

# Instituto Tecnológico de Hermosillo

## División de Estudios de Posgrado e Investigación

Sistema de Monitoreo Inteligente de Bioseñales No Invasivo para la  
Detección de Presencia de Apneas en Pacientes Neonatales

T E S I S

Presentada por:

Ing. Oscar Daniel Atanacio Pérez

Como requisito parcial para  
obtener el grado de

Maestro en Ciencias de la Computación

Director de tesis:

Dra. María Trinidad Serna Encinas

Hermosillo, Sonora, México

Diciembre de 2018



Instituto Tecnológico de Hermosillo

SECCIÓN: DIV. EST. POS. E INV.  
No. OFICIO: DEPI/356/18.  
ASUNTO: AUTORIZACIÓN DE IMPRESIÓN  
DE TESIS.

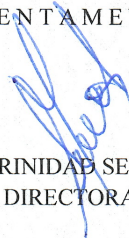
07 de Diciembre de 2018

C. OSCAR DANIEL ATANACIO PÉREZ,  
P R E S E N T E.

Por este conducto, y en virtud de haber concluido la revisión del trabajo de tesis que lleva por nombre **“SISTEMA DE MONITOREO INTELIGENTE DE BIOSEÑALES NO INVASIVO PARA LA DETECCIÓN DE PRESENCIA DE APNEAS EN PACIENTES NEONATALES”**, que presenta para el examen de grado de la MAESTRÍA EN CIENCIAS DE LA COMPUTACIÓN, y habiéndola encontrado satisfactoria, nos permitimos comunicarle que se autoriza la impresión del mismo a efecto de que proceda el trámite de obtención de grado.

Deseándole éxito en su vida profesional, quedo de usted.

ATENTAMENTE



DRA. MARÍA TRINIDAD SERNA ENCINAS  
DIRECTORA



M.C. CÉSAR ENRIQUE ROSE GÓMEZ  
SECRETARIO



M.C. ROSALÍA DEL CARMEN GUTIÉRREZ URQUIDEZ.  
VOCAL



M.C.O. ROSA IRENE SÁNCHEZ FERMÍN  
JEFA DE LA DIVISIÓN DE ESTUDIOS DE POSGRADO E INVESTIGACIÓN



S.E.P.

INSTITUTO TECNOLÓGICO  
DE HERMOSILLO  
DIVISIÓN DE ESTUDIOS  
DE POSGRADO

RISF/momv\*



# Lista de contenido

<b>Lista de figuras</b>	<b>IV</b>
<b>Lista de Tablas</b>	<b>VII</b>
<b>1. Introducción</b>	<b>1</b>
1.1. Introducción . . . . .	1
1.2. Antecedentes . . . . .	2
1.3. Planteamiento del problema . . . . .	3
1.4. Obejtivos . . . . .	4
1.4.1. Objetivo General . . . . .	4
1.4.2. Obejtivos Particulares . . . . .	5
1.5. Justificación de la Tesis . . . . .	5
1.6. Alcances y delimitaciones . . . . .	6
1.7. Metodología . . . . .	7
1.8. Organización de la tesis . . . . .	8
<b>2. Estado del arte</b>	<b>9</b>
2.1. Fisiopatología de la Apnea Neonatal . . . . .	9
2.1.1. Definición y Diagnóstico de una Apnea Neonatal . . . . .	10
2.1.2. Tipos de Apnea Neonatal y Manifestaciones Clínicas . . . . .	10
2.2. Desarrollo y Fisiología del Recién Nacido . . . . .	11
2.2.1. Desarrollo General del Neonato . . . . .	11
2.2.2. Sistema Respiratorio . . . . .	12
2.2.3. Sistema Cardiovascular . . . . .	15
2.2.4. ElectroCardioGramma (ECG) . . . . .	17

2.2.5.	Saturación Parcial de Oxígeno (SpO2)	20
2.3.	Señales Biomédicas, Procesamiento e Instrumentación	24
2.3.1.	Procesamiento de Señales Biomédicas	25
2.3.2.	Adquisición de Señales Biomédicas	28
2.4.	Inteligencia Artificial (IA) y aprendizaje automático (Machine Learning)	29
2.4.1.	Redes Neuronales Artificiales (RNA)	31
2.5.	Trabajos Relacionados	33
<b>3.</b>	<b>Análisis y Diseño</b>	<b>34</b>
3.1.	Introducción	34
3.2.	Metodología	34
3.3.	Análisis del Sistema	34
3.4.	Diagrama de Casos de Uso	36
3.4.1.	Caso de Uso 1: Login	37
3.4.2.	Caso de Uso 2: Catálogos	38
3.4.3.	Caso de Uso 3: Detección	40
3.4.4.	Caso de Uso 4: Reporte	41
3.5.	Diseño del Sistema	41
3.5.1.	Diagrama de Actividades	41
3.5.2.	Diagrama de Clases	44
3.5.3.	Diagrama de Componentes y Despliegue	45
3.6.	Modelo de Datos	46
3.7.	Arquitectura Propuesta	47
<b>4.</b>	<b>Implementación del sistema y Análisis de Resultados</b>	<b>49</b>
4.1.	Introducción	49
4.2.	Instrumentación del prototipo	49
4.3.	Adaptación de los sensores	51
4.3.1.	Sensor vestible ECG	51
4.3.2.	Sensor de saturación de Oxígeno	54
4.4.	Adquisición de señales	55

4.4.1.	Sparkfun AD8232, señal ECG . . . . .	55
4.4.2.	Maxim Integrated MAX30102, señal de SpO2 . . . . .	56
4.5.	Acondicionamiento de las Señales . . . . .	58
4.5.1.	Procesamiento de ECG . . . . .	59
4.5.2.	Procesamiento de SpO2 . . . . .	64
4.5.3.	Procesamiento EDR . . . . .	69
4.6.	Conjunto de Entrenamiento . . . . .	80
4.6.1.	MIMIC III Database Physionet . . . . .	81
4.6.2.	Detección de casos de Apnea . . . . .	82
4.6.3.	Selección de características . . . . .	93
4.7.	Red Neuronal para clasificación de Apneas . . . . .	93
4.7.1.	Preprocesamiento y creación de vectores de datos . . . . .	95
4.7.2.	Modelo de RNA . . . . .	96
4.7.3.	Métricas de Evaluación de RNA . . . . .	97
<b>5.</b>	<b>Resultados y Análisis</b>	<b>103</b>
5.1.	Adaptación del Hardware . . . . .	103
5.2.	Procesamiento Digital de las Señales . . . . .	104
5.2.1.	Procesamiento ECG . . . . .	104
5.2.2.	Procesamiento SpO2 . . . . .	105
5.2.3.	Procesamiento Señales Respiratorias EDR . . . . .	106
5.2.4.	Conjunto de Entrenamiento y RNA . . . . .	108
<b>6.</b>	<b>Conclusiones y Trabajo a Futuro</b>	<b>112</b>
6.1.	Conclusiones de la Tesis . . . . .	112
6.2.	Trabajo a Futuro . . . . .	116

## Lista de Figuras

1.1. Sistema Instrumental Biomédico Generalizado [3]. . . . .	2
1.2. Proceso general del Sistema de Monitoreo de Apneas para paciente neonatal . . . . .	8
2.1. Diagrama general del Sistema Respiratorio [11]. . . . .	12
2.2. Esquema general del proceso de intercambio gaseoso en el alvéolo humano [12]. . . . .	15
2.3. Gráfica de despolarización del miocito y ECG [13]. . . . .	17
2.4. Modelo de dipolo eléctrico del potencial de acción [14]. . . . .	18
2.5. Modelo esférico de dipolo y vector corazón [14]. . . . .	19
2.6. Distribución del eje cardíaco en el plano frontal, de acuerdo a derivaciones en extremidades [14]. . . . .	20
2.7. Características espectrales de la sangre [16]. . . . .	22
2.8. Interacción Luz incidente con dedo en medición de oximetría [16]. . .	23
2.9. Respuesta en frecuencia de un filtro pasa bajas [20] . . . . .	27
2.10. Tarjeta de Adquisición sencilla de ECG Sparkfun Electronics [22]. . .	29
2.11. Tarjeta de Adquisición de saturación de oxígeno [23]. . . . .	29
2.12. Diagrama General de una Red Neuronal Multicapa, con $n$ entradas y una salida [25]. . . . .	32
3.1. Metodología Propuesta para Sistema de Monitorización y Detección de apneas MIBB [Fuente Propia]. . . . .	35
3.2. Diagrama de contexto nivel 0 sistema de detección de apneas neonatal [Fuente Propia]. . . . .	35

3.3.	Diagrama de contexto nivel 1 sistema de detección de apneas neonatal [Fuente Propia]. . . . .	36
3.4.	Diagrama de casos de uso del sistema de detección de apneas neonatales [Fuente Propia]. . . . .	37
3.5.	Diagrama de actividad para Caso de Uso Login[Fuente Propia] . . . .	42
3.6.	Diagrama de actividad para Caso de Uso Detección [Fuente Propia] .	43
3.7.	Diagrama de actividad para Caso de Uso Reportes [Fuente Propia] . .	44
3.8.	Diagrama de clases Sistema MIBB . . . . .	45
3.9.	Diagrama de Componentes y Despliegue Sistema MIBB . . . . .	46
3.10.	Diagrama Modelo de Datos Sistema MIBB. [Fuente Propia] . . . . .	47
3.11.	Arquitectura de Sistema MIBB [Fuente Propia] . . . . .	48
4.1.	Diagrama electrónico de conexión de sensores [Fuente Propia]. . . . .	50
4.2.	Implementación electrónica circuito de adquisición de sistema MIBB. .	51
4.3.	Implementación de electrodo seco para adquisición de señales cardíacas.	52
4.4.	Implementación final de dispositivo de medición vestibular para adquisición de señales cardíacas neonatales. . . . .	53
4.5.	Comparación señales captadas mediante utilización de electrodos convencionales AgCl y electrodos secos. . . . .	54
4.6.	Diseño estuche sensor MAX30102 con espacio para vibradores, modelo de impresión 3D. . . . .	55
4.7.	Diagrama de componentes sensor MAX30102 [23]. . . . .	56
4.8.	Diagrama de Buffer FIFO de datos obtenidos por MAX30102 [23]. . .	58
4.9.	Diagrama general de procesamiento de señal cardíaca basado en [33].	59
4.10.	Diagrama de preprocesamiento de la señal de ECG basado en Rose et al [33]. . . . .	60
4.11.	Diagrama procesamiento de ECG para la localización de picos del complejo QRS basado en Rose et al [33]. . . . .	62
4.12.	Señal resultante de la implementación de preprocesamiento de ECG y localización de picos QRS. . . . .	63
4.13.	Diagrama general de limpieza de señal de SpO2 obtenida de MAX30102.	65

4.14. Señal resultante de la implementación de preprocesamiento de datos obtenidos de sensor MAX30102. . . . .	65
4.15. Diagrama general de implementación de localización de picos señal de SpO2 basado en Zong [34] y Rose et al [33]. . . . .	66
4.16. Caracterización de valores RR utilizando oxímetro comercial. . . . .	68
4.17. Diagrama general de procesamiento para EDR con amplitud RS ba- sado en [35]. . . . .	70
4.18. Diagrama general de procesamiento para EDR con área de QRS ba- sado en [35]. . . . .	70
4.19. Diagrama general de procesamiento para EDR filtrado por Arritmia Sinusal Respiratoria (RSA) basado en [35]. . . . .	71
4.20. Diagrama general de procesamiento para EDR con picos de SpO2 basado en [35]. . . . .	72
4.21. Señales Respiratorias derivadas de ECG [Fuente Propia]. . . . .	72
4.22. Diagrama de funcionamiento general de un filtro Kalman [41] . . . . .	77
4.23. Frecuencias Respiratorias derivadas de ECG en un tramos de 300 segundos con subdivisión de 5 segundos [Fuente Propia]. . . . .	79
4.24. Interfaz Gráfica Physiobank ATM para obtención de datos MIMIC III. 82	
4.25. Proceso de detección Apneas y cálculo de características [Fuente Pro- pia]. . . . .	83
4.26. Descripción de Apnea en forma gráfica con fenómenos fisiológicos no- tables [Fuente Propia]. . . . .	85
4.27. Diagrama Función procsignals() [Fuente Propia]. . . . .	88
4.28. Diagrama Función featureformation() [Fuente Propia]. . . . .	90



## Lista de Tablas

2.1. Energías aplicables al cuerpo humano, variables y señales biomédicas producidas [19]. . . . .	25
3.1. Tabla descriptiva de caso de uso Login [Fuente Propia] . . . . .	37
3.2. Tabla descriptiva de caso de uso catálogos [Fuente Propia] . . . . .	38
3.3. Tabla descriptiva de caso de uso altas [Fuente Propia]. . . . .	38
3.4. Tabla descriptiva de caso de uso baja [Fuente Propia]. . . . .	39
3.5. Tabla descriptiva de caso de uso cambio [Fuente Propia]. . . . .	39
3.6. Tabla descriptiva de Caso de Uso Detección [Fuente Propia]. . . . .	40
3.7. Tabla descriptiva de Caso de Uso Reporte [Fuente Propia]. . . . .	41
4.1. Tabla de resultados porcentaje de error 20 lapsos de tiempo de 300 segundos [Fuente Propia]. . . . .	80
4.2. Tabla de Características por variable para Conjunto de Entrenamiento [Fuente Propia]. . . . .	86
4.3. Tabla de organización de características de acuerdo a método de Filtrado [Fuente Propia]. . . . .	93
4.4. Estructura de Modelo de RNA utilizado. . . . .	97
4.5. Tabla resultados exactitud de entrenamiento. . . . .	100
4.7. Tabla de Resultados Evaluación de Validación de RNA. . . . .	101
4.6. Tabla de Resultados Evaluación de Entrenamiento de RNA. . . . .	102

## Lista de Códigos

4.1. configSensor ECG . . . . .	56
4.2. Función readSample de sensor MAX30102. . . . .	58
4.3. Código Función HRCalculation . . . . .	63
4.4. Cálculo de Saturación de Oxígeno . . . . .	67
4.5. Cálculo del Pulso Cardíaco. . . . .	68
4.6. Cálculo del índice de Pureza de la señal . . . . .	74
4.7. Filtrado de tipo Kalman basado en [41]. . . . .	77
4.8. Función get_data() . . . . .	86
4.9. Función calcmeancontrol() . . . . .	89
4.10. Función verifyApnoea() . . . . .	90
4.11. Función rocaucalc() . . . . .	91
4.12. Código de Preprocesamiento de Conjunto de Entrenamiento . . . . .	95
4.13. Función de creación de modelos RNA automatizado() . . . . .	96
4.14. Código de Aplicacion de la RNA al Conjunto de Entrenamiento. . . . .	98

# Resumen

En la presente Tesis se describe el Diseño e Implementación de un Sistema de Monitoreo Inteligente No Invasivo para la detección de casos de Apnea Neonatal. Una Apnea Neonatal es la interrupción del proceso respiratorio que puede tener implicaciones mortales o serias en el paciente.

Cada año, de acuerdo a la Organización Mundial de la Salud, un 45 % de los niños muertos menores a 5 años corresponden a lactantes o recién nacidos. El sistema fue desarrollado en un ambiente embebido Raspberry Pi 3 con el uso de sensores especializados y específicos que fueron adaptados durante el desarrollo de este proyecto para la adquisición de señales cardíacas y de oximetría de pulso de una manera segura y sin riesgo para el paciente neonatal, se desarrolla también el diseño general de software, incluyendo Modelos de 4+1 Vistas y modelo de datos.

La aplicación de Algoritmos de Derivación de Frecuencia Respiratoria Secundaria a partir de ECG y Pulso Oximetría en conjunto con el uso de Redes Neuronales Artificiales, fue realizado para dar al sistema la capacidad de distinguir entre un caso de apnea positivo y una falsa alarma sin correr el riesgo de la aplicación de corriente eléctrica en el cuerpo. Para este propósito, se creó un Conjunto de Entrenamiento para enseñar a la Red Neuronal como hacer la clasificación de las Apneas, obteniendo buenos resultados basados en literatura con un error aproximado de un 3 % entre distintos algoritmos inteligentes.

# Abstract

In this Thesis, it is described the Design and Implementation of an Intelligent Non-Invasive Monitoring System for Detection of cases of Neonatal Apnea. An apnea is the interruption of the respiratory process that can have mortal or serious implications in the patient.

Every year, according to the World Health Organization 45% of deaths in children under 5 years old are infants or newborns. The system is developed in an embedded Raspberry Pi 3 environment with the use of specialized and specific sensors that were adapted during the development of the project for the acquisition of pulse oximetry and electrocardiography signal in a safe way and without risk for the Neonatal patient. It is also described the general design of the software, including 4+1 views and data models.

The application of Secondary Respiratory Frequency Derivation Algorithms from ECG and Pulse Oximetry is carried out in addition with the use Artificial Neural Networks to give the system the capacity to distinguish a positive apnea case from a false alarm. For that purpose, a Training Dataset is built to teach the Neural Network how to make the classification of apnoeas, having a very good result based on the results reported literature with a 3% of error between intelligent algorithms.

Aqui van los agradecimientos...

# Capítulo 1: Introducción

## 1.1. Introducción

De acuerdo a datos de la Organización Mundial de la Salud, aproximadamente el 45 % de los niños menores de cinco años que fallecen cada año son lactantes recién nacidos; es decir, bebés de menos de 28 días o en período neonatal; además, tres de cuatro fallecimientos de recién nacidos se producen en la primera semana de vida [1].

Se define como apnea neonatal a aquel episodio de interrupción a la respiración o a la ausencia de flujo respiratorio en las vías aéreas menores con una duración mayor a 20 segundos, el cual se ve acompañado de repercusiones en el sistema cardiocirculatorio que pueden llevar a la muerte o a la afectación severa de los órganos vitales y de gran importancia para la homeostasis corporal; también entran en esta definición aquellas pausas mayores a 10 segundos asociadas con Bradicardia (Pulso Cardíaco  $<80$  latidos/min), y Saturación de Oxígeno deficiente ( $<80-85\%$  por Oximetría de Pulso) [2].

Hablando específicamente de seres humanos recién nacidos o neonatos, se tiene identificado a nivel mundial un padecimiento general que conlleva a un alto índice de mortalidad infantil, la apnea patológica neonatal o apnea de prematuridad dependiendo del caso del paciente. Dentro de las causas de muerte de este padecimiento se encuentran un detenimiento en el proceso de respiración de los recién nacidos por un periodo prolongado o una combinación de factores fisiológicos con pausas respiratorias cortas pero críticas.

En este proyecto de tesis se propone la implementación de un prototipo de un Sistema de Monitoreo Inteligente de Bioseñales No Invasivo para la Detección de

Presencia de Apnea en Pacientes Neonatales, con el cual se pretende poder dar aviso al personal médico cuando se presenta un caso de apnea y dar un estímulo al paciente para la interrupción del episodio cuando sea posible.

## 1.2. Antecedentes

El cuerpo humano es uno de los temas de estudio por excelencia a nivel mundial, el cual, con el paso del tiempo y la aplicación de conocimientos de ingeniería, se ha ido especificando cada vez más para lograr una mejor comprensión de su funcionamiento.

Se ha logrado la obtención de señales eléctricas características a diferentes fenómenos fisiológicos que ocurren en el funcionamiento diario del ser humano, mismas que se han utilizado para realizar mediciones tales como: el movimiento de los músculos, el corazón, la respiración, entre otros, que permiten dar solución y apoyo al diagnóstico, monitorización y tratamiento de diversas enfermedades.

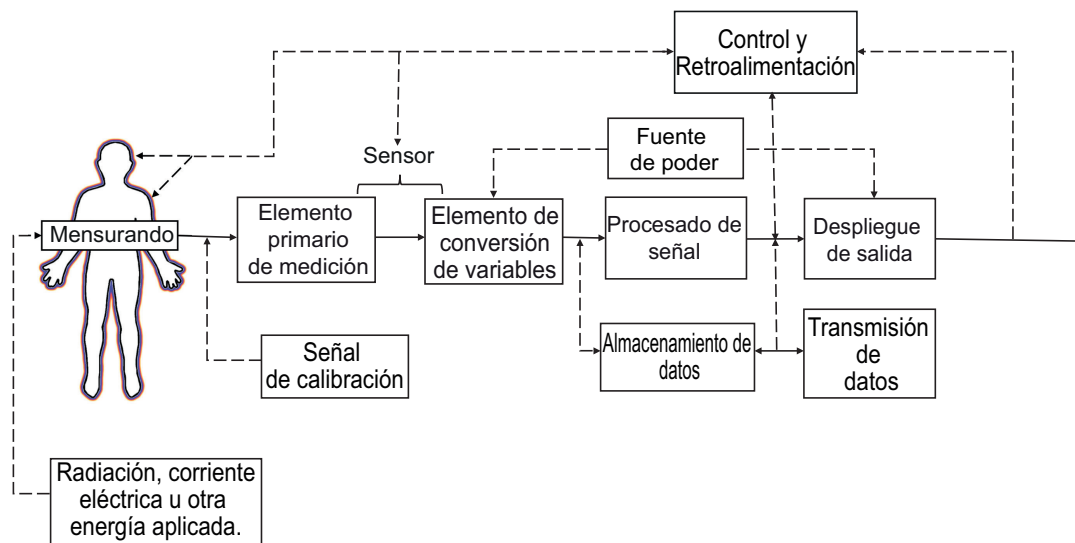


Figura 1.1: Sistema Instrumental Biomédico Generalizado [3].

