DLP. [Último acceso: 2023].
- [12] Analog Devices, «Single and Dual-Supply, Rail-to-Rail, Low Cost Instrumentation Amplifier,» 2020. [En línea]. Available: <https://www.analog.com/media/en/technical-documentation/data-sheets/ad623.pdf>. [Último acceso: 2023].
- [13] CUI Devices, «MODEL: CMC-6015-42P,» 2022. [En línea]. Available: <https://www.cuidevices.com/product/resource/cmc-6015-42p.pdf>. [Último acceso: 2022].
- [14] Adafruit, «Adafruit AGC Electret Microphone Amplifier - MAX9814,» 2014. [En línea]. Available: <https://learn.adafruit.com/adafruit-agc-electret-microphone-amplifier-max9814/downloads> . [Último acceso: 2022].
- [15] Analog Devices, «Microphone Amplifier with AGC and Low-Noise Microphone Bias,» [En línea]. Available: <https://www.analog.com/media/en/technical-documentation/data-sheets/max9814.pdf>. [Último acceso: 2023].
- [16] Texas Instruments, «LMx24, LMx24x, LMx24xx, LM2902, LM2902x, LM2902xx, LM2902xxx, Quadruple Operational Amplifiers,» 2023. [En línea]. Available: https://www.ti.com/lit/ds/symlink/lm324.pdf?ts=1666806120023&ref_url=https%253A%252F%252Fwww.ti.com%252Fsitesearch%252Fen-us%252Fdocs%252Funiversalsearch.tsp%253FlangPref%253Den-US%2526searchTerm%253Dlm324n%2526nr%253D43 . [Último acceso: 2023].
- [17] Diodes Incorporated, «1N5817 - 1N5819,» 2023. [En línea]. Available: <https://www.diodes.com/assets/Datasheets/1N5817-1N5819.pdf>. [Último acceso: 2023].

- [18] ATMEL, «ATtiny25/V / ATtiny45/V / ATTiny85/V,» [En línea]. Available: https://ww1.microchip.com/downloads/en/DeviceDoc/Atmel-2586-AVR-8-bit-Microcontroller-ATTiny25-ATTiny45-ATTiny85_Datasheet.pdf. [Último acceso: 2023].
- [19] e-Gizmo Mechatronix Central , «Bluetooth Modules Wireless UART Cable Replacement,» [En línea]. Available: <https://cdn.instructables.com/ORIG/FQ1/UVZ/HXA9PUVQ/FQ1CUVZHXA9PUVQ.pdf>. [Último acceso: 2023].
- [20] Texas Instrument, «Single-Supply, Electret Microphone Pre-Amplifier,» Enero 2015. [En línea]. Available: https://www.ti.com/lit/ug/tidu765/tidu765.pdf?ts=1660042538064&ref_url=https%253A%252F%252Fwww.google.com%252F. [Último acceso: 2022].
- [21] Texas Instruments, «Single-supply, 2nd-order, Sallen-Key band-pass,» 2021. [En línea]. Available: https://www.ti.com/lit/an/sboa229/sboa229.pdf?ts=1661745229326&ref_url=https%253A%252F%252Fwww.google.com%252F. [Último acceso: 2022].
- [22] jjoe64, «Chart and Graph Library for Android,» github, [En línea]. Available: <https://github.com/jjoe64/GraphView>.
- [23] N. Rajlich y M. Prestel, «Java-WebSocket,» org.java-websocket, [En línea]. Available: <https://central.sonatype.com/artifact/org.java-websocket/Java-WebSocket/1.5.3>.
- [24] N. Mechatronics, «CONFIGURACIÓN DEL MÓDULO BLUETOOTH HC-05 USANDO COMANDOS AT,» [En línea]. Available: https://naylampmechatronics.com/blog/24_configuracion-del-modulo-bluetooth-hc-05-usando-comandos-at.html.