La tecnología biomédica ayuda a la obtención de manera electrónica de las características físicas de los diferentes sistemas corporales, y ha sido generalizada en base a métodos de desarrollo de instrumentación específica. Cada sistema de instrumentación tiene elementos globales como los presentes en la figura 1.1, en donde se tiene de manera esquemática el proceso general de obtención y tratamiento de

señales biomédicas, para la creación de dispositivos capaces de utilizar estas señales con algún fin específico, tomando en cuenta la aplicación o captación de un tipo de energía y un sensor específico, un método de análisis, almacenamiento y despliegue de la información obtenida para la interpretación de las mediciones realizadas por el personal de salud [3].

En la actualidad, se tienen diferentes tipos de sistemas de detección de apneas con diversa complejidad entre ellos, pueden ser encontrados ya sea desde casa, con sistemas simples o en Unidades de Cuidados Intensivos Neonatales (UCIN), siendo estos últimos más específicos y complejos. Los sistemas caseros pueden ser sistemas estáticos o portables, haciendo uso de tecnología inalámbrica y de telefonía móvil en algunos casos; mientras que los encontrados dentro de la UCIN, por lo general son monitores de signos vitales, con la función de detección o sistemas con algoritmos de detección.

Para el censo y detección de los eventos de apnea se tienen diferentes formas para realizar la medición de parámetros, entre ellos: electrofisiológica con electrodos; sensores de movimiento y posición; sensores de presión; entre otros. Todos ellos ayudan a tener una señal fisiológica con la cual se puede realizar la determinación de la presencia de apnea en pacientes neonatales. Esta determinación puede ser realizada mediante el uso de algoritmos computacionales y algoritmos inteligentes, que analizan las señales obtenidas de los sensores y adquieren ciertas características de las mismas, para poder realizar la determinación de la presencia de apnea y generación de la alarma.

### **1.3. Planteamiento del problema**

En 2003, México registró 20,806 defunciones neonatales, de las cuales el 49.4 % corresponden a muertes por asfixia o apnea neonatal[4]. Tal y como se ha mencionado con anterioridad, este tipo de patologías se observan con mayor frecuencia en pacientes neonatales y neonatales prematuros, quienes, al no recibir la monitorización y tratamiento adecuado a tiempo, pueden desarrollar afecciones nerviosas y musculares debido al nivel de oxigenación presente en la sangre, e incluso llegar hasta la muerte.



Al abordar el fenómeno en estudio, se pueden generar las siguientes preguntas a manera de guía para la comprensión de la problemática enfrentada tanto a nivel fisiológico, biomédico y computacional:

- ¿Qué es la apnea neonatal, sus tipos y características individuales?
- ¿Cómo se diagnostica y trata la apnea neonatal?
- ¿Qué tipos de equipos existen actualmente que facilitan el monitoreo de pacientes neonatales?
- ¿Qué tipo de señales biomédicas e instrumentación se utilizarían para caracterizar las condiciones patológicas de la apnea neonatal?
- ¿Es posible la utilización de los sensores actuales para la detección de las señales biomédicas, o se tendrá que realizar la adaptación de sensores no invasivos para la utilización con pacientes neonatales?
- ¿Es posible la utilización de sistemas inteligentes para la detección de un caso de apnea neonatal?

Una vez conocidas estas preguntas a investigar se puede definir una problemática específica para el estudio del fenómeno de la siguiente manera:

- ¿Qué metodología biomédica, componentes instrumentales y computacionales son necesarios para la implementación de un sistema inteligente que detecte la presencia de apnea en pacientes neonatales?

## **1.4. Obejtivos**

Se plantean los siguientes objetivos para el proyecto de tesis:

### **1.4.1. Objetivo General**

Diseñar e implementar un sistema de monitoreo no invasivo capaz de realizar la determinación de la presencia de apnea en pacientes neonatales para facilitar su

detección y tratamiento, mediante el procesamiento de señales biomédicas y sistemas inteligentes, incluyendo la generación de alertas y un estímulo para la interrupción del episodio.

### **1.4.2. Obejtivos Particulares**

A partir del objetivo general de esta investigación se desarrollaron los siguientes objetivos particulares, a fin de tener una visión más clara de las actividades a realizar durante la duración de la investigación:

- Conocer la fisiopatología de la apnea neonatal y su relación con las señales biomédicas.
- Determinar las señales biomédicas a utilizar y su instrumentación específica.
- Diseñar la instrumentación correspondiente al prototipo para la detección de la apnea neonatal y otras señales importantes.
- Implementar la instrumentación necesaria para la obtención de señales biomédicas, para el prototipo.
- Adaptar los sensores a utilizar para las condiciones específicas del usuario neonatal.
- Seleccionar algoritmos computaciones o inteligentes para la clasificación de la apnea neonatal.
- Diseñar el protocolo de pruebas para verificar y validar la funcionalidad del sistema.

## **1.5. Justificación de la Tesis**

Teniendo este panorama en mente, se considera a la muerte neonatal por apnea uno de los problemas eje en el aumento de la tasa de mortalidad infantil, tanto a nivel mundial como a nivel nacional. La muerte por apnea es considerada desde el ámbito de calidad en atención a la salud como un “Evento Centinela”, el cual cumple

todas las características necesarias al afectar directamente al paciente recién nacido, y tener repercusiones tanto a nivel neural, así como la muerte.

El alto grado de repercusión hospitalaria en dichos eventos es lo que ha llevado a la creación de estudios, como el realizado por la Sociedad Argentina de Pediatría donde se observó que la práctica general de monitoreo del recién nacido no es lo suficiente para la detección y prevención del fallecimiento por apnea neonatal, teniendo resultados como el que un 42 % de los infantes fallecidos se encontraban con monitorización de Saturación de Oxígeno, 13 % por monitor multi-paramétrico y 8.4 % sin monitorización[5]; cabe aclarar que los equipos de monitorización utilizados en dichos casos son desconocidos en el estudio y por lo tanto, sus capacidades de detección de fenómenos fisiológicos son desconocidos, por lo que la detección de apnea por medio visual realizado por la enfermera o familiar del paciente sigue siendo practicada con un alto grado de incertidumbre.

Lo que lleva a la investigación de este fenómeno y al diseño de soluciones posibles y favorables para el buen desarrollo de las buenas prácticas en la salud del paciente neonatal, que sean accesibles para los distintos sectores sociales y de salud del país, aumentando y mejorando la calidad de los servicios y con mayor impacto en la búsqueda de procesos tecnológicos que permitan disminuir los niveles de mortalidad de dicha enfermedad, al mezclar sistemas computacionales e inteligentes con fenómenos fisiológicos desde un punto de vista biomédico y para la mejora del paciente.

## 1.6. Alcances y delimitaciones

Del conjunto de las señales biomédicas presentes en el cuerpo humano, el equipo de trabajo decidió la utilización de tres señales específicas siendo las siguientes: Saturación Parcial de Oxígeno ( $SpO_2$ ), Electrocardiograma (ECG) y la medición de la Frecuencia Respiratoria.

Otra de las delimitaciones del proyecto es la simple detección de una apnea, mas no la especificación del tipo de apnea presente en el paciente, por lo que aun así se necesitará la presencia del personal de salud lo más rápido posible para detectar el tipo de apnea y actuar con el tratamiento necesario, así como también la correcta identificación de la presencia de apnea.

## 1.7. Metodología

Para la realización del proyecto en estudio se tendrán tres fases como se ilustra en la Figura 1.2:

- La fase I consistirá en un análisis teórico-práctico del sistema, incluyendo el estudio exhaustivo para el desarrollo del estado del arte con las temáticas principales de la problemática presentada, el desarrollo de las ideas propuestas para la elección de la mejor solución, así como también un análisis de los equipos y dispositivos que existen actualmente para atacar la problemática. Una vez conocida la problemática en su totalidad, se realizará la especificación y selección de las señales corporales necesarias para el desarrollo del proyecto, la planeación del uso o adaptación de sensores necesarios para captar dichas señales y el diseño general del sistema con instrumentación teórica y arquitectura del sistema.
- La fase II, fase de armado del prototipo, comenzando con el análisis y diseño del mismo prototipo, una vez diseñado totalmente el dispositivo, se procederá a la construcción de los módulos que conforman cada una de las partes instrumentales para la obtención de los biopotenciales seleccionados para el estudio, finalmente se tendrá la implementación del sistema. En la segunda fase también incluirá el desarrollo del sistema, que facilite el registro, procesamiento y tratamiento de las señales obtenidas de manera digital para implementación de un algoritmo que permita clasificar la presencia de una apnea en el paciente neonatal.
- Durante la fase III, se creará el protocolo de pruebas para la validación del funcionamiento del sistema y se llevará a cabo la ejecución y evaluación del protocolo para determinar los cambios a realizar y el funcionamiento general del sistema.

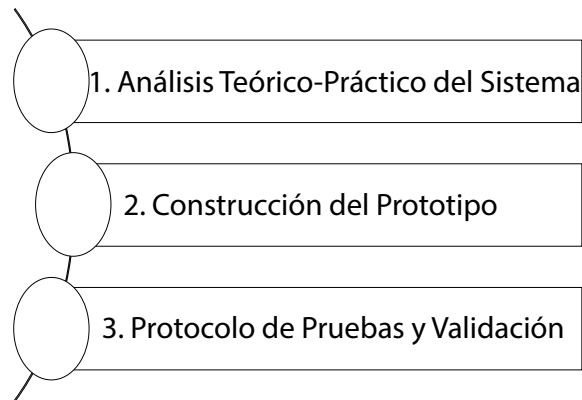


Figura 1.2: Proceso general del Sistema de Monitoreo de Apneas para paciente neonatal

## 1.8. Organización de la tesis

Después del capítulo de Introducción se tendrá en el capítulo II, el estado del arte, donde se incluirá un estudio y análisis de los temas a estudiar basado en las preguntas de investigación, una especificación y descripción de la tecnología existente para la realización de la tarea de detección de las apneas en pacientes neonatales y la aplicación de sistemas inteligentes a problemas del ámbito biomédico, con mayor énfasis en la detección de fenómenos fisiológicos a partir de señales biomédicas.

En el capítulo III se describe el diseño y especificación de la instrumentación biomédica, modelo de datos requerido y la arquitectura de software propuesta para la realización del proyecto. El Capítulo IV consiste en la implementación del sistema de monitorización, y de los módulos requeridos para su procesamiento, así como el análisis de resultados obtenidos. Finalmente, el Capítulo V presenta las conclusiones obtenidas en esta investigación e incluye algunas sugerencias de trabajo a futuro.

## Capítulo 2: Estado del arte

Para el estudio y desarrollo del proyecto, es necesario tener un conocimiento previo para el correcto abordaje de la problemática que se expuso en el Capítulo 1; es por eso, que en el Capítulo 2 se desarrollarán cada una de las líneas de investigación del proyecto de manera individual, siendo las siguientes:

- Fisiopatología de la Apnea Neonatal.
- Desarrollo y Fisiología del Recién Nacido.
- Señales Biomédicas, Instrumentación y Procesamiento.
- Inteligencia Artificial.
- Trabajos Relacionados

### 2.1. Fisiopatología de la Apnea Neonatal

La apnea neonatal está presente en el 25 % de los recién nacidos pretérmino con un peso menor de 2500 gramos y entre el 84-100 % en los que pesan menos de 1000 gramos. De acuerdo con la Academia Americana de Pediatría la apnea neonatal se presenta con mayor frecuencia en neonatos con gestación menor a 37 semanas de edad y puede presentarse hasta las 43 semanas de edad, por lo que se puede decir, que la presencia y gravedad de la apnea neonatal se relaciona de forma inversa con la edad gestacional y la madurez del centro respiratorio [6], el cual no se ve completamente maduro para el nacimiento del bebé al ser prematuro y siendo esta considerada como la causa del cese del esfuerzo respiratorio en los neonatos nacidos antes de la edad gestacional normal [4][7].

### **2.1.1. Definición y Diagnóstico de una Apnea Neonatal**

La Academia Americana de Pediatría define a la Apnea del Prematuro a una pausa en la respiración con una duración mayor a 20 segundos, o una pausa con una duración menor a 20 segundos pero que tiene una relación directa con otros síntomas fisiológicos que ponen en riesgo la vida del neonatos, la disminución de la frecuencia cardíaca (bradicardia), cambio de coloración en la piel (cianosis) por disminución de nivel de oxigenación presente en la sangre del neonato [7].

Sin embargo, en los neonatos se presenta otro fenómeno conocido como respiración periódica la cual es un patrón respiratorio irregular con pequeñas pausas con una duración pequeña, pero las cuales no tienen una repercusión cardiocirculatoria a diferencia de la apnea y que cuentan con una recuperación espontánea normal. Éstos patrones son considerados normales en la respiración de un recién nacido [8].

### **2.1.2. Tipos de Apnea Neonatal y Manifestaciones Clínicas**

La apnea en neonatos puede ser clasificada en dos grupos primarios: La Apnea Primaria, causada por la inmadurez de los mecanismos respiratorios y el control global de la respiración del neonato, y la Apnea Secundaria, causada como efecto secundario de alguna otra enfermedad que ataque al paciente neonatal, sin importar las condiciones de nacimiento [8].

En ambos casos las apneas pueden ser clasificadas en dos grupos de acuerdo a su origen; la apnea de origen central se caracterizan por tener una ausencia total de flujo de aire en las vías respiratorias y un cese en los movimientos respiratorios que conllevarán a una afectación cardiovascular, este tipo de apnea es de origen neural y corresponde a un 40 % de los casos, [6]. Por el otro lado, se tiene el caso de la apnea por obstrucción, en la que, se presenta una ausencia de flujo en las vías aéreas del neonato pero con la presencia de esfuerzos inspiratorios, ésta corresponde a un 10 % de los casos; como última clasificación se tiene una mezcla de ambos tipos o un tipo mixto, en el cual, una apnea central es producida como consecuencia de una obstrucción en la vía aérea [6].

## 2.2. Desarrollo y Fisiología del Recién Nacido

Es necesario conocer la anatomía y fisiología de un paciente neonatal, debido a que los procesos fisiológicos que permiten un buen funcionamiento del cuerpo humano se ven afectados con el paso del tiempo, iniciando en procesos de adaptación al medio en los primeros años de vida, mantención del homeostasis general del cuerpo en la edad adulta y procesos geriátricos en la edad avanzada.

### 2.2.1. Desarrollo General del Neonato

El desarrollo se inicia con la fecundación, proceso en el cual el gameto masculino o espermatozoide y el gameto femenino u ovocito se fusionan y originan un cigoto.[9]. Durante la tercera y la octava semana de gestación se da un proceso llamado gastrulación, en el cual se crean tres capas germinales conocidas como: ectodermo, mesodermo y endodermo, donde se originan diversos tejidos y órganos específicos, en un periodo conocido como organogénesis. Cuando el periodo embrionario llega a su fin, ya se han generado los principales sistemas de órganos y hacia el final del segundo mes ya es posible reconocer las principales características corporales externas [9].

Después de la tercera semana, los pliegues se juntan formando las paredes de las cavidades del cuerpo del bebé; se forma el cordón umbilical y las cavidades peritoneal, pleural y pericárdica que alojarán al sistema digestivo, respiratorio y cardíaco; se forma también el diafragma que se encarga casi en su totalidad de la funcionalidad de la respiración y que separa la cavidad peritoneal de la torácica, la cual está dividida en una cavidad pericárdica y dos cavidades pleurales [9].

Un feto que nace en el sexto mes, tiene pocas posibilidades de sobrevivir pues aunque muchos de los órganos ya son capaces de funcionar, el sistema respiratorio y el nervioso no se encuentran completamente desarrollados y no se tiene una buena condición entre ambos.



## 2.2.2. Sistema Respiratorio

El sistema respiratorio se compone por vías respiratorias, cuya función primordial es dirigir y acondicionar el aire inspirado, pero también cumplen funciones externas al proceso respiratorio como la fonación, la deglución y la protección de los pulmones. La vía aérea general del ser humano comienza con la nariz, faringe y laringe (también denominada vía aérea superior), continúa con el árbol tranqueo bronquial que incluye la tráquea, bronquios mayores, (también conocida como vía aérea central) y por último con subdivisiones hasta el bronquiolo alveolar donde se realiza el intercambio de gases, éstos últimos son conocidos como la vía aérea pequeña; la estructura general del sistema respiratorio se muestra en la figura 2.1 [10].

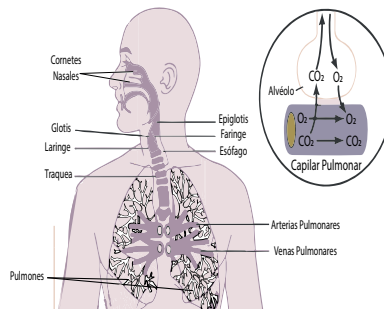


Figura 2.1: Diagrama general del Sistema Respiratorio [11].

Los órganos que conforman el sistema respiratorio trabajan en conjunto para realizar el proceso de respiración, el cual es vital para poder subsistir y mantener un equilibrio homeostático en los procesos generales de las células y del cuerpo, durante este proceso, se genera a su vez un intercambio de gases que permite que el oxígeno, sustancia necesaria para mantener vivo el cuerpo humano, pueda llegar a todas las células del cuerpo y mantener su vitalidad y funcionamiento. Sin embargo, este proceso no comienza solamente con el inicio funcional del sistema respiratorio si no que desde la gestación con el paso de nutrientes a través del cordón umbilical.

### Proceso de Respiración

La respiración permite la llegada del oxígeno inspirado al interior de los pulmones a los tejidos y el desecho del dióxido de carbono una vez que es aprovechado.

Se compone de cuatro funciones principales para completar el proceso general: la ventilación pulmonar, difusión de oxígeno y dióxido de carbono entre los alvéolos y la sangre, el transporte de los iones de oxígeno y dióxido de carbono hacia las células y viceversa, la regulación de la ventilación y demás facetas de la respiración [11].

Los pulmones contienen alrededor de  $3 \times 10^8$  alvéolos, en donde el aire y la sangre son aproximados a nivel molecular para que el intercambio gaseoso pueda ocurrir. En circunstancias normales, la membrana alveolar presenta una barrera a la difusión que, la sangre presente en los capilares alveolares alcanza un equilibrio con el aire alveolar antes de entrar al capilar [12]. Los pulmones pueden realizar su movimiento de expansión y contracción siguiendo dos maneras: 1) mediante el movimiento vertical del diafragma para alargar o acortar la cavidad torácica y 2) mediante la elevación y el descenso de las costillas para aumentar el diámetro de la cavidad torácica y dar más espacio a la expansión pulmonar. Durante la inspiración, la contracción del diafragma, tira hacia debajo de las superficies inferiores de los pulmones; durante la espiración el diafragma se relaja y permite el retroceso elástico de los pulmones, y la pared torácica y abdominal comprime a los pumones para expulsar el aire [11]

### **Intercambio Gaseoso**

En los seres humanos, las ramas de la tráquea se vuelven cada vez más numerosas y se forman capas en una distribución parecida a las ramas de un árbol, donde decrece en tamaño y los anillos cartilagosos de la tráquea son sustituidos por mucosa y músculo liso, sin la presencia de alvéolos, la unidad funcional del proceso de la respiración [2].

El área en donde el intercambio ocurre es conocido como zona respiratoria, lugar en el que se ubican los alvéolos, los cuales, son los primeros en generar el intercambio gaseoso. Los alvéolos están envueltos en una fina capa de tejido capilar donde se generará el intercambio gaseoso y se le conoce como el parénquima pulmonar [2].

El fenómeno del intercambio de gases se ve relacionado enteramente a un gradiente de presiones del flujo sanguíneo, ya que ocurre por medio de un transporte por difusión, de un lugar de alta presión a un lugar de baja presión, como ocurre en

el movimiento de los iones de oxígeno y el dióxido de carbono presentes en el aire y la sangre [2].

El gradiente de presión que fluye tanto en vías aéreas como vasculares en el cuerpo humano, contiene un conjunto de presiones parciales de los gases que conforman la presión general del aire o la presión de la sangre ya que se encuentran diluidos en la misma [2]. La presión parcial de un gas es definida como la fracción molar del gas multiplicada por la presión total del gas. Si se tuviera un gas a una concentración  $U$  del gas en un líquido, se consideraría que:

$$U = \sigma * P_s \quad (2.1)$$

Donde es  $\sigma$  es la solubilidad del gas en el líquido y la presión parcial dependerá enteramente de la solubilidad del gas con la concentración del mismo presente en el líquido donde está disuelto [12].

Dentro de los alvéolos, el oxígeno es difundido a la sangre del capilar pulmonar, siguiendo la Ley de Fick, que describe la difusión del oxígeno a través de los tejidos tomando en cuenta sus características físicas. Si se tiene un gas a una presión parcial específica  $P_g$  entra en contacto con un líquido que contiene el mismo gas disuelto a una concentración  $U$  y una presión parcial  $P_s$ , se esperaría una difusión del mismo dependiente de una diferencia de las presiones presentes dentro de la interfaz del contacto superficial, por lo que un flujo de volumen del gas  $q$  por unidad de área del gas (positiva al fluir hacia la parte disuelta y negativa hacia la parte aérea) dependerá de una constante de difusión de superficie que corresponde al área de superficie de contacto dividido entre el grosor del tejido y la diferencia de presiones parciales [2, 12]:

$$q = D_s * (P_g - P_s) \quad (2.2)$$

$$D_s = \frac{\text{Área de superficie}}{\text{Grosor de tejido}} \quad (2.3)$$

El pulmón humano puede contener un máximo de volumen de 5000 ml de aire, de los cuales, sólo 350 ml participarán en el intercambio gaseoso. Si se consideran dos flujos,  $\dot{V}$  y  $Q$  para el paso interior y exterior de los capilares alveolares, y un

gasto cardíaco de 70 ml por latido con 72 latidos por minuto, se puede obtener una proporción de ventilación-perfusión, la cuál correlaciona el funcionamiento general del corazón con el intercambio gaseoso en los pulmones. A su vez, esta proporción es factor determinante más importante en el intercambio gaseoso pulmonar [12].

De acuerdo a la figura 2.2, si se consideran dos medidas de flujo volumétrico,  $\dot{V}$  y  $Q$  para el paso dentro y fuera del capilar alveolar, se podrá calcular una proporción de flujo que dependerá del presente en el vaso sanguíneo y una diferencia de las presiones inspiratoria y alveolar ( $P_i$  y  $P_a$ ). Para el flujo sanguíneo de los capilares en el alvéolo se podrá determinar una diferencia de concentraciones del gas disuelto en la sangre arterial y venosa ( $c_a$  y  $c_v$ ) y un flujo general  $Q$ . Como debe existir un equilibrio entre el gas que sale y el flujo sanguíneo entrante en el alvéolo, se debe crear una nueva proporción que corresponderá a la proporción de ventilación-perfusión [12].

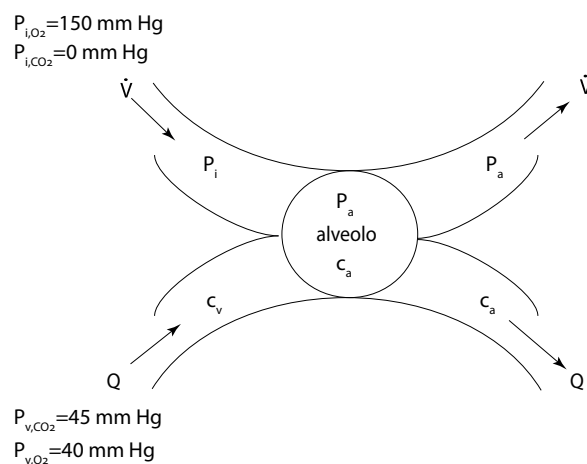


Figura 2.2: Esquema general del proceso de intercambio gaseoso en el alvéolo humano [12].

### 2.2.3. Sistema Cardiovascular

El sistema cardiovascular está compuesto por dos subsistemas: el sistema cardíaco y el sistema vascular. El sistema cardíaco abarca la anatomía y funcionamiento del corazón, órgano con la función de bombear sangre hacia el resto del cuerpo para que esta pueda ser aprovechada en las diferentes estructuras que son irrigadas por el

sistema circulatorio. Está compuesto por cuatro cámaras interconectadas con válvulas cardíacas con características específicas individuales, estas cámaras son divididas como dos bombas separadas: un corazón derecho que bombea la sangre hacia los pulmones y un corazón izquierdo que bombea la sangre hacia los órganos periféricos compuestos por una aurícula y un ventrículo, cuyo funcionamiento rítmico permite que se dé el impulso de la sangre hacia la circulación pulmonar o periférica [11].

Por su parte, el sistema circulatorio o vascular está formado por órganos cuya función es de distribuir la sangre al cuerpo, con estructuras para transporte de sangre oxigenada o arterias y estructuras para sangre desoxigenada o venas, ambas con subdivisiones en arteriolas y vénulas respectivamente y una subdivisión en capilares arteriales y venosos donde se realiza el intercambio de iones entre las células y el líquido intersticial [11].

### **Fisiología del sistema cardíaco**

El corazón está formado por tres tipos principales de músculo cardíaco: músculo auricular, músculo ventricular y fibras musculares especializadas de excitación y de conducción. El funcionamiento rítmico de las mismas permite la generación de movimientos pulsátiles mejor conocido como ciclo cardíaco [11, 12], el cual está formado por un periodo de relajación denominado diástole y un periodo de contracción llamado sístole, el ciclo cardíaco es el valor inverso de la frecuencia cardíaca [11].

La actividad del corazón inicia en un conjunto de células llamado nodo sinoauricular (SA) que genera una señal eléctrica de despolarización que recorre al corazón produciendo las contracciones necesarias para el ciclo cardíaco, las células del nodo SA son osciladores autónomos y rítmicos. Después de las aurículas, se encuentra el nodo auriculoventricular (AV) que pasará la despolarización a través de las fibras de Purkinje que se distribuyen en la pared ventricular, por lo que la contracción ventricular ocurrirá tras el paso del impulso eléctrico eyectando la sangre hacia afuera del corazón [12].

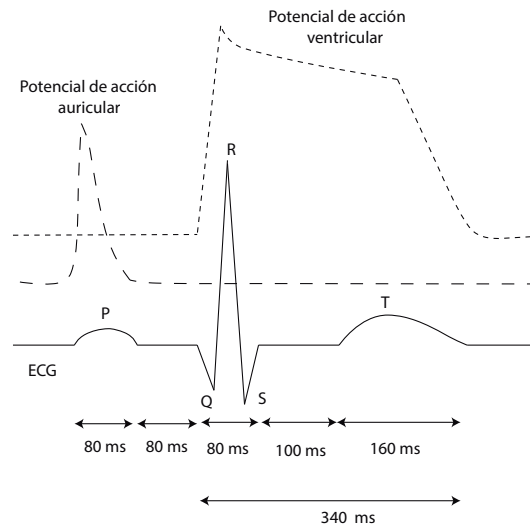


Figura 2.3: Gráfica de despolarización del miocito y ECG [13].

#### 2.2.4. ElectroCardioGramma (ECG)

El estudio del sistema cardíaco ha sido una de las herramientas más antiguas y estudiadas para el tratamiento de enfermedades ya que permite conocer el estado general del corazón y la conductividad eléctrica del mismo, al colocar electrodos en el pecho o en extremidades y observar la señal resultante [12, 13].

El Electrocardiograma ha sido conocido desde 1877, cuando se obtuvo por primera vez al observarse que los potenciales de acción generados a partir de la actividad cardíaca puede medirse a través de la superficie corporal [12]. El cuerpo humano puede ser descrito como un conductor volumétrico, por lo que, al generarse un paso de corriente iónica en las células hay un cambio en el potencial eléctrico que en la superficie corporal. La adquisición de esta señal se da al calcular la diferencia entre dos potenciales de superficie en el torso; normalmente, el vector obtenido tiene una estructura periódica que puede ser observada en la figura 2.3. Cuando el potencial de acción se propaga a través de las aurículas, se genera la onda P, cuando el potencial se propaga a través de los ventrículos se genera una deflexión más larga, conocida como complejo QRS, finalmente a la recuperación del tejido ventricular se conoce como onda T [12].

Si el corazón estuviera suspendido en un medio conductor homogéneo e isotró-

pico y fuera observado a una distancia lo suficientemente grande comparado con su tamaño, entonces, se pudiera asumir que todos los dipolos individuales se originan en un punto en común en el espacio y que la actividad total del corazón puede ser representada como un dipolo global equivalente, cuya magnitud y dirección correspondería a la suma de las magnitudes y direcciones de todos los dipolos individuales de las células. Es decir, que se puede ver el dipolo en la superficie corporal como un vector del dipolo del corazón, el cual se forma a través de un comportamiento aditivo en lazo cerrado en cada ciclo de transmisión del potencial de acción de una célula cardíaca a otra esquematizado en la figura 2.4 regido por la ecuación 2.4.

$$H(t) = \int_V JdV \quad (2.4)$$

Donde  $J$  representa la densidad del dipolo en cada punto del corazón y  $V$  el volumen del corazón, siendo  $H$  un valor que cambia en función del tiempo [12, 14].

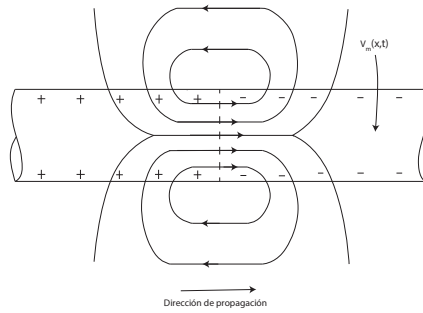


Figura 2.4: Modelo de dipolo eléctrico del potencial de acción [14].

La distribución de las superficies de las corrientes y potenciales resultantes depende de las propiedades eléctricas del torso; el modelo del dipolo trata al cuerpo como un conductor esférico lineal, isotrópico, homogéneo de radio  $R$  y conductividad  $\sigma$ . La fuente de corriente es representada como un dipolo variante en tiempo localizado en el centro de la esfera. Donde  $\Theta(t)$  es el ángulo entre la dirección del vector del corazón  $M(t)$  y  $OA$  es el vector resultante de la unión del centro de la esfera  $O$  con el punto de observación  $A$ ,  $|M|$  es la magnitud del vector corazón, un modelo de este análisis se presenta en la figura 2.5. De manera general, los potenciales medidos en dos puntos en la superficie del torso formarán un vector denominado “Vector de

Derivación" que conecta los puntos A y B, localizados en el brazo derecho o RA, brazo izquierdo o LA y pierna izquierda o LL [14].

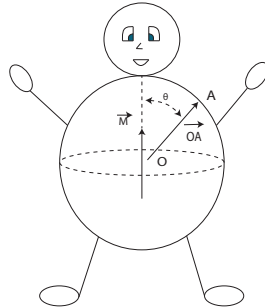


Figura 2.5: Modelo esférico de dipolo y vector corazón [14].

Einthoven es conocido como el "padre de la Electrocardiografía" gracias a sus estudios en la señal cardíaca que en 1924 lo hizo acreedor del Premio Nobel en Fisiología y Medicina al derivar tres mediciones de ECG a partir de los potenciales de superficie derivados del vector corazón  $V_I$ ,  $V_{II}$  y  $V_{III}$ . Estos vectores están asociados con la orientación de las derivaciones y la diferencia de los potenciales es la amplitud del vector corazón  $H$  en el vector de derivación correspondiente en el plano frontal del torso. Entonces,  $L_j = l_j \cdot H$ , y  $V_j = |L_j|$  para  $j = I, II, III$  [12].

$$V_I = \Phi_{LA} - \Phi_{RA}, \quad (2.5)$$

$$V_{II} = \Phi_{LL} - \Phi_{RA}, \quad (2.6)$$

$$V_{III} = \Phi_{LL} - \Phi_{LA} \quad (2.7)$$

Obtenidas a partir de las derivaciones de Einthoven, se encuentran las derivaciones aumentadas (denotadas con una "a") que representan los potenciales en cada una de las extremidades con respecto al promedio del potencial de las dos extremidades restantes.  $aVF$  es la diferencia entre LL y promedio de las derivaciones de los brazos,  $aVR$  es la diferencia entre RA y el promedio de LL y LA, y  $aVL$  es la diferencia entre LA y el promedio de RA y LL [14]. Para una interpretación cardiográfica, los vectores de las derivaciones  $I$ ,  $aVR$ ,  $II$ ,  $aVF$ ,  $III$  y  $aVL$  se asume que dividen el plano frontal del torso en sectores de  $30^\circ$  de acuerdo a la Figura 2.6.



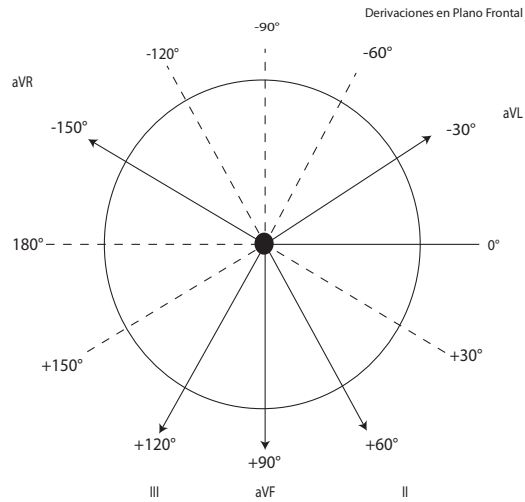


Figura 2.6: Distribución del eje cardíaco en el plano frontal, de acuerdo a derivaciones en extremidades [14].

Cabe destacar que el eje del vector del corazón del complejo QRS es perpendicular a la derivación isoeletrica (una derivación con equilibrio en sus magnitudes) de las derivaciones con respecto a los electrodos, por lo que los cambios significativos en el eje QRS se verán reflejados como problemas cardíacos o características de medición indirectas en la señal (movimientos musculares y respiratorios) [12, 14].

### 2.2.5. Saturación Parcial de Oxígeno (SpO<sub>2</sub>)

El funcionamiento global de las células depende casi en su totalidad del oxígeno, éstas realizan un proceso de respiración denominado aeróbico. La respiración es una de las vías clave por las que las células ganan energía utilizable, ya que es utilizada en la síntesis de Adenosin Trifosfato (ATP), que puede ser usado para realizar los procesos vitales de las células, incluyendo biosíntesis, movimientos o transportación de moléculas a través de la membrana [15].

El encargado del transporte del oxígeno a través del cuerpo es el glóbulo rojo o eritrocito, éste contiene hemoglobina (Hb) la cual sirve como andamio para el transporte de oxígeno y el CO<sub>2</sub>, generando Hemoglobina Oxigenada ( $HbO_2$ ) y hemoglobina carboxigenada ( $HbCO_2$ ), la combinación de la  $HbO_2$  con una alta cantidad de volumen de oxígeno hace que se sature hasta un 97%. La concentración de

oxígeno puede ser medida a través de un método no invasivo basado en absorbancia lumínica siguiendo la Ley de Lambert Beer por medio de la oximetría de pulso [16].

La oximetría se refiere a la determinación del porcentaje de la saturación de oxígeno de la sangre arterial circulante, siguiendo la ecuación (2.8), donde  $[HbO_2]$  es la concentración de hemoglobina oxigenada y  $[Hb]$  en la concentración de hemoglobina desoxigenada.

$$[O_2] = \frac{[HbO_2]}{[HbO_2] + [Hb]} \quad (2.8)$$

Esta saturación es una constante biológica que indica el desempeño de las funciones cardio respiratorias, se mantiene en un valor casi constante con un variación porcentual muy pequeña cercana a un 100 %. La aplicación principal de la oximetría es utilizada para el diagnóstico de anomalías cardíacas y vasculares, así como la monitorización en el tratamiento post operatorio, al tener información inmediata y directa con el nivel de oxigenación en la sangre [16].

La medida del grado de saturación de oxígeno en la sangre puede ser medido por un método de espectrofotometría, en donde las concentraciones de las sustancias en una solución son medidas de acuerdo a las atenuaciones relativas de la luz, que las sustancias absorbentes de luz causan a ciertas longitudes de onda. Se utiliza el modelo de Beer-Lambart en el que una luz que atravieza en un medio absorbente a una concentración  $C$  a un grosor  $b$ , la intensidad de la luz  $I$  es relacionada con la intensidad incidente  $I_o$ , donde  $k$  es un coeficiente de extinción y varía de acuerdo a las sustancias y la longitud de onda, también conocida como absorbancia  $A$ .

$$I = I_o^{-kCb} \quad (2.9)$$

Las características de transmisión espectrales de la oxihemoglobina y la hemoglobina reducida pueden ser estudiadas utilizando los espectros visibles e infrarrojo, con longitudes de onda entre 600 y 700 nm, donde los coeficientes de la sangre oxidada y reducida es mayor. Generalmente, la hemoglobina total varía de la persona y del grosor de los cambios equivalentes en tiempo con una periodicidad determinado por la función cardíaca, dichos factores pueden ser anulados al utilizar dos longitudes de onda y sus concentraciones de hemoglobina. [16].

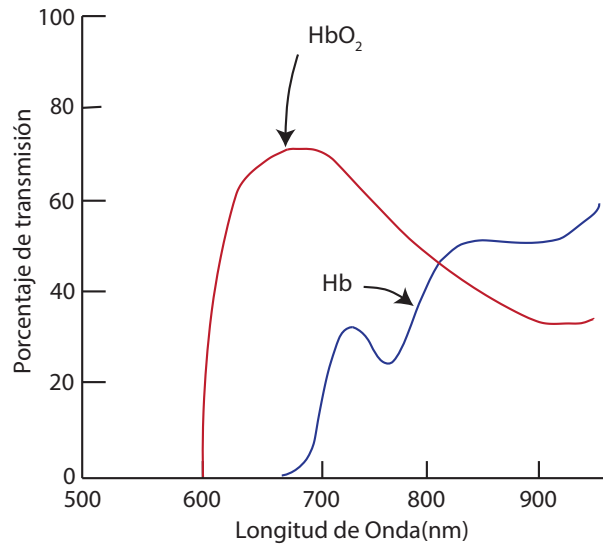


Figura 2.7: Características espectrales de la sangre [16].

Una medida es realizada a una longitud de onda de 650 nm o luz roja y la segunda a 805nm o infrarroja. El canal rojo provee una señal que depende de la cantidad de oxígeno en la sangre y la cantidad de sangre y tejidos en el camino óptico. La señal de canal infrarrojo es dependiente de la saturación de oxígeno y lleva información de la cantidad de sangre y tejido en el camino óptico. La señal de oximetría puede ser medida a través de dos métodos, oximetría de transmisión y oximetría de reflexión.

La mayoría de las absorbancias son fijas y no cambian con el tiempo. Incluso la sangre en los capilares y las venas bajo un estado estacionario metabólico es constante en su composición y flujo al menos por una pequeña cantidad de tiempo. Por lo tanto, al medir solamente la señal cambiante, se mide sólo la absorbancia de la sangre arterial, y hace posible la determinación de la saturación arterial de oxígeno (SaO<sub>2</sub>). La absorción de las longitudes de onda de la luz que incide en un tejido vivo puede ser medido por un foto sensor [16].

La salida del fotodiodo tiene una señal cruda representada en la Figura 2.8, hay una señal que representa la absorción de la luz roja y otra de la luz infrarroja. La señal de AC es generada por el movimiento pulsátil de la sangre arterial, mientras que la señal de DC es producida por aquellas absorciones no pulsátiles del tejido. La saturación de oxígeno en la sangre es estimada de acuerdo a un índice de las absorbancias de la luz roja con las absorbancias de la luz infrarroja:

$$R = \frac{ac660/dc660}{ac940/dc940} \quad (2.10)$$

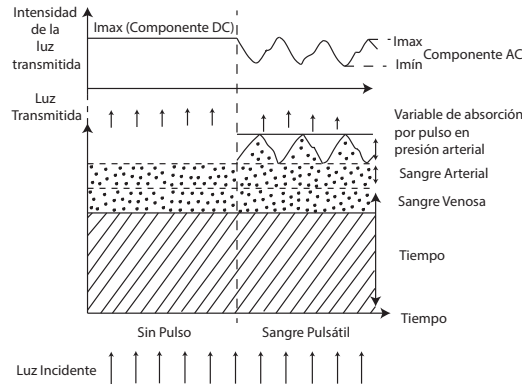


Figura 2.8: Interacción Luz incidente con dedo en medición de oximetría [16].

### Respiración Derivada del ECG (EDR)

Con procesamiento digital de las señales de ECG y SpO<sub>2</sub>, es posible obtener una medida de respiración derivada de manera indirecta. La señal respiratoria es usualmente obtenida por medio del uso de herramientas como la espirometría, pneumografía o pletismografía corporal. Estas técnicas requieren del uso de dispositivos molestos que pueden interferir con la respiración natural y que no pueden ser manejables en cierto tipo de aplicaciones o en ciertos pacientes [14].

El estudio en conjunto de los sistemas cardíacos y respiratorio ha sido de gran interés para la creación de métodos que permitan la obtención de manera indirecta de la información respiratoria de los pacientes. Es bien sabido que la actividad respiratoria influye directamente en mediciones tales como el ECG, y la señal de pulso cardíaco por SpO<sub>2</sub>. Durante el ciclo respiratorio los movimientos en el pecho y los cambios en la distribución de la impedancia torácica causan la rotación del eje eléctrico del corazón que afecta la morfología de los latidos cardíacos registrados [14]. Dicha rotación eléctrica es causada por el movimiento de los electrodos en relación con el corazón, lo que, aunado a la acción mecánica de la respiración, resulta en la agregación de componentes frecuenciales en el espectro de la señal de ECG y SpO<sub>2</sub>.

A partir de esa observación, se han desarrollado diversos algoritmos para la derivación de la medida de respiración derivada de ECG o algoritmos EDR, los cuales necesitan un alto grado de preprocesamiento para un buen desempeño. Dentro de estos algoritmos se utilizan características propias de las señales de ECG y SpO2 tales como, la amplitud del pico R, el área bajo la curva del complejo QRS, los cambios en la Arritmia sinusoidal del corazón, también conocida como el efecto de la expansión torácica por la frecuencia respiratoria en la medición de ECG, el uso de la transformada de ondoleta discreta y técnicas de computo suave como PCA [17].

### **2.3. Señales Biomédicas, Procesamiento e Instrumentación**

Desde el comienzo de la medicina, los profesionales de la salud han utilizado instrumentos de medición que permitan el diagnóstico de enfermedades y evitar a toda costa la intervención innecesaria, al corroborar con métodos de observación y análisis del estado de sus pacientes. Las señales biomédicas pueden ser encontradas en diferentes niveles en el cuerpo humano, desde sistemas completos como el sistema cardiovascular o nervioso; en entidades específicas como el corazón, hígado, cerebro, o incluso a nivel celular [18].

El objetivo principal de la adquisición de señales biomédicas, es el obtener una representación de un fenómeno fisiológico que otorgue información que será tratada de acuerdo a metas específicas, las cuáles serán establecidas por la persona que analiza y procesa dichas señales y sus características [18].

Dentro de las características que distinguen a las señales biomédicas de otro tipo de señales se encuentra que son derivadas de sistemas dinámicos complejos, que tienen interacciones variadas con otros sistemas biológicos, pueden presentar variabilidad de una persona a otra o en diferentes medidas del mismo paciente y que las señales comunmente pueden presentar ruido por distintas fuentes tanto endógenas como de fuentes externas [18].

Las señales biomédicas deben ser "acarreadas" por algún tipo de energía, normalmente estas fuentes de energía; pueden ser químicas, mecánicas y eléctricas. Dentro

de las señales biomédicas que comunmente se registran en la medicina diagnóstica están: Actividad eléctrica del corazón, músculos y cerebro; Presión Sanguínea; Ritmo Cardíaco; Concentración de un gas en la sangre; Concentración de otros componentes sanguíneos; Sonidos generados por las válvulas del corazón y otros [19].

En la tabla 2.1 se muestra la relación de tipo de energía, variables y la medición resultante en la adquisición de señales biomédicas. Para el desarrollo de la solución descrita en esta tesis se utilizarán medidas obtenidas por medio de energía eléctrica con el uso de señales cardíacas (ECG), y señales externas con el uso de fotoplestis-mografía de pulso (SpO2) de manera instrumental y la frecuencia respiratoria por procesamiento de las señales mencionadas.

<b>Energía</b>	<b>Variables (Fluctuaciones Específicas)</b>	<b>Medidas Comunes</b>
Química	Actividad química y/o concentración	Iones sanguíneos, O <sub>2</sub> , CO <sub>2</sub> , Concentraciones hormonales, y otra química
Mecánica	Posición, Fuerza, Torque, Presión.	Movimiento muscular, Presiones cardiovasculares, Contractividad muscular, Sonidos cardíacos
Eléctrica	Voltaje(energía potencial de portadores de cambio), Corriente (cambio de flujo de portadores).	EEG, ECG, EMG, EOG, ERG, EGG, GSR.
Térmica	Temperatura	Temperatura corporal, Termografía.

Tabla 2.1: Energías aplicables al cuerpo humano, variables y señales biomédicas producidas [19].

### 2.3.1. Procesamiento de Señales Biomédicas

El propósito del procesamiento y análisis de señales en la instrumentación médica es el de extraer la información útil del biopotencial “crudo”, esto mediante la aplicación de operadores matemáticos que pueden ser en dominio continuo o varian-

te en tiempo y en el dominio discreto o basado en muestras. Matemáticamente una operación o proceso, puede ser representado por una función de transferencia  $f(t)$ . cuando una señal  $x(t)$  es procesada por  $f(t)$ , la salida  $y(t)$  es igual a la convolución entre la señal de entrada y la función de transferencia [20].

$$y(t) = \int_0^t f(t - \lambda)x(\lambda)d\lambda \quad o \quad y(t) = f(t) * x(t) \quad (2.11)$$

En el dominio de la frecuencia, la relación matemática entre la señal de entrada y salida donde la señal de entrada es procesada por la función de transferencia  $F$  esta dada por:

$$Y(\omega) = F(\omega)X(\omega) \quad (2.12)$$

donde  $X(\omega)$  es la señal de entrada,  $Y(\omega)$  es la señal de salida,  $F(\omega)$  es la función de transferencia del proceso,  $\omega$  es la velocidad angular  $\omega = 2\pi f$ . La señal de salida es el producto de la entrada y la función de transferencia en el dominio frecuencial, es por eso que el análisis de señales es realizado en el dominio frecuencial [20].

## Filtrado Digital

Un filtro es un sistema de procesamiento de señales, algoritmos o método realizado en hardware o software que es capaz de alterar los componentes frecuenciales de una señal y modificarla en una manera particular, al remover los componentes no deseados que son referidos como ruido o artefactos, esto al separar las señales de acuerdo a sus frecuencias. La mayoría de los filtros realiza esto al atenuar la parte de la señal perteneciente a una o más regiones frecuenciales, un filtro puede ser descrito por su función de transferencia que es dependiente de la frecuencia. Los filtros pueden categorizarse de acuerdo a su tipo de respuesta en: Lineales o no lineales, Estacionarios y no estacionarios, Fijos o adaptativos, Activos o pasivos y estadísticos o determinísticos. [13, 20].

Los tipos básicos de filtros pueden ser clasificados en cuatro categorías: pasabaja, pasaalta, pasa banda y rechaza banda. Un filtro es diseñado basado en especificaciones para su banda de paso, banda de rechazo y su banda de transición, tomando en cuenta su respuesta en frecuencia. La banda de paso del filtro es el rango de frecuencias en las que la ganancia de la amplitud es aproximadamente unitaria, la

banda de rechazo del filtro es definida como el rango de frecuencias sobre el que la magnitud de la respuesta del filtro es atenuada para eliminar los componentes frecuenciales que están fuera del rango de trabajo. La banda de transición es el rango de frecuencias entre la banda de paso y la banda de rechazo, la relación entre la tres se esquematiza en la figura 2.9 [20].



Figura 2.9: Respuesta en frecuencia de un filtro pasa bajas [20]

Los filtros digitales pueden ser clasificados también dependiendo de su respuesta al impulso, en Filtros de Respuesta al Impulso Finita (FIR), y los filtros de Respuesta al impulso infinita (IIR). Un filtro FIR esta descrito por una función en modo discreto de la siguiente ecuación:

$$\begin{aligned}
 y(n) &= \sum_{i=0}^k b_i x(n-i) \\
 &= b_0 x(n) + b_1 x(n-1) + b_2 x(n-2) + \dots + b_k x(n-K)
 \end{aligned}
 \tag{2.13}$$

donde  $b_i$  representa los coeficientes y donde  $K + 1$  representa la longitud del filtro FIR. Al transformar esta ecuación con la transformada Z, se logra obtener una función de transferencia:

$$H(z) = \frac{Y(z)}{X(z)} = b_0 + b_1 z^{-1} + \dots + b_k z^{-k}
 \tag{2.14}$$

A diferencia de los filtros FIR, un filtro IIR, tiene la siguiente función en modo discreto de la siguiente ecuación:



$$\begin{aligned}
y(n) = & b_0x(n) + b_1x(n - 1) + \dots + b_Mx(n - M) \\
& - a_1y(n - 1) - \dots - a_Ny(n - N)
\end{aligned} \tag{2.15}$$

De igual manera que con el filtro FIR, utilizando la transformada Z y obteniendo una función de transferencia:

$$H(z) = \frac{Y(z)}{X(z)} = \frac{b_0 + b_1z^{-1} + \dots + b_Mz^{-M}}{1 + a_1z^{-1} + \dots + a_Nz^{-N}} \tag{2.16}$$

Tras la observación de ambas funciones de transferencia, se puede considerar los filtros FIR como un sistema de lazo abierto y los filtros IIR como filtros recursivos, los primeros con solamente ceros en su función de transferencia y los segundos con polos y ceros que alterarán la estabilidad funcional y los tipos de respuesta que pueden otorgar [21]. La utilización de estos tipos de filtros es primordial para el preprocesamiento de las señales de los biopotenciales y la derivación indirecta de la señal de respiración utilizada en la presente tesis.

### 2.3.2. Adquisición de Señales Biomédicas

Para la adquisición de los datos en la presente Tesis se utilizaron dispositivos comerciales de adquisición, para el ECG se utilizó el sensor AD8232 de Sparkfun Electrónica y para la adquisición de la oximetría se utilizó el sensor MAX300102 de la empresa MAXIM INTEGRATED.

#### **Sensor ECG: AD8232**

El sensor AD8232 (véase figura 2.10) adquiere la señal cardíaca y genera una salida analógica con la medición de la actividad eléctrica del corazón, contiene arreglos de amplificadores operacionales para la adquisición de los intervalos PR y QT; está diseñado para extraer, amplificar y filtrar biopotenciales en presencia de condiciones de ruido. Utiliza tres electrodos de entrada como se especificó en el triangulo de Einthoven, RA (brazo derecho), LA (brazo izquierdo) y RL (pierna derecha) [22] .

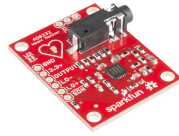


Figura 2.10: Tarjeta de Adquisición sencilla de ECG Sparkfun Electronics [22].

### Sensor SpO<sub>2</sub>: MAX30102

El sensor MAX3000102 es utilizado para el pulso cardíaco y la oximetría de pulso en las personas, incluye LEDs internos y fotodetectores, elementos ópticos, electrónica de bajo ruido y rechazo de luz ambiental. Fue diseñado para ser utilizado en dispositivos móviles y vestibles, debido a su operación de naja potencia, usa una comunicación basada en protocolo I2C, genérica para distintos tipos de sensores y tiene una frecuencia de muestreo y corriente a los LEDs programable para el ahorro de energía [23].

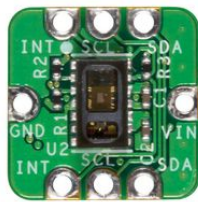


Figura 2.11: Tarjeta de Adquisición de saturación de oxígeno [23].

## 2.4. Inteligencia Artificial (IA) y aprendizaje automático (Machine Learning)

Se ha establecido que la IA tiene cuatro grandes objetivos que son: modelar, formalizar, programar e implementar máquinas que sean capaces de realizar tareas cognitivas y científico-técnicas de una manera más eficiente y completa con descripciones poco claras, incompletas, imprecisar y con alto grado de dudas y errores potenciales debido a su complejidad [24].

La IA tiene su comienzo en los años 60, donde se tenía la hipótesis de que en algunos años de desarrollo iba a ser posible sintetizar los procesos cognitivos y conseguir "Inteligencia general en las máquinas". El desarrollo de los algoritmos de IA inician con la búsqueda de predicciones de manera artificial en fenómenos biológicos, siendo incorrectos desde un inicio, debido a la implementación como sistemas estáticos y poco maleables, generando las dos aproximaciones dominantes en la IA: la simbólica o representacional y la del conexionismo situado [24, 25].

La primera haciendo énfasis en el uso de conceptos, hechos y reglas en la representación de conocimiento para la toma de decisiones y la segunda en el uso de conceptos de más bajo nivel semántico, considerando la inteligencia como un método de adaptación superior y realizando máquinas capaces de interactuar con un entorno real concreto.

Otro punto de vista en la IA como Ciencia e Ingeniería es que nbbn debe tener un carácter instrumental y sus objetivos deben ayudar a comprender los procesos neurofisiológicos, cognitivos y sociales, así como también prolongar los analizadores humanos y complementar sus deficiencias. A partir de este punto de vista y tomando en las cuenta las limitaciones tecnológicas comparadas con las capacidades humanas, se han alcanzado resultados en técnicas de representación e inferencia, aprendizaje automático, "visión" artificial y sistemas basados en conocimiento.

La utilización de algoritmos de aprendizaje automático ha permitido la realización de aplicaciones o sistemas "inteligentes", utilizados para la toma de decisiones, basados en algortimos que generalizan ejemplos conocidos utilizando distintos métodos de aprendizaje: supervisado, no supervisado y de refuerzo.

El aprendizaje supervisado utiliza grupos de datos de entrenamiento en donde las salidas o posibles respuestas son conocidas que le darán la capacidad al algoritmo de crear una clasificación en base al aprendizaje obtenido, al predecir la categoría perteneciente al conjunto de datos de entrada, en base a la membresía de dichos datos. En su modelo predictivo, el aprendizaje supervisado permite otorgar una etiqueta de clasificación a un conjunto de datos presentado después del entrenamiento. El aprendizaje no supervisado se usa para explorar la estructura de los datos y extraer información significativa sin la guía del conocimientos de las respuestas del conjunto de entrenamiento. Por último ,el aprendizaje por refuerzo busca el desarrollo

de agentes que mejoren el desempeño basado en las interacciones con el ambiente incluyendo una señal de recompensa [26].

Actualmente se ha avanzado en técnicas de representación lógica formal y el tratamiento de incertidumbre, y en la solución de problemas donde se tiene más datos que conocimiento, con la creación de modelos matemáticos del comportamiento biológico del cerebro, en las Redes Neuronales Artificiales, las cuáles servirán de base para la toma de decisiones en la presente tesis [24].

### **2.4.1. Redes Neuronales Artificiales (RNA)**

Las RNA son utilizadas para resolver problemas cuya formulación es desconocida, ya sea por no ser un modelo causal o por la falta de una representación matemática del fenómeno estudiado. Las RNA utilizan información para derivar parámetros que son relevantes en la diferenciación de grupos. El concepto detrás de las RNA fue formulado a través de hipótesis y modelos de como trabaja el cerebro humano para resolver tareas complejas [25, 26].

El estudio de las RNA data de los años 40, cuando Warren McCulloch y Walter Pitt desarrollaron la descripción del funcionamiento de las neuronas humanas, llegando a una problemática global al momento de modelar la interacción de múltiples capas de neuronas sin tener una solución concreta para el entrenamiento de las redes de neuronas. En 1962, se introduce el modelo del perceptrón, basado en el funcionamiento de la activación de áreas en la retina, usando tres tipos de pesos, para determinar la activación de los receptores de luz, teniendo un tiempo de entrenamiento muy largo. En 1986, D.E. Rumelhart, G.E. Hilton y R.J. Williams, desarrollaron un algoritmo de retropropagación para el entrenamiento de las redes neuronales con múltiples capas [25, 26].

Las RNA forman parte de la subdivisión del aprendizaje automático conocido como “aprendizaje profundo”, el cual puede ser entendido como el grupo de algoritmos desarrollados para el entrenamiento de las RNA de una manera eficiente [26]

Las RNA se conforman de neuronas artificiales interconectadas, las cuáles reciben información y generan información de salida, dicha salida se basa en la interacción de pesos en cada una de las entradas de la RNA y la interacción de las conexiones

de cada una de las neuronas. Las RNA se clasifican de acuerdo al número de capas y de conexiones que éstas pueden tener; un diagrama general de una Red Neuronal Artificial se presenta en la figura 2.12, se denomina multicapa, y cuenta con una capa de entrada, una capa oculta y una capa de salida, el número de capas ocultas podrá variar de acuerdo al fenómeno en estudio [25, 26].

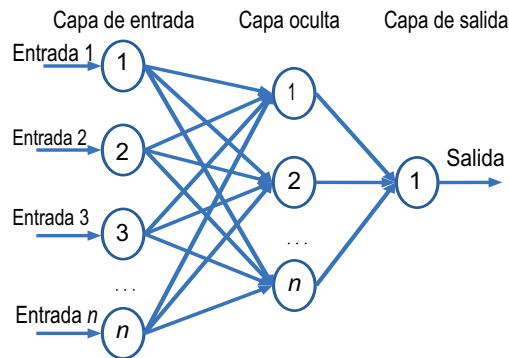


Figura 2.12: Diagrama General de una Red Neuronal Multicapa, con  $n$  entradas y una salida [25].

Las primeras redes neuronales tenían una estructura de solamente dos capas, posteriormente se desarrollaron RNA multicapas que se clasificaron en redes de adelante, la cual calcula los pesos para determinar la salida de un nodo, que se pasará como entrada del nodo siguiente, la red de retropropagación utiliza patrones para determinar generar los vectores de entrada. Un nodo es la representación de una neurona biológica, el rango de activación de un nodo indica el valor que éste puede asumir, en algunos casos estos valores pueden ser binarios o estar normalizados; la salida de un nodo es determinada de acuerdo a una sumatoria y un offset que se utilizan como argumento en una función de activación de acuerdo a la ecuación:

$$y = f\left(\sum_{i=1}^n (w_i x_i - \theta)\right) \quad (2.17)$$

Donde el nodo tiene  $n$  entradas,  $w_i$  es el peso asociado a la entrada,  $\theta$  es el offset o el valor límite interno y  $f$  es definida por el algoritmo, las cuales pueden ser escalones, rampas, sigmoides, entre otras [25].

## 2.5. Trabajos Relacionados

Alrededor del mundo se ha tratado el problema planteado anteriormente con la aplicación de la Inteligencia Artificial en la Ingeniería Biomédica. En la Universidad Simón Bolívar de Venezuela en la tesis “Entrenamiento evolutivo de redes neuronales dinámicas para la detección de apnea-bradicardia en neonatos prematuros”, se realizó la clasificación de Apnea-Bradycardia con el uso de Redes Neuronales Dinámicas, con datos de la señal de Electrocardiograma (ECG) de 32 pacientes neonatales prematuros, y además realiza una combinación de las Redes Neuronales con Algoritmos Genéticos Especializados por medio de un aprendizaje supervisado [27].

Por otro lado, en la Universidad Tecnológica de Delf, en la tesis “Automated Detection of Central Apnea in Preterm Infants”, se realiza la clasificación de los casos de apnea neonatal tomando datos de señales de ECG, Saturación Parcial de Oxígeno ( $SpO_2$ ) e Impedancia Torácica, usando una base de 10 pacientes neonatos y un clasificador de tipo Máquina de Soporte Vectorial [28]. La Universidad de Manipal en la India, en la tesis “Machine Learning Techniques for Neonatal Apnea Prediction”, se utiliza un total de 229 registros de ECG y se agregan factores demográficos, datos maternos y factores fisiológicos de acuerdo al desarrollo del paciente, que son introducidos a un árbol de decisión y finalmente a una Máquina de Soporte Vectorial, para generar una predicción y una clasificación de los episodios de apnea [29].

En la Universidad de Zaragoza en conjunto con la Universidad de Oxford, Monasterio et al. realizaron la implementación de un clasificador de falsas alarmas para la apnea neonatal, basado en características de señales de Electrocardiograma, Saturación Parcial de Oxígeno, Impedancia Torácica, Respiración Derivada de ECG, con un total de 20 características en los cinco minutos posteriores a la detección de un evento de desaturación ( $\%O_2$ ), en total se calcularon 1616 eventos de 27 admisiones neonatales en la Unidad de Cuidados Intensivos Neonatales, se dividieron los pacientes en dos grupos, uno de entrenamiento y otro de validación, implementando una Máquina de Soporte Vectorial, obteniendo una sensibilidad del 86 %, una especificidad del 91 % y una exactitud del 90 % en el grupo de validación [30]

## Capítulo 3: Análisis y Diseño

### 3.1. Introducción

Para la realización del sistema descrito en la presente tesis, se llevó a cabo el análisis y diseño del sistema en general siguiendo modelos genéricos de Ingeniería de Software e instrumentación; dentro de los temas a revisar en el actual capítulo se encuentran: la metodología del proyecto en general, análisis y diseño del sistema, siguiendo el modelo 4+1 vistas, el modelo de datos y la instrumentación utilizada con los componentes del sistema.

### 3.2. Metodología

Dentro de la metodología seguida para el desarrollo del sistema, se establecieron tres fases. La fase I corresponde al proceso de planeación del proyecto en general, tomando en cuenta el estudio del estado del arte, la especificación de las señales a utilizar, la adaptación de los sensores presentados en el capítulo 2, así como también el diseño de la instrumentación y arquitectura del sistema en general. La fase II incluye el análisis y diseño del prototipo, construcción del prototipo y la implementación del sistema. Por último la fase III, incluye la definición, ejecución y evaluación del protocolo de pruebas, véase fig. 3.1.

### 3.3. Análisis del Sistema

Como parte del análisis realizado para el diseño de la solución presentada en la presente tesis, se desarrollaron los diagramas de las figuras 3.2 y 3.3, donde se detalla

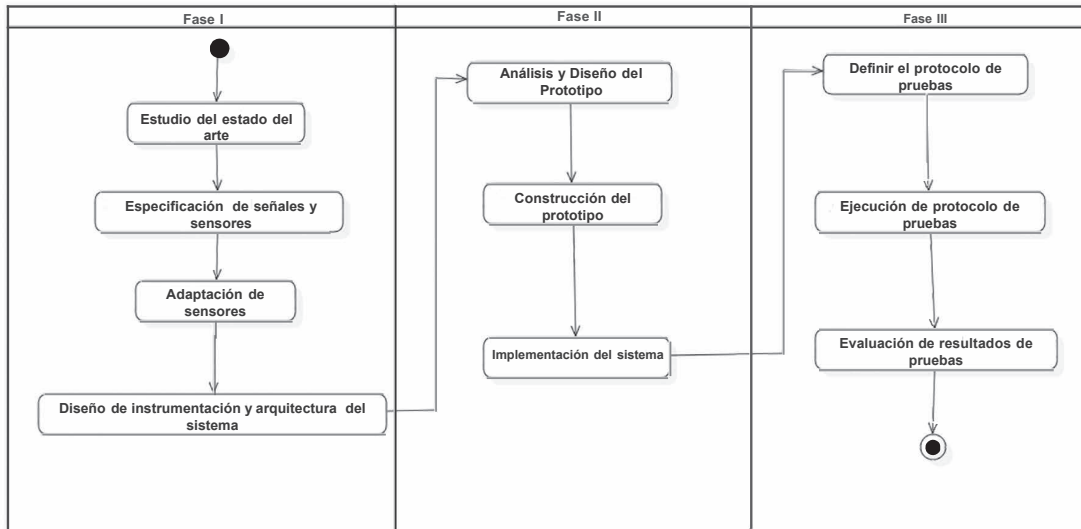


Figura 3.1: Metodología Propuesta para Sistema de Monitorización y Detección de apneas MIBB [Fuente Propia].

el contexto a seguir en la implementación del sistema en un nivel general o nivel 0 y un nivel más específico o nivel 1 respectivamente.

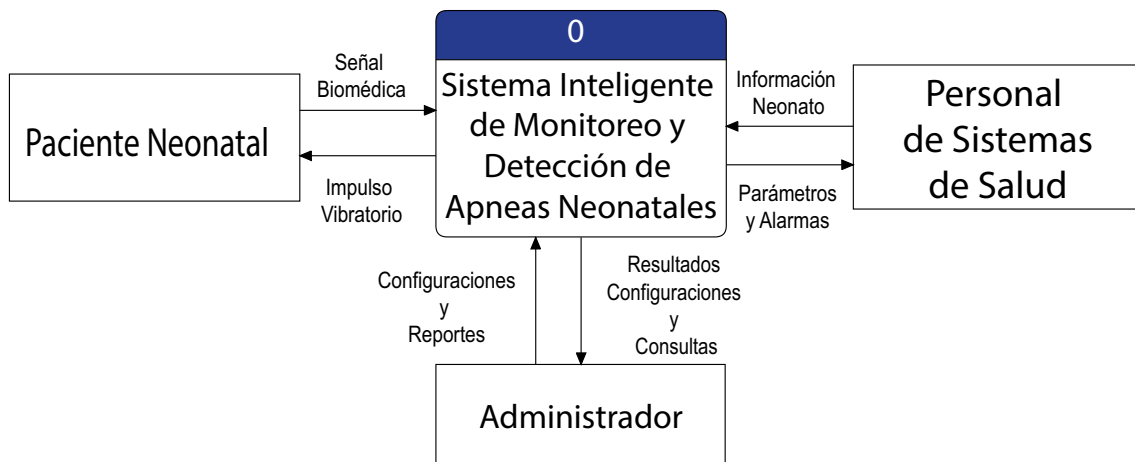


Figura 3.2: Diagrama de contexto nivel 0 sistema de detección de apneas neonatal [Fuente Propia].



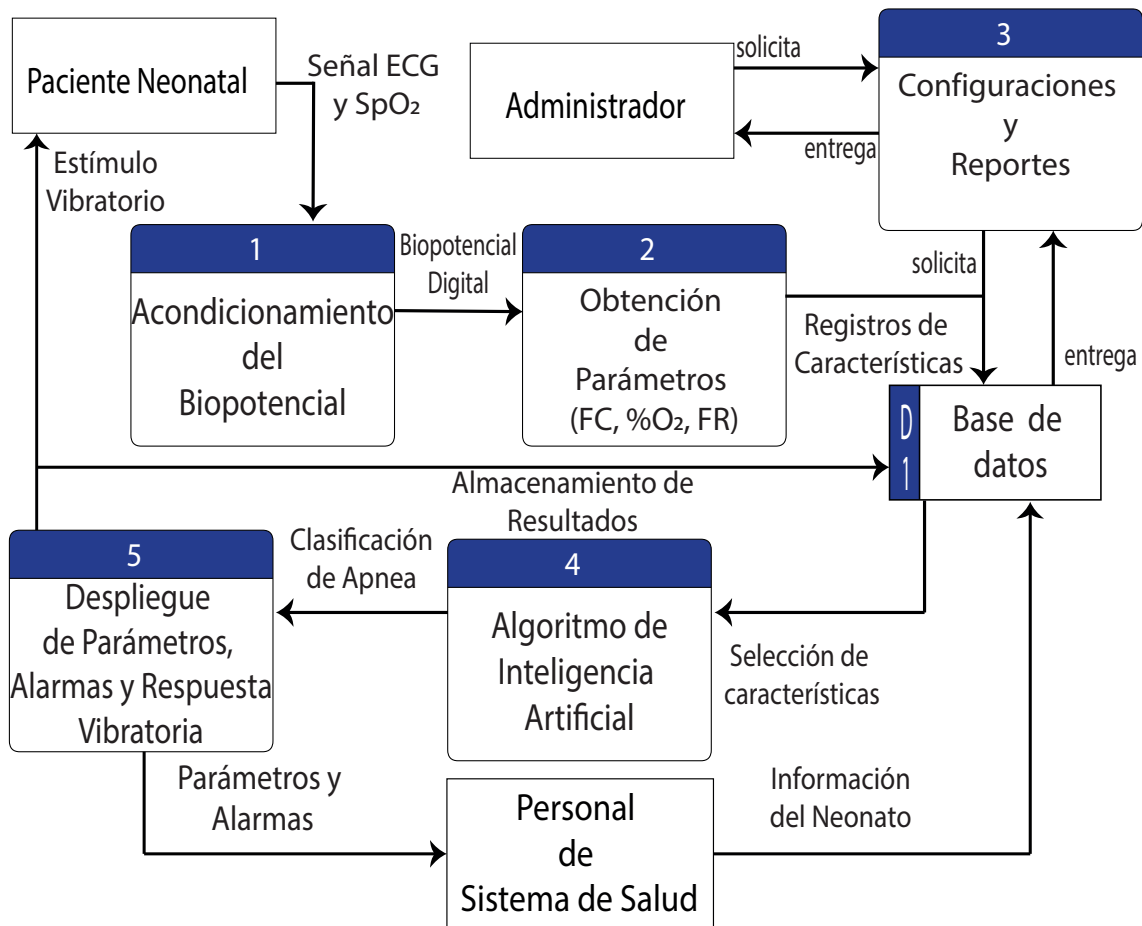


Figura 3.3: Diagrama de contexto nivel 1 sistema de detección de apneas neonatal [Fuente Propia].

### 3.4. Diagrama de Casos de Uso

Al analizar los requerimientos a satisfacer por parte del sistema, se realizó un diagrama con los casos de uso mostrado en la Fig. 3.4. Se especificaron un total de 4 casos de uso principales (Login, Catálogos, Detección y Reportes) con sus casos de uso secundarios; se realiza la descripción del proceso interno de cada uno de los casos así como los actores que intervienen en cada uno y el flujo alternativo.

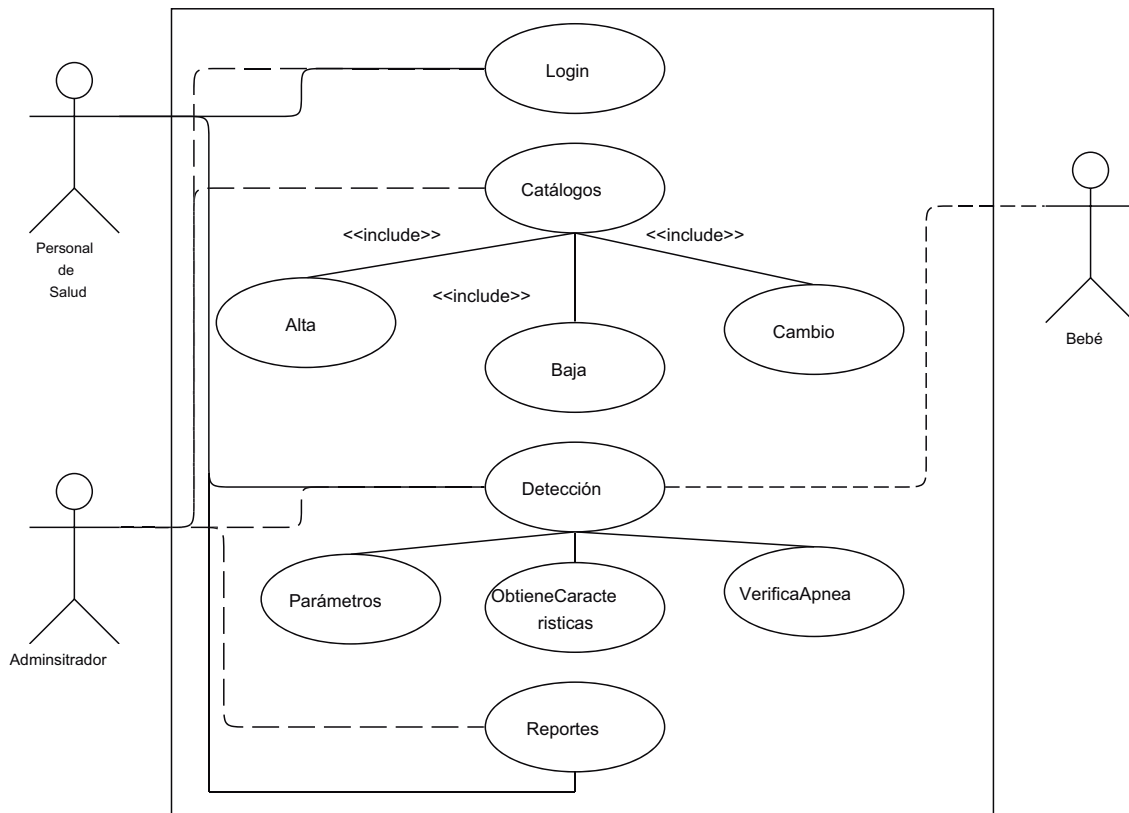


Figura 3.4: Diagrama de casos de uso del sistema de detección de apneas neonatales [Fuente Propia].

### 3.4.1. Caso de Uso 1: Login

<b>Caso de uso:</b>	<b>Login</b>
<b>Actores:</b>	Usuario Administrador
<b>Descripción:</b>	Sirve para acceder al sistema de acuerdo a perfiles de usuario.
<b>Flujo normal de eventos:</b>	<ol style="list-style-type: none"> <li>1. El usuario rellena las credenciales de inicio de sesión y selecciona la opción continuar.</li> <li>2. El sistema valida las credenciales y despliega la ventana principal.</li> </ol>
<b>Flujo Alternativo de eventos:</b>	<ol style="list-style-type: none"> <li>1. Si las credenciales del usuario no coinciden con las registradas en los catálogos, el sistema mostrará un mensaje de error.</li> </ol>

Tabla 3.1: Tabla descriptiva de caso de uso Login [Fuente Propia]

### 3.4.2. Caso de Uso 2: Catálogos

<b>Caso de uso: Catálogos</b>	
Actores:	Administrador
Descripción:	Sirve para dar de alta, baja y cambio de un catálogo seleccionado.
Flujo normal de eventos:	<ol style="list-style-type: none"> <li>1. El sistema muestra un menú con los catálogos disponibles.</li> <li>2. El usuario selecciona un catálogo.</li> <li>3. El sistema muestra opción de: alta, bajo y cambio.</li> </ol>
Flujo Alternativo de eventos:	

Tabla 3.2: Tabla descriptiva de caso de uso catálogos [Fuente Propia]

#### Caso de Uso Alta:

<b>Caso de Uso: Alta</b>	
Actores:	Administrador
Descripción:	Sirve para dar de alta un elemento en un catálogo seleccionado.
Flujo normal de eventos:	<ol style="list-style-type: none"> <li>1. El sistema muestra un menú con los catálogos disponibles.</li> <li>1.1 Usuarios: nombreUsuario(varchar, 50), tipo de usuario (int)</li> <li>2. El usuario inserta los datos correspondientes.</li> <li>3. El sistema da de alta el elemento</li> </ol>
Flujo Alternativo de eventos:	En caso de cancelar el sistema regresa al menú de catálogos

Tabla 3.3: Tabla descriptiva de caso de uso altas [Fuente Propia].

### Caso de Uso Baja:

<b>Caso de uso:</b>	<b>Baja</b>
Actores:	Administrador
Descripción:	Da de baja un elemento de un catálogo disponible.
Flujo normal de eventos:	<ol style="list-style-type: none"><li>1. El sistema muestra una pantalla con los elementos del catálogo seleccionado.</li><li>2. El usuario selecciona el elemento a dar de baja.</li><li>3. Una vez que el usuario confirme, el sistema da de baja el registro que el usuario seleccionó.</li></ol>
Flujo Alternativo de eventos:	<ol style="list-style-type: none"><li>1. Si no hay elementos para borrar, se muestra un mensaje de error.</li></ol>

Tabla 3.4: Tabla descriptiva de caso de uso baja [Fuente Propia].

### Caso de Uso Cambio:

<b>Caso de uso:</b>	<b>Cambio</b>
Actores:	Administrador
Descripción:	Cambia la información de un elemento de un catálogo disponible.
Flujo normal de eventos:	<ol style="list-style-type: none"><li>1. El sistema muestra una pantalla con los elementos de un catálogo seleccionado.</li><li>2. El usuario selecciona un elemento para cambiar.</li><li>3. El sistema muestra la información ya capturada del elemento, pero en modo edición.</li><li>4. Una vez que el usuario confirma los cambios, el sistema modifica el registro.</li></ol>
Flujo Alternativo de eventos:	<ol style="list-style-type: none"><li>1. Si hay un error en la captura de datos, se muestra un mensaje de error.</li></ol>

Tabla 3.5: Tabla descriptiva de caso de uso cambio [Fuente Propia].

### 3.4.3. Caso de Uso 3: Detección

Tabla 3.6: Tabla descriptiva de Caso de Uso Detección [Fuente Propia].

<b>Caso de uso:</b>	<b>Detección</b>
Actores:	Usuario Administrador
Descripción:	Sirve para realizar la detección de la presencia de apnea utilizando un algoritmo inteligente.
Flujo normal de eventos:	<ol style="list-style-type: none"><li>1. El usuario selecciona la opción de nuevo paciente.</li><li>2. El sistema registra los datos del nuevo paciente y prepara sensores en bebé.</li><li>3. El sistema realiza la configuración de sensores y pruebas de conexión.</li><li>4. El sistema realiza la obtención de parámetros, obtención de características y la detección por medio de la red neuronal.</li></ol>
Flujo Alternativo de eventos:	<ol style="list-style-type: none"><li>1. Si hay un error en la captura de datos, se muestra un mensaje de error.</li><li>2. Si hay un error de conexión, el sistema muestra un mensaje de error.</li><li>3. Si hay variaciones fuera de límite en parámetros el sistema muestra mensaje de error.</li></ol>

### 3.4.4. Caso de Uso 4: Reporte

Caso de uso:	Reporte
Actores:	Usuario Administrador
Descripción:	Sirve para generar un reporte de eventos presentados por el paciente.
Flujo normal de eventos:	<ol style="list-style-type: none"><li>1. El usuario selecciona la opción de reportes- nuevo reporte.</li><li>2. El sistema el sistema despliega los pacientes registrados o la opción de búsqueda.</li><li>3. El usuario selecciona al paciente deseado para generar el reporte.</li><li>4. El sistema realiza la consulta de eventos para el paciente seleccionado y genera un reporte con datos generales, eventos y los parámetros presentados.</li></ol>
Flujo Alternativo de eventos:	<ol style="list-style-type: none"><li>1. Si el usuario selecciona la opción de búsqueda, el sistema mostrará los pacientes de acuerdo a la consulta realizaada.</li><li>2. El usuario es capaz de cancelar la búsqueda, el sistema regresará a la ventana principal.</li></ol>

Tabla 3.7: Tabla descriptiva de Caso de Uso Reporte [Fuente Propia].

## 3.5. Diseño del Sistema

### 3.5.1. Diagrama de Actividades

A continuación se presentan los diagramas de actividad para los casos de uso principales que fueron identificados en la sección 3.4, que son parte de las tareas a realizar por parte del usuario que va a operar el sistema de detección de apneas.

- Caso de Uso Login (Fig. 3.5)
- Caso de Uso Detección (Fig. 3.6)
- Caso de Uso Reportes (Fig. 3.7)

### Caso de Uso Login:

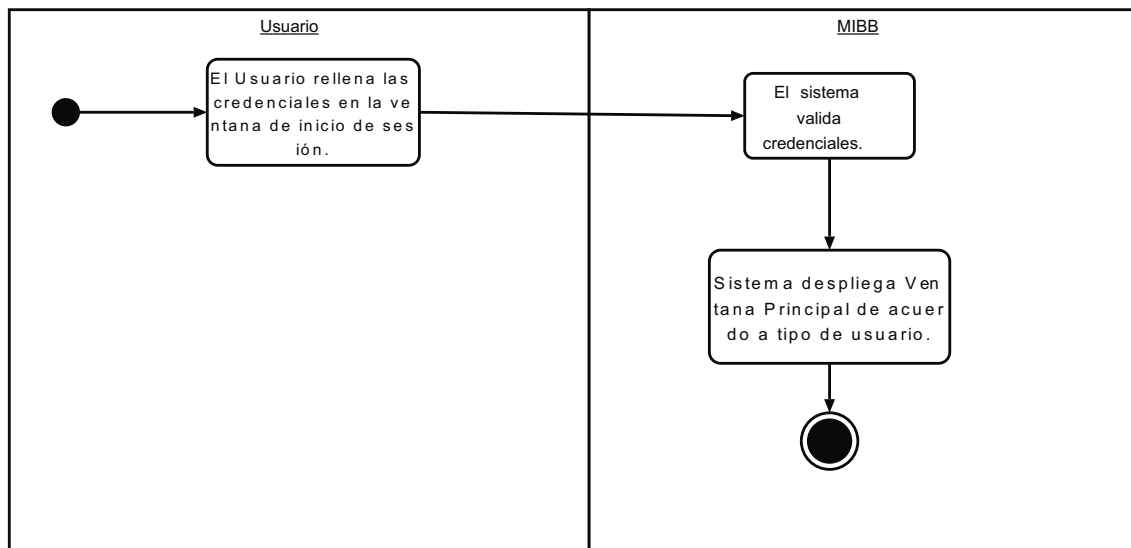


Figura 3.5: Diagrama de actividad para Caso de Uso Login[Fuente Propia]

## Caso de Uso Detección

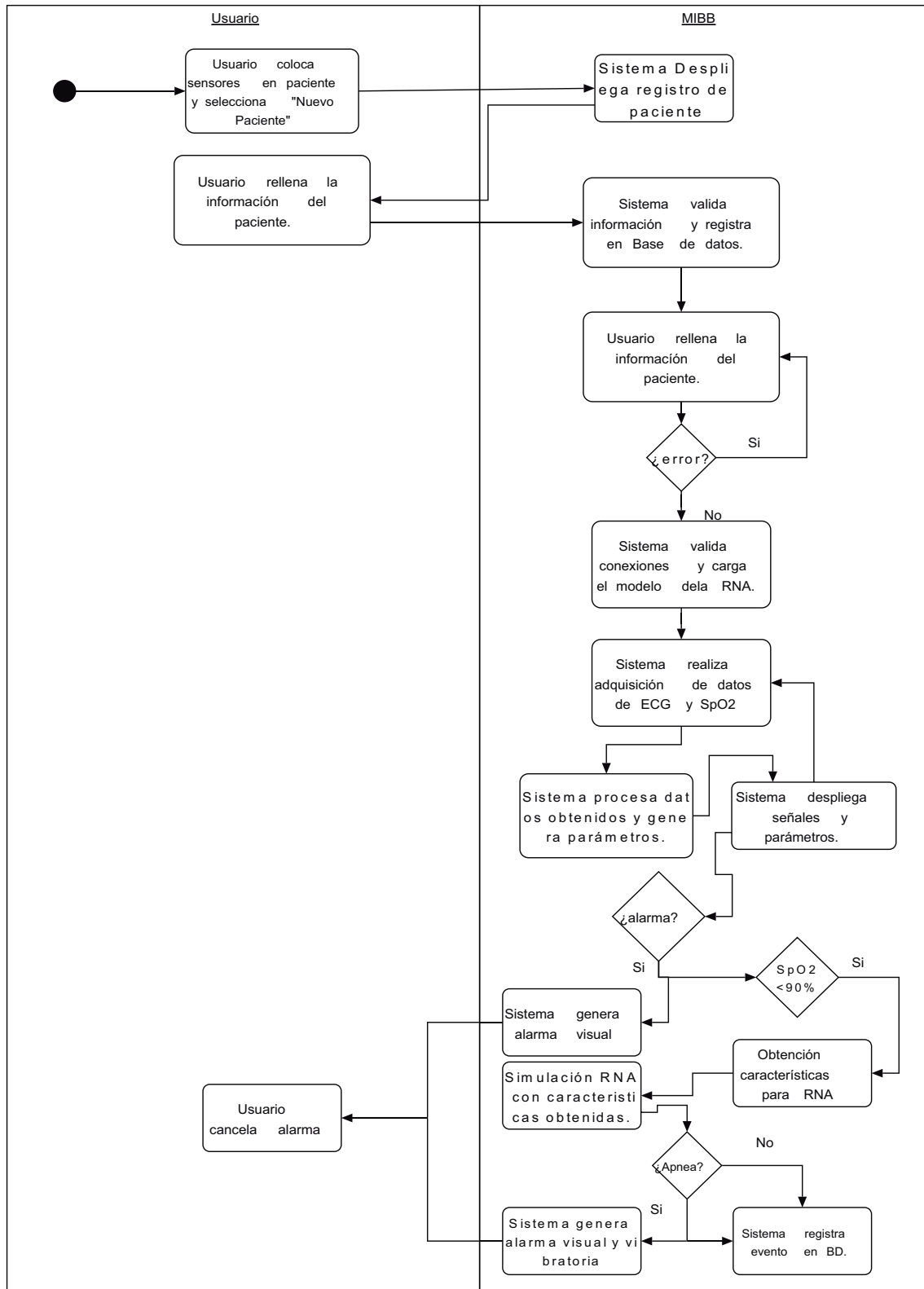


Figura 3.6: Diagrama de actividad para Caso de Uso Detección [Fuente Propia]



## Caso de Uso Reporte

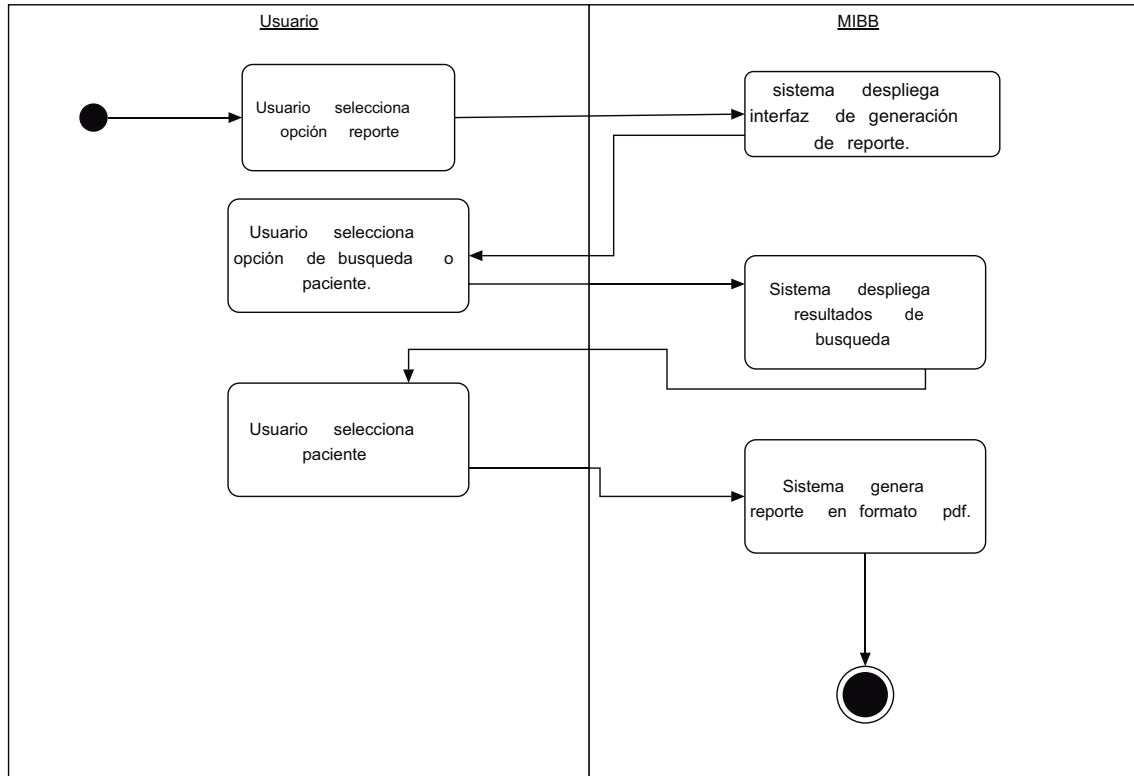


Figura 3.7: Diagrama de actividad para Caso de Uso Reportes [Fuente Propia]

### 3.5.2. Diagrama de Clases

Una vez generado el diseño en Casos de Uso y especificado el funcionamiento general de cada uno de ellos, se presenta el Diagrama de Clases que servirá como modelo para la implementación del sistema como tal en la Fig. 3.8. Cabe mencionar que dicho diagrama constituye la parte lógica de la adquisición y procesamiento de los datos obtenidos, sin incluir el despliegue y funcionamiento de la interfaz gráfica.

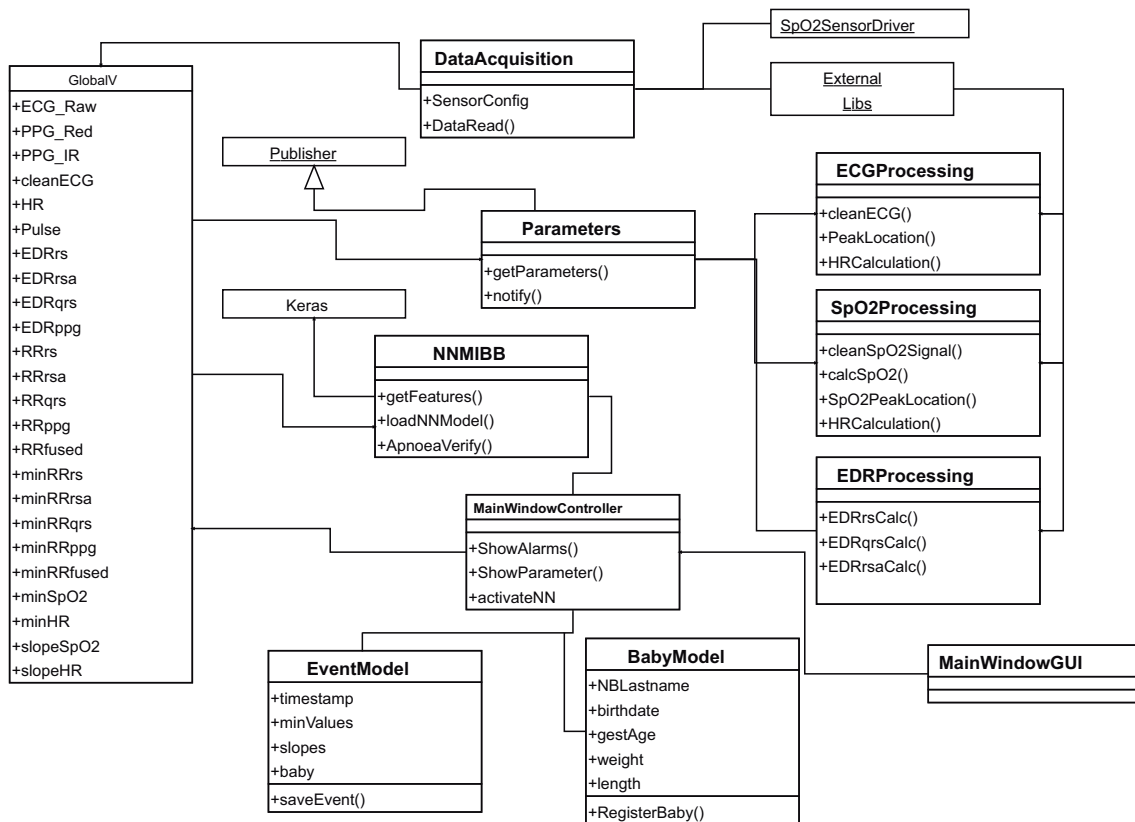


Figura 3.8: Diagrama de clases Sistema MIBB

### 3.5.3. Diagrama de Componentes y Despliegue

En base a las necesidades que se tienen para la implementación y de acuerdo a las especificaciones mencionadas en el capítulo 1, se realizó un Diagrama de Componentes y Despliegue (Fig. 3.9) en donde se entenderá la interacción que tendrá la Raspberry Pi con los sensores y la forma de despliegue y almacenamiento de la información obtenida en el monitoreo.

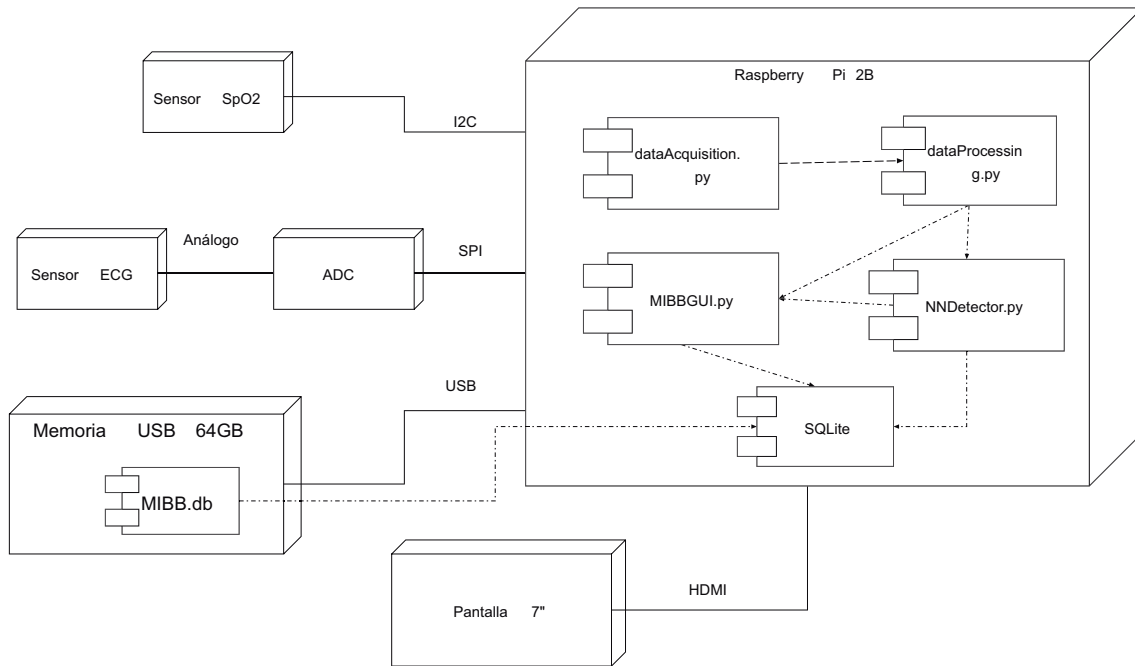


Figura 3.9: Diagrama de Componentes y Despliegue Sistema MIBB

### 3.6. Modelo de Datos

El almacenamiento de la información se realiza utilizando el manejador SQLite, siguiendo en una base de datos que sigue el Modelo ER presentado en la fig. 3.10 para el funcionamiento general del sistema.

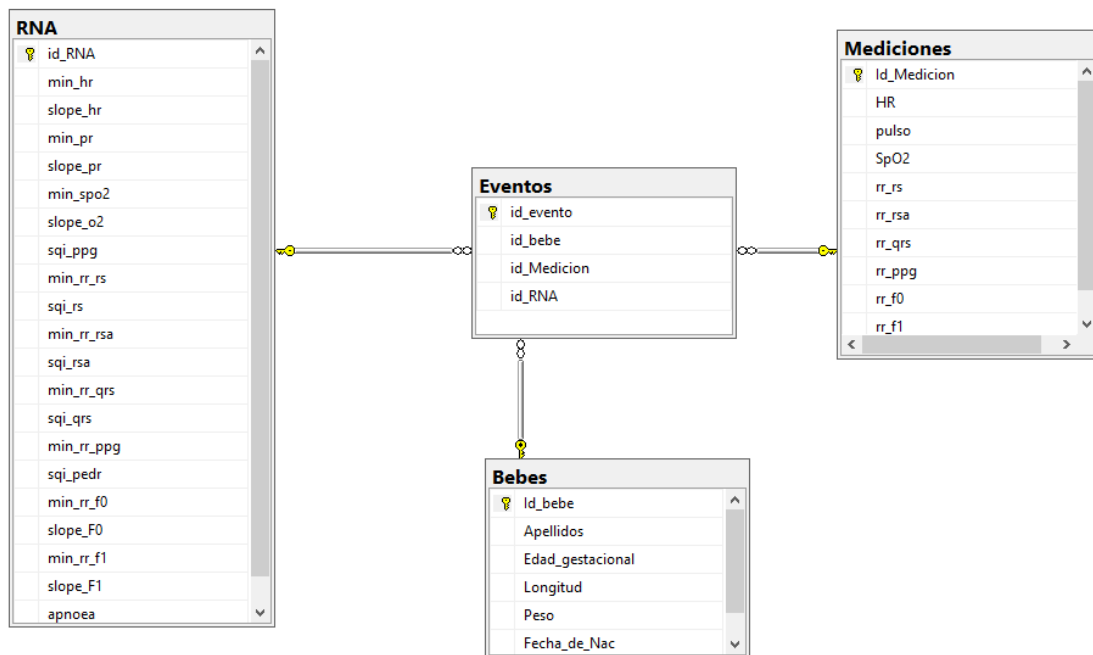


Figura 3.10: Diagrama Modelo de Datos Sistema MIBB. [Fuente Propia]

### 3.7. Arquitectura Propuesta

Finalmente se presenta la arquitectura del sistema de acuerdo a la Fig. 3.11, la cual se conforma de tres módulos: el Módulo de Lectura, el Módulo de Procesamiento y el Módulo de Resultados.

- **Módulo de Lectura:** Contiene el control de la instrumentación de los sensores neonatales no invasivos, configuración de los sensores, verificación de conexiones y la adquisición de las señales crudas de electrocardiograma y saturación de oxígeno.
- **Módulo de Procesamiento:** En este módulo se realizan diversas tareas, acondicionamiento de la señal, obtención de parámetros de monitorización, obtención de características a utilizar dentro del algoritmo clasificador inteligente, aquí se aloja el proceso de conexión y registro en la base de datos.
- **Modulo de Resultados:** contiene los resultados de salida de la etapa de proce-

samiento con datos en forma numérica y vectores datos para el despliegue en pantalla, también contiene la generación de alarmas preestablecidas en los parámetros y la respuesta vibratoria para el intento de interrupción del episodio de apnea.

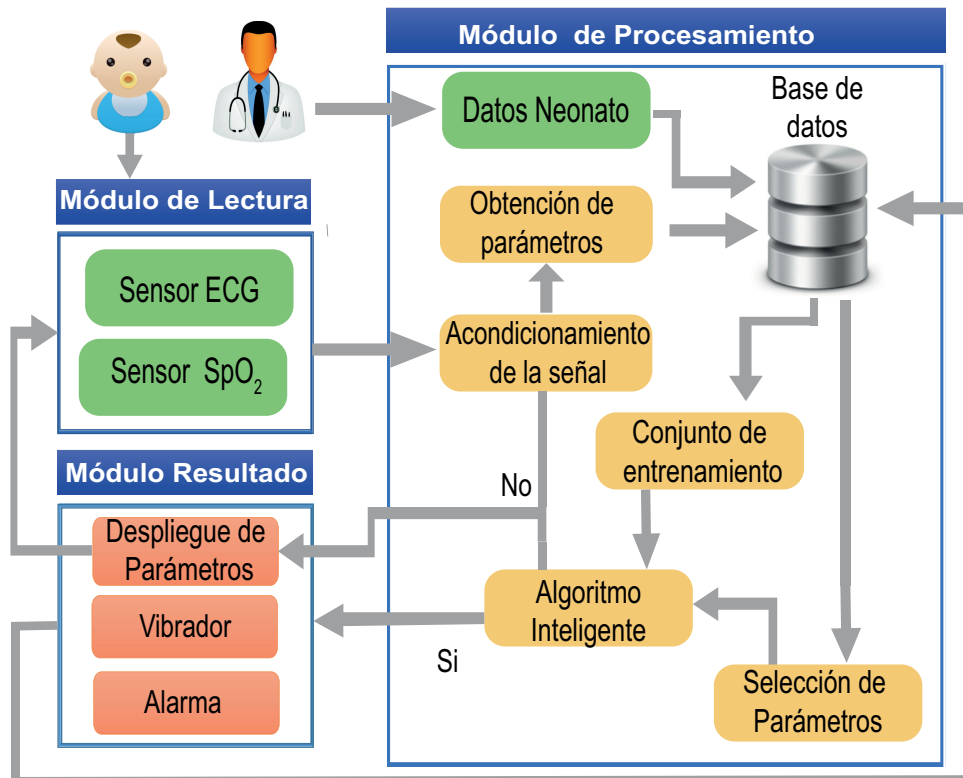


Figura 3.11: Arquitectura de Sistema MIBB [Fuente Propia]

# Capítulo 4: Implementación del sistema y Análisis de Resultados

## 4.1. Introducción

En el presente capítulo se desarrollará la implementación en hardware y software para el sistema MIBB; dicho sistema fue desarrollado en un entorno embebido utilizando una Raspberry Pi 3, la cual, funje como central de procesamiento para el funcionamiento general del sistema, y que tiene las conexiones de hardware necesarias para la adquisición de las señales por medio de los sensores mencionados en el estado del arte, y la conexión a una pantalla táctil para el despliegue de la información.

El Raspberry Pi funciona con un sistema operativo Raspbian que es una imagen de Linux desarrollada específicamente para gestionar el uso de las distintas capacidades de conexión por medio de la utilización de los pines de propósito general, los cuáles funcionan como interfaces de entrada o salida para la conexión del Raspberry con otros periféricos para el desarrollo de soluciones tecnológicas embebidas [31].

## 4.2. Instrumentación del prototipo

Los componentes descritos en el análisis y diseño del sistema fueron ensamblados de acuerdo al diagrama presentado en la figura 4.1. En la figura 4.2 se puede observar la implementación en físico del prototipo conectado de acuerdo al diagrama electrónico, para la conexión del circuito con los pines de propósito general del Raspberry Pi se utilizó un adaptador T-cobbler Plus desarrollado por la empresa

Adafruit.

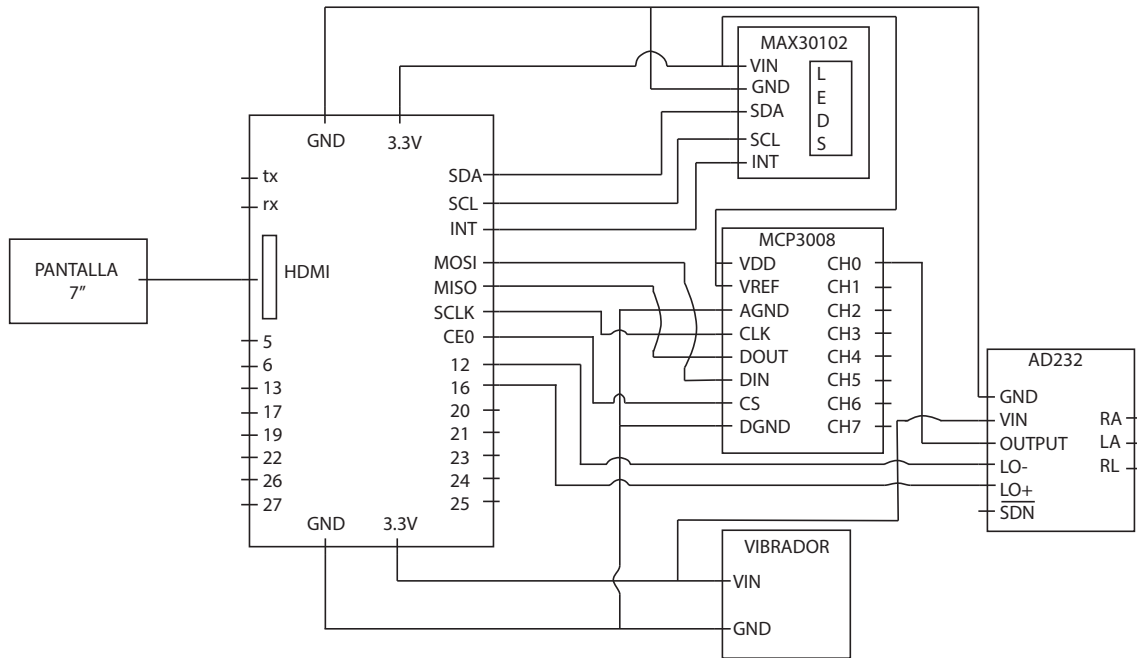


Figura 4.1: Diagrama electrónico de conexión de sensores [Fuente Propia].

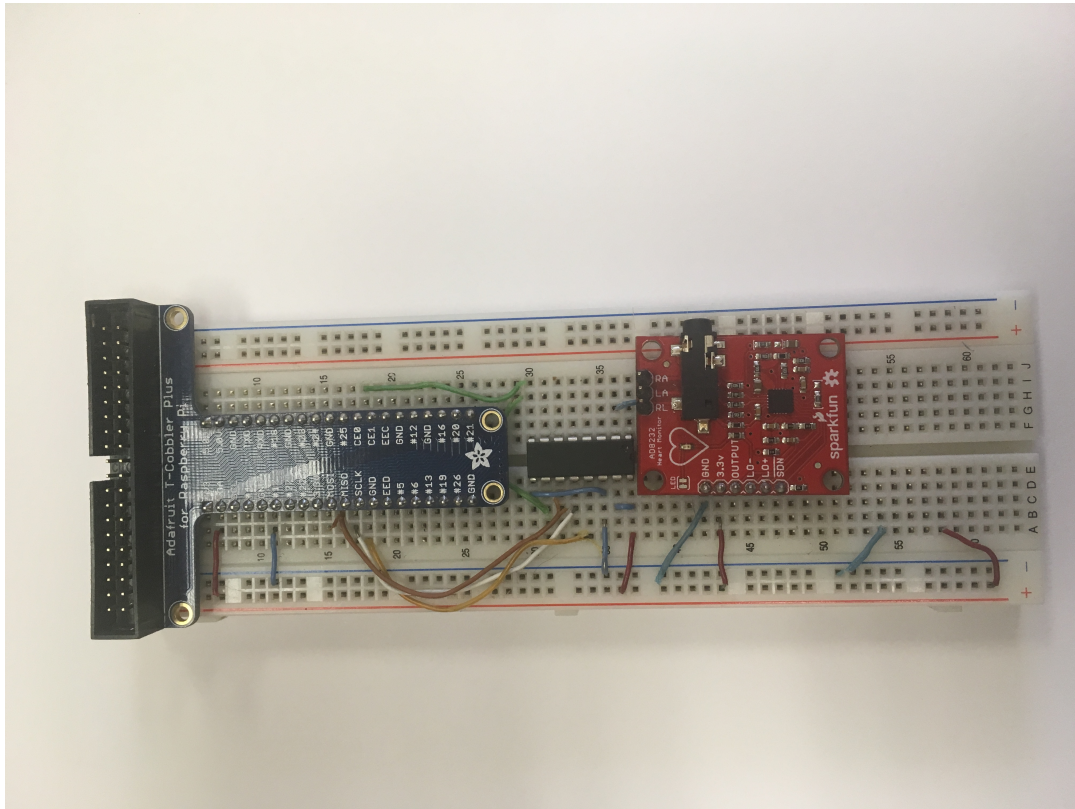


Figura 4.2: Implementación electrónica circuito de adquisición de sistema MIBB.

### 4.3. Adaptación de los sensores

Debido al tipo de paciente, en el cual se utilizará el sistema MIBB, es necesaria la adaptación de los sensores a utilizar para la adquisición de las señales, se realizaron adaptaciones para evitar el uso de electrodos convencionales que causan daños en la piel de los neonatos debido al adhesivo que contienen, se desarrolló un dispositivos vestible y un dispositivo capaz de medir la saturación parcial de Oxígeno en conjunto con la respuesta vibratoria.

#### 4.3.1. Sensor vestible ECG

Dada la problemática presentada en la Unidad de Cuidados Intensivos Neonatales, en donde el personal refería que al momento de retirar los electrodos convencionales, se causaba una lesión en la piel de los neonatos debido al pegamento de los



electrodos. Se optó por buscar una opción de implementación del sensor cardíaco, que causara la menor reacción posible en el bebé y que fuera reutilizable.

Para el cumplimiento de esa meta se desarrolló una camisa capaz de adquirir las señales de ECG, utilizando electrodos secos basados en el trabajo de Bowstra et al [32]. En la figura 4.3 se observa la conformación de los electrodos desarrollados; para la producción de los electrodos textiles se utilizó tela conductiva, listón conductivo e hilo conductivo, todos conformados de hilos de plata para una mejor conducción. El prototipo final de adquisición de datos se muestra en la figura 4.4 teniendo tres almohadillas de adquisición conectadas a tres pines para la obtención de las derivaciones cardíacas bipolares RL, RA, y LA. Los resultados de la adquisición de señales con electrodos convencionales y secos se presenta en la figura 4.5.



Figura 4.3: Implementación de electrodo seco para adquisición de señales cardíacas.



Figura 4.4: Implementación final de dispositivo de medición vestible para adquisición de señales cardíacas neonatales.

El resultado obtenido a partir de la implementación se observa en la figura 4.12, en línea punteada identificada como integral ECG y la señal filtrada identificada como ECG Filtrado.

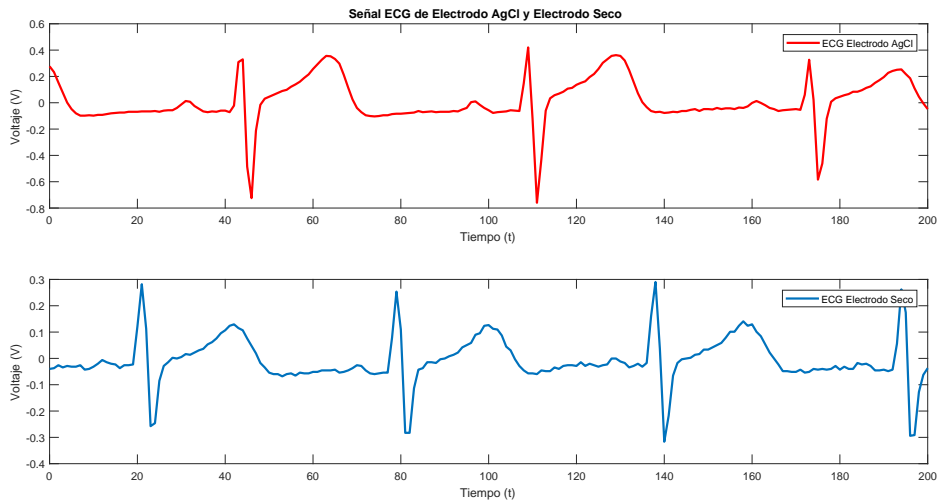


Figura 4.5: Comparación señales captadas mediante utilización de electrodos convencionales AgCl y electrodos secos.

### 4.3.2. Sensor de saturación de Oxígeno

Debido al tamaño del sensor, fue posible el diseño de un estuche para el sensor MAX30102 y el dispositivo vibratorio, mismo que fue diseñado de manera digital e impresa en 3D para su utilización, en la figura 4.6 se observa el diseño del estuche implementado.

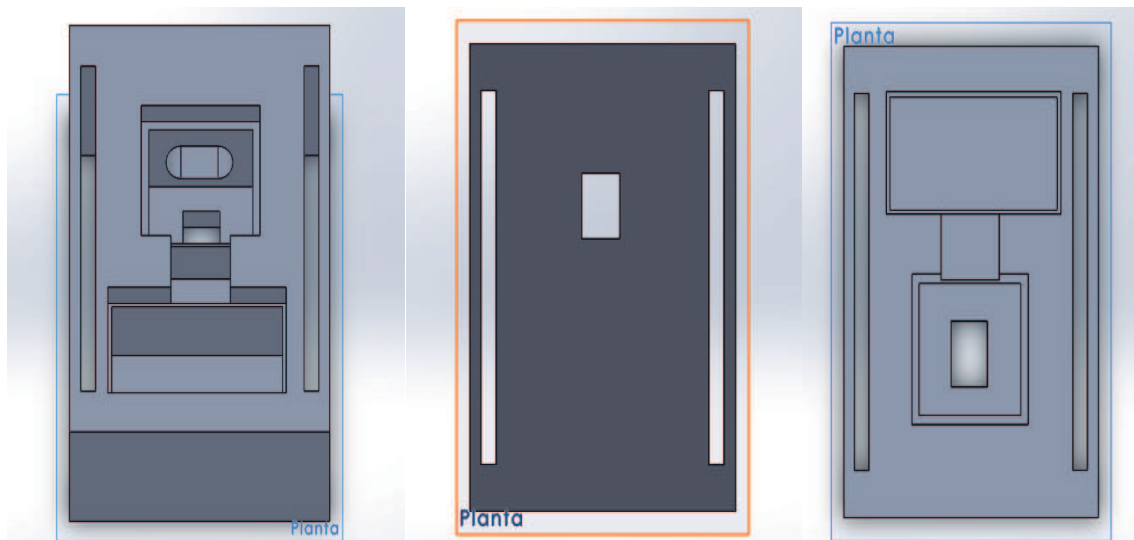


Figura 4.6: Diseño estuche sensor MAX30102 con espacio para vibradores, modelo de impresión 3D.

## 4.4. Adquisición de señales

En la presente sección se abarcará la descripción de la implementación realizada para la adquisición de las señales utilizadas en el desarrollo del sistema MIBB, se describirán las características de los sensores utilizados, así como la implementación en software para la utilización de los sensores en el entorno de la Raspberry Pi.

### 4.4.1. Sparkfun AD8232, señal ECG

Como se mencionó con anterioridad, la tarjeta AD8232 arroja una señal de salida en voltaje o una salida analógica, es por eso que para poder utilizar las señales obtenidas, se utilizó un convertidor analógico digital MCP3008 de la empresa Microchip, el cual es un convertidor de 10-bits y 8 canales con una interfaz SPI.

La implementación y utilización del convertidor en el entorno Raspbian, se realizó utilizando una librería desarrollada por Adafruit Industries y la declaración de los pines de propósito general en la función configSensor de la clase ECGSensor CLK, MISO, MOSI, CS que serán utilizados para la utilización de la librería mencionada en el código 4.1.

```

1 #Utilizacion de GPIOs de acuerdo a numero de base
2 GPIO.setmode(GPIO.BCM)
3 class ECGSensor():
4 def configSensor(self, SampleRate):
5     #Definicion de pines de GPIO y funcionamiento de libreria Adafruit_MCP3008(CLK,
6     CS,MISO,MOSI)
7     CLK = 11
8     MISO = 9
9     MOSI = 10
10    CS = 8
11    mcp = Adafruit_MCP3008(CLK,CS,MISO,MOSI)
12    print "ECG Ready"

```

Código 4.1: configSensor ECG

#### 4.4.2. Maxim Integrated MAX30102, señal de SpO2

Para la obtención de la señal de Fotopletismografía de Pulso se utilizó el sensor MAX30102, un biosensor para la medición de Saturación Parcial de Oxígeno y Ritmo Cardíaco basado en LEDs, mismo que puede ser configurado para trabajar a distintos modos de trabajo por medio controladores para los LEDs Rojo e Infrarrojo con una alimentación de bajo voltaje y filtrado para la eliminación de ruido ambiental, en la figura 4.7 se muestra el diagrama general de componentes del sensor MAX30102. El sensor MAX30102 utiliza un protocolo de comunicación de tipo I2C el cual es un protocolo universal de sensores en la industria y que es compatible los pines propósito general del Raspberry Pi.

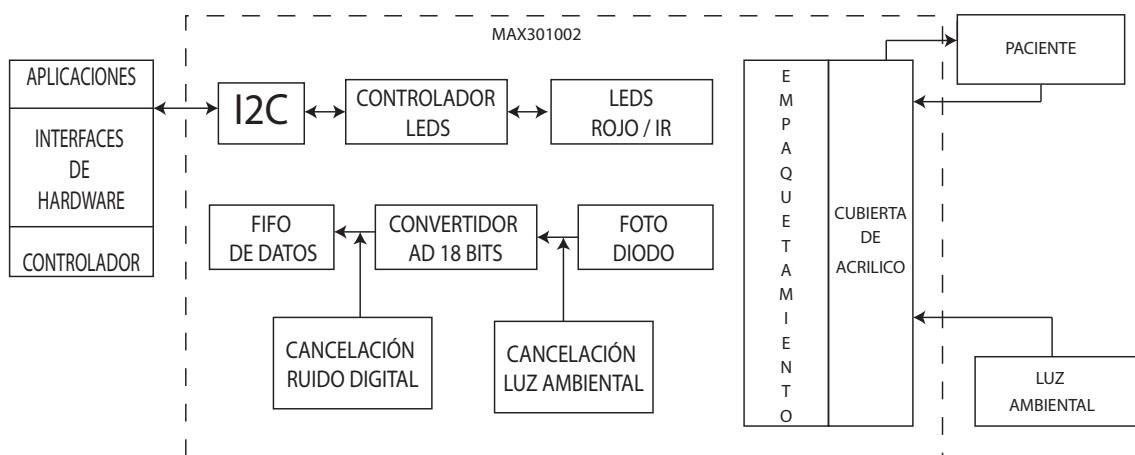


Figura 4.7: Diagrama de componentes sensor MAX30102 [23].

## Comunicación MAX30102/Raspberry Pi

Debido a la falta de implementación de este sensor para trabajo con el Raspberry Pi se desarrolló una librería basada en las existentes para arduino y C++, en lenguaje Python utilizando la información de los registros configurables del mismo sensor siendo los siguientes:

- Configuración de estado.
- Configuración de trabajo.
- Configuración de temperatura de dado.
- Configuración de proximidad.

El sensor MAX30102 registra los cambios de corriente a través de el convertidor analógico-digital de 18 bits, al momento de almacenar una nueva medida de corriente para los LEDs Rojo e Infrarrojo, y se almacena en un bus de datos de tipo FIFO (véase figura 4.8), mismo que puede ser leído de manera individual o utilizando las propiedades de interrupción del dispositivo MAX30102. Para hacer uso del sensor, éste debe ser configurado a través del constructor de clase SpO2SensorDriver, utilizando los métodos set definidos en el controlador del sensor, por defecto el sensor está configurado de la siguiente manera:

- mode='SPO2'
- ledCurrent= 6.4
- sampleAvg = 1
- sampleRate = 200
- pulseWidth = 411
- ADCrange = 16384

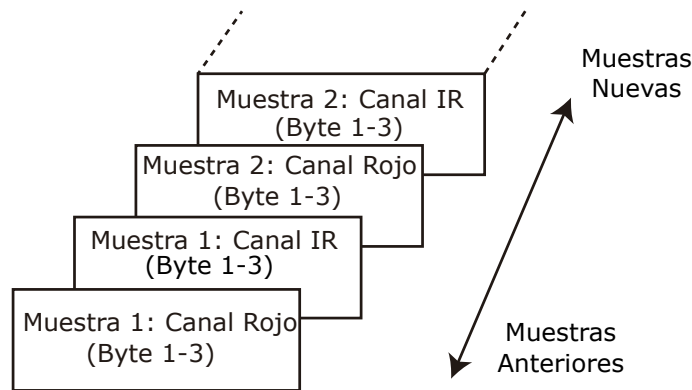


Figura 4.8: Diagrama de Buffer FIFO de datos obtenidos por MAX30102 [23].

## Función ReadSample

La lectura del buffer FIFO de datos, se realiza por medio de la función readSample, código 4.2, en donde se lee el registro de datos de FIFO por medio del protocolo I2C, obteniéndose un arreglo de bytes que son combinados para generar una medición.

```

1 def readSample(self):
2     #Lee Registro de Buffer FIFO y adquiere mediciones
3     Samples = self.i2c.read_i2c_block_data(self.ADDRESS,self.FIFODATAREG,6)
4     HR = 0
5     IR = 0
6     #Combina bytes y valida para obtener un valor final
7     IR = (Samples[0]<<16) | (Samples[1]<<8) | Samples[2]
8     IR = IR & 0x3FFFF
9     HR = (Samples[3]<<16) | (Samples[4]<<8) | Samples[5]
10    HR = HR & 0x3FFFF
11    self.buffer_red = np.append(self.buffer_red,HR)
12    self.buffer_ir = np.append(self.buffer_ir, IR)
13    self.buffer_red = self.buffer_red[-self.max_buffer_len:]
14    self.buffer_ir = self.buffer_ir[-self.max_buffer_len:]

```

Código 4.2: Función readSample de sensor MAX30102.

## 4.5. Acondicionamiento de las Señales

El acondicionamiento de la señal se refiere al pre procesamiento que se le dió a las señales obtenidas, así como a la obtención de las variables indirectas de medición de la respiración, y la preparación del conjunto de entrenamiento para la red neuronal.

### 4.5.1. Procesamiento de ECG

El procesamiento del ECG se basó principalmente en el código realizado por Rose et al. [33] cuyo algoritmo fue desarrollado en C y matlab, y fue adaptado para lenguaje Python; la adaptación se estructura básicamente en tres etapas de acuerdo a la figura 4.9. La limpieza de la señal cardíaca con la función `cleanECG`, posteriormente la localización de los picos QRS y finalmente el cálculo del Ritmo Cardíaco.

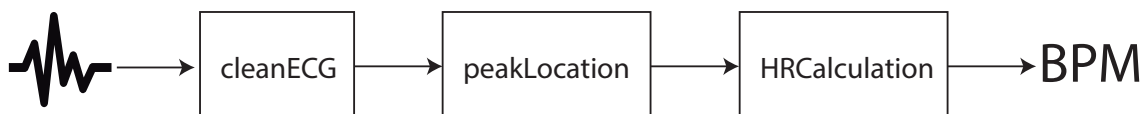


Figura 4.9: Diagrama general de procesamiento de señal cardíaca basado en [33].

#### Función `cleanECG`

La función `cleanECG` engloba el preprocesamiento de preparación para la localización del complejo QRS y sus picos característicos, en la figura 4.10 se muestra el proceso realizado:

1. `getACComponent`: Proceso encargado de la obtención del rizo de la señal al sustraer de la señal original el promedio de la misma, con lo que se eliminará el offset presentado al momento de la adquisición.
2. `eraseBLD`: Proceso encargado de la eliminación del corrimiento de línea, utilizando dos filtros de media con ventanas de 200ms y 600ms.
3. `bandpassFilterECG`: Este proceso está conformado por dos filtros digitales en cascada, un filtro IIR pasabajas de tipo Butterworth con orden 1 a 40Hz y un filtro FIR pasa altas de orden 4 a .5Hz; a partir de la aplicación de esta etapa de filtrado, se generará una señal que será utilizada posteriormente para la detección de QRS.
4. `DerivativeFilter`: Proceso encargado del cálculo de la primer derivada de la señal, realizado por medio de la aplicación de un filtro de tipo Savitzky-Golay con una ventana de 5 muestras, de orden= 2 y derivada = 1.



5. `squaringSignal`: Proceso encargado del cálculo de la cuadrática de la derivada obtenida en el paso anterior, con el fin de la obtención de una señal sin picos negativos.
6. `tempWinIntegraton`: Proceso encargado de la obtención de la integral de la señal al aplicar un filtro de media con una ventana de 100ms variable de acuerdo a la frecuencia de muestreo de la señal generando la señal `cleanECGIntegrated`.

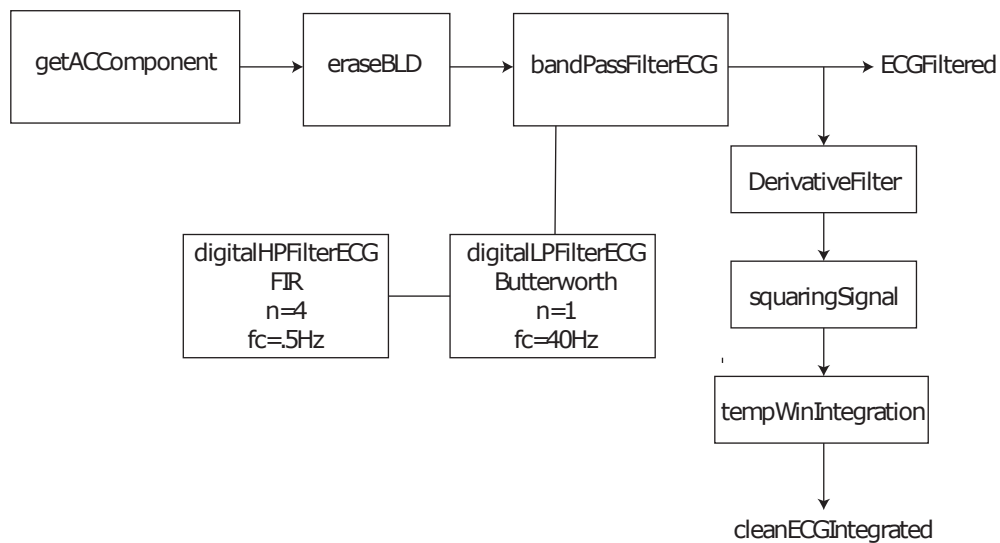


Figura 4.10: Diagrama de preprocesamiento de la señal de ECG basado en Rose et al [33].

### **Función `peakLocation`**

Una vez realizado el preprocesamiento de la señal de ECG, se localizan los picos del complejo QRS con la utilización del algoritmo presentado en la figura 4.11, utilizando las señales obtenidas de la función `cleanECG`: `cleanECGFilteredLP` y `cleanECGIntegrated`.

La función `PeakLocation` se conforma de los siguientes pasos:

1. Definición del límite: Se define el límite a utilizar para la detección de picos, utilizando la señal integrada obtenida en `cleanECG`, el límite está definido por el producto del máximo y el promedio de la señal integrada.

2. Lugares de cruces: Este proceso detecta los cruces del límite establecido generando un vector de registro de posición con un valor binario para denotar en 0 el estado basal, y en 1 el punto de cruce del límite, posteriormente son generados dos vectores a partir de la argumento de las diferencias entre los componentes del vector de registro de posición generando un vector *left* y un vector *right*, cuando la diferencia entre dos registro de posición es igual a 1 y -1 respectivamente.
3. validación L/R: Este proceso se encarga de la validación de la integridad de la detección, al analizar los valores iniciales de L y R para evitar la detección de picos con complejos QRS intermedio o detectados a la mitad.
4. Extracción ECG Temporal: Este proceso se encarga de la extracción de una ventana temporal basado en los argumentos de la señal almacenados en los vectores *left* y *right* que coinciden con el área entre la pendiente positiva y la pendiente negativa de la señal integrada, correspondiendo al complejo QRS de la señal filtrada.
5. Obtención de picos QRS: Este proceso es el encargado de la obtención de los valores y argumentos de los picos del complejo QRS, para eso, se realiza la detección primeramente del punto más alto del complejo, el pico R, posteriormente se utilizan los valores y argumentos de R en conjunto con los vectores L/R para la creación de vectores temporales para Q y S, y terminar con la localización de los puntos máximos o mínimos de los puntos Q y S.

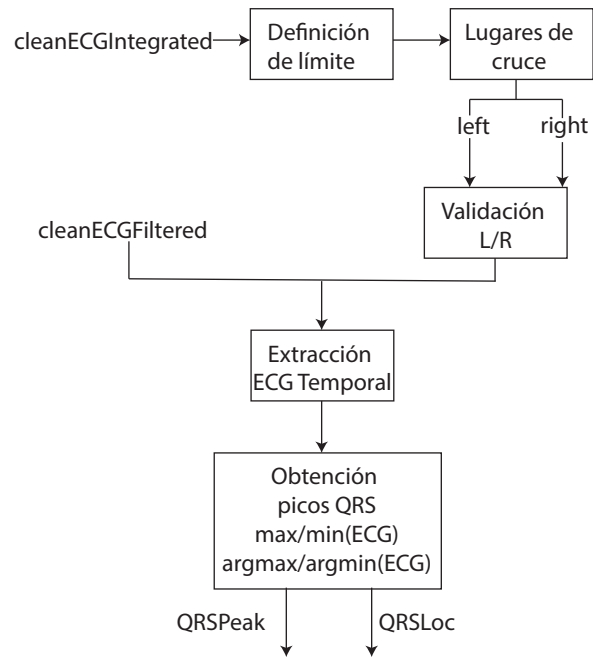


Figura 4.11: Diagrama procesamiento de ECG para la localización de picos del complejo QRS basado en Rose et al [33].

El resultado de la aplicación del algoritmo presentado se puede observar en la figura 4.12 donde los picos R están representados con círculos, los picos Q con cruces y los S con triángulos.

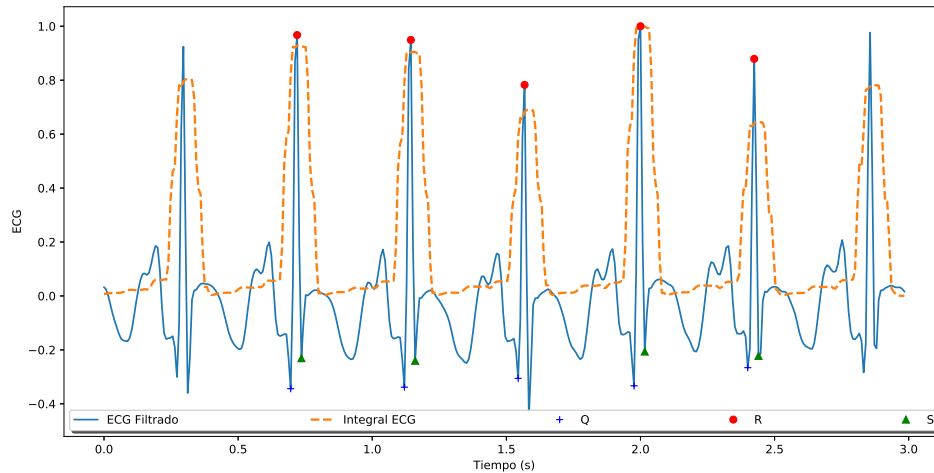


Figura 4.12: Señal resultante de la implementación de preprocesamiento de ECG y localización de picos QRS.

### Función HRCalculation

La función HRCalculation retoma los vectores obtenidos a partir de la localización de los picos del complejo QRS de la señal cardíaca, exclusivamente los picos R con los que, de acuerdo al código 4.3, primeramente se revisa la longitud del vector de localización de los picos R, en donde, al ser mayor a 3 picos (se esperan mas de complejos QRS en un segundo de medición) se crea un vector temporal a partir del cual se generarán intervalos entre los picos R detectados, almacenados en la variable  $RR\_t$ , una vez obtenidos los valores de los intervalos de tiempo entre los picos R consecutivos, se calculará un promedio que será utilizado para el cálculo de la frecuencia cardíaca basado en la cantidad de intervalos presentes en un minuto, de acuerdo a la ecuación 4.1, finalmente, dicha medición es almacenada en un vector HR.

$$HR = \frac{60}{Promedio\ RR} \quad (4.1)$$

```

1 def HRcalculation(self, ECGSignal, fs):
2     #Calcula Ritmo Cardiac
3     if len(globalV.r_Loc) > 3:
4         #crea vector temporal para la medicion de intervalos RR

```

```

5         t = np.arange(0, len(globalV.cleanECGFilteredLP), 1./fs)
6         RR = np.array([])
7         for x in range(len(globalV.r_Loc)-1):
8             RR_t= t[int(globalV.r_Loc[x+1])] - t[int(globalV.r_Loc[x])]
9             RR = np.append(RR, RR_t)
10
11         meanRR = np.mean(RR)
12         HR = 60/meanRR
13         #print "HR: ", HR
14         globalV.HR_vector = np.append(globalV.HR_vector, HR)

```

Código 4.3: Código Función HRCalculation

## 4.5.2. Procesamiento de SpO2

El procesamiento de la señal de SpO2 se basó principalmente en el código existente para Arduino, desarrollado en C. Se tienen cuatro procesos principales que fueron implementados en Python, los cuáles abarcan la limpieza de la señal obtenida de los sensores, el cálculo del porcentaje de Saturación de Oxígeno, y el cálculo del Pulso Cardíaco, mismas que se enlistan en orden cronológico a continuación:

1. cleanSpO2Signal
2. SpO2PeakLocation
3. calcspo2
4. PulseCalculation

### Función cleanSpO2Signal

Este proceso es el encargado de la limpieza de la señal obtenida a partir del método de adquisición para el sensor MAX30102, en la figura 4.13 se muestra el proceso a seguir para la realización de dicha tarea. Primeramente la señal obtenida es normalizada para generar un rango de valores entre -1 y 1, después se le aplica un filtro Butterworth IIR de tipo pasabanda de un orden 4 y con frecuencias de corte en .5-5 Hz, una vez que se da la aplicación del filtro, se utiliza un filtro de mediana con una ventana de 3 muestras para suavizar la señal misma que es invertida para generar la señal característica de la Pletismografía de Pulso. La señal resultante de la limpieza de las muestras obtenidas a partir del sensor MAX30102 se muestran en la figura 4.14.

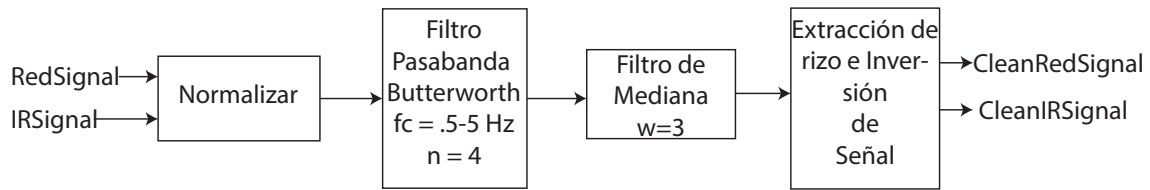


Figura 4.13: Diagrama general de limpieza de señal de SpO2 obtenida de MAX30102.

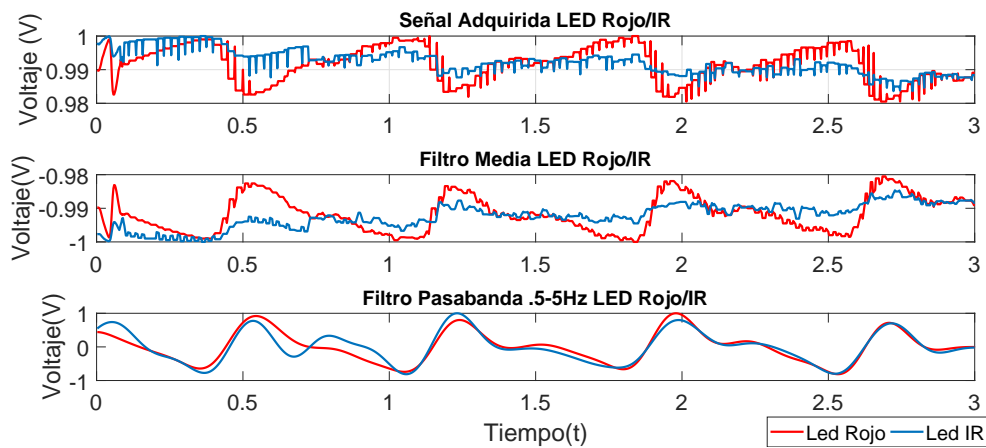


Figura 4.14: Señal resultante de la implementación de preprocesamiento de datos obtenidos de sensor MAX30102.

### Función SpO2PeakLocation

Este proceso es el encargado de la detección de picos en la señal obtenida para el cálculo del Pulso Cardíaco y el almacenamiento de los máximos que serán utilizados en las secciones posteriores. La detección de picos fue implementada basado en el algoritmo desarrollado por Zong [34] y una adaptación de la detección de picos utilizado en el procesamiento de ECG, el diagrama general del procesamiento implementado se muestra en la figura 4.15.

Para la detección de picos sólo se utiliza solamente el componente de la luz IR. Primeramente se aplica un filtro Butterworth de tipo pasa baja con una frecuencia de corte en 16 Hz y de orden 2, posteriormente, se realiza la suma de las pendientes calculadas de una ventana de 128ms de la señal obtenida, calculando la pendiente al diferenciar la muestra actual con la muestra anterior. Un último paso para la

detección de picos es la adaptación del algoritmo de detección de picos en ECG, es decir, la generación de los vectores L y R para la extracción temporal de la señal y la búsqueda de los máximos y el argumento del máximo valor para ser almacenado en las variables peakVal y peakLoc.

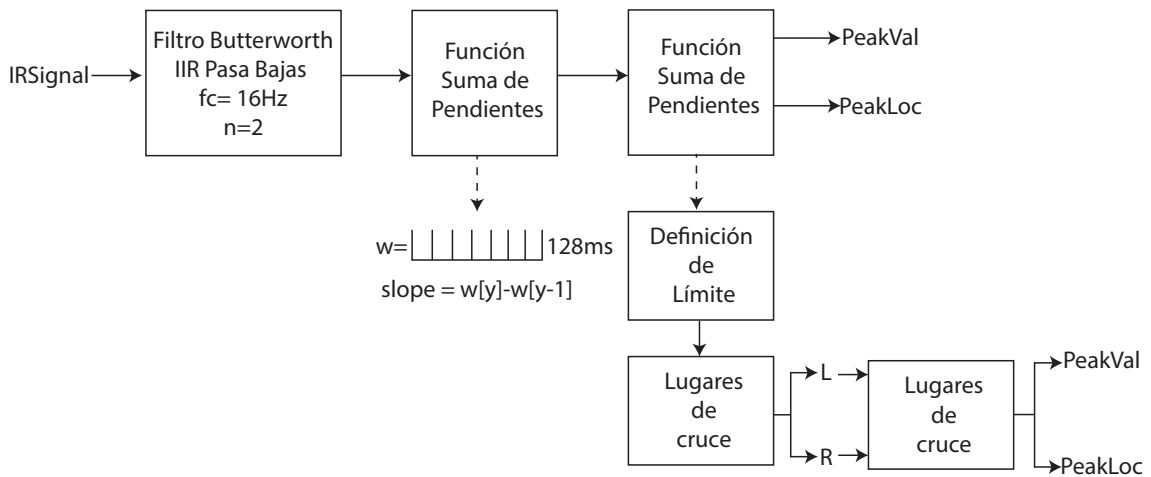


Figura 4.15: Diagrama general de implementación de localización de picos señal de SpO2 basado en Zong [34] y Rose et al [33].

### Función calcSpO2

Este proceso se basa en la información presentada por Santiago López en [15], en donde se especifican dos formas del cálculo del porcentaje de Saturación de Oxígeno, tomándose para la realización de esta medición el índice de índices de la relación entre el rizo o componente AC y el offset o componente DC de las señales de los LEDs Rojo e Infrarrojo, también conocido como índice de absorción (IO) y que sigue la ecuación 4.2.

$$IO = \frac{(AC_{660})/(DC_{660})}{(AC_{940})/(DC_{940})} \quad (4.2)$$

La implementación del cálculo de la SpO2, se muestra en el código 4.4, en donde se obtiene el valor de los componentes AC y DC de las señales Roja e Infrarroja y se almacenan en las variables acRed, acIR y DCRed y DCIR, se realiza la operación descrita en la ecuación 4.2 de la cual se obtiene un promedio y se genera un redondeo a cuatro dígitos, mismo que se almacena en la variable RR.

```

1 def calcSpO2(self, signalRed, signalIR):
2     gP = generalProcessing()
3     #Obtiene componentes de rizo y offset de las Senales Roja e Infrarroja
4     DCRed = gP.getDCComponent(signalRed)
5     acRed = gP.getACcomponent(signalRed)
6     DCIR = gP.getDCComponent(signalIR)
7     acIR = gP.getACcomponent(signalIR)
8
9     #Calcula indice de indices de AC y DC
10    RR = round(np.mean((acRed/DCRed) / (acIR/DCIR)), 4)
11
12    #Calcula valor de saturacion en base a indice de indices
13    spo2Value = 96.545 + 0.616 * RR
14    Spo2Value = int(np.round(spo2Value, 0))
15    globalV.O2sat.append(Spo2Value)

```

Código 4.4: Cálculo de Saturación de Oxígeno

Para poder obtener un valor final del nivel de Saturación de Oxígeno en la sangre, se realizó una caracterización de los valores arrojados por el sensor con la variable RR y un sensor de oximetría comercial (figura 4.16), registrando la medición de diversos sujetos de prueba a los cuales se realizaron diez tomas del valor de RR. A partir de estos datos se realizó una regresión lineal obteniendo un valor específico para la variación de acuerdo a la ecuación 4.3, generando un valor de % de Saturación de Oxígeno.

$$spo2Value = 96.545 + 0.616 * RR \quad (4.3)$$



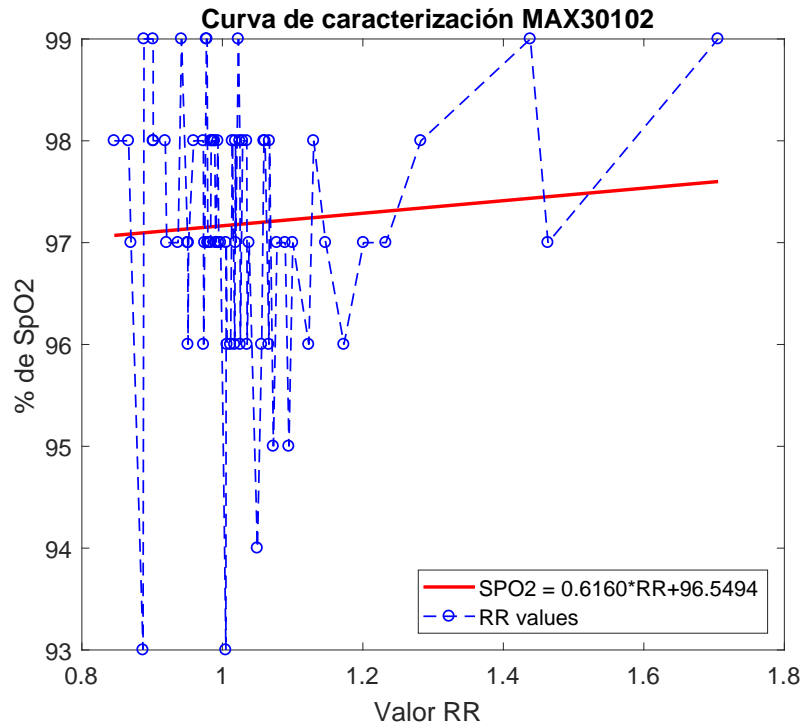


Figura 4.16: Caracterización de valores RR utilizando oxímetro comercial.

## PulseCalculation

Este proceso es el encargado del cálculo del Pulso Cardíaco, utilizando una adaptación del proceso realizado con el procesamiento de la señal cardíaca, una vez que es realizada la detección de picos, se toman los valores de tiempo de esos picos para la derivación de intervalos entre cada uno, y finalmente se calcula cuántos intervalos se esperarían en un minuto, el desarrollo de la implementación de esta parte del algoritmo se presenta en el código 4.5.

```

1 def PulseCalculation(self, signal, sampleFreq):
2     #Realiza detección de picos
3     globalV.spO2SlopeSum = self.slopeSum(signal, sampleFreq)
4     peaksLoc = self.spO2PeakDetection(signal)
5
6     #Analiza picos detectados y calcula intervalos
7     if len(peaksLoc) > 1:
8         t = np.arange(0, len(peaksLoc), 1./sampleFreq)
9         RR = np.array([])
10
11         for n in range(1, len(peaksLoc)-1):

```

```

12         RR_t = t[int(peaksLoc[n+1])] - t[int(peaksLoc[n])]
13         RR = np.append(RR, RR_t)
14
15         #Calcula promedio de intervalos para calculo de pulso cardiaco
16         meanRR = np.mean(RR)
17         pulse = 60/meanRR
18         print pulse
19     return pulse

```

Código 4.5: Cálculo del Pulso Cardíaco.

### 4.5.3. Procesamiento EDR

Usualmente, la respiración es adquirida por diversos métodos, entre los que se incluyen: Espirometría, Pneumografía, o Pletismografía, sin embargo, todas estas necesitan la utilización de dispositivos de apoyo que pueden ser molestos para los neonatos y que interfieren con la señal de respiración, también existen diversos métodos que utilizan los periféricos de otros sensores para la realización de la adquisición de la señal respiratoria, tal es el caso de la respiración por Impedancia Torácica, que utiliza los electrodos del ECG para la adquisición de la señal. Diversos métodos de adquisición de la Frecuencia Respiratoria, han sido desarrollados desde hace varios años tomando en cuenta señales como ECG y Pletismografía de Pulso, ya que la actividad respiratoria tiene influencia directa con la señal electrocardiográfica, en donde el “eje eléctrico” del corazón tiene una rotación que puede ser medida y que modula las señales de ECG en su amplitud pero que no se encuentra en fase con la señal cardíaca [14].

#### Función EDR\_RS

La primera estimación se realizó basado en el trabajo de [35], en donde se especifica el proceso a seguir para la obtención de la estimación de la Frecuencia Respiratoria en base al pico R y S de la señal de ECG, este proceso se presenta en la figura 4.17, en donde una vez obtenidos los valores de los picos R y S, se realiza el cálculo de la diferencia entre los absolutos de los dos valores, obteniendo un conjunto de diferencias que serán interpoladas a través del método de *spline cúbico* utilizando un vector temporal y la localización de los picos R de la señal de ECG.

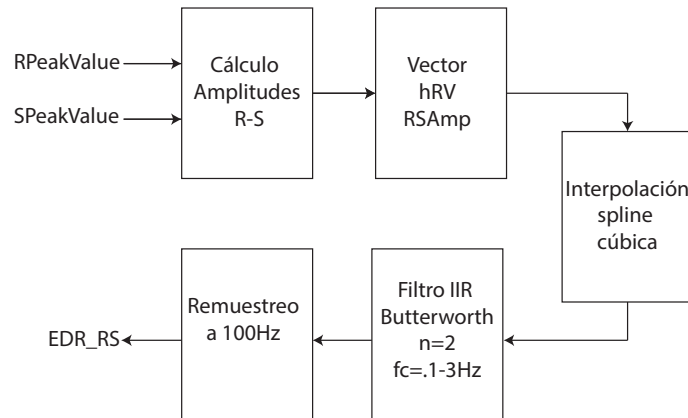


Figura 4.17: Diagrama general de procesamiento para EDR con amplitud RS basado en [35].

### Función EDR\_QRSAREA

Otra de las estimaciones realizadas para la obtención de la Frecuencia Respiratoria, se hizo de manera similar a la estimación explicada en el caso del pico R y S, ésta vez se utilizó el área bajo la curva en el complejo QRS para la amplitud del pulso a ser interpolado basado en [36], se utilizó una interpolación con el mismo método que en el cálculo de  $EDR_{RS}$ ; en la figura 4.18 se esquematiza el proceso a seguir para dicho cálculo.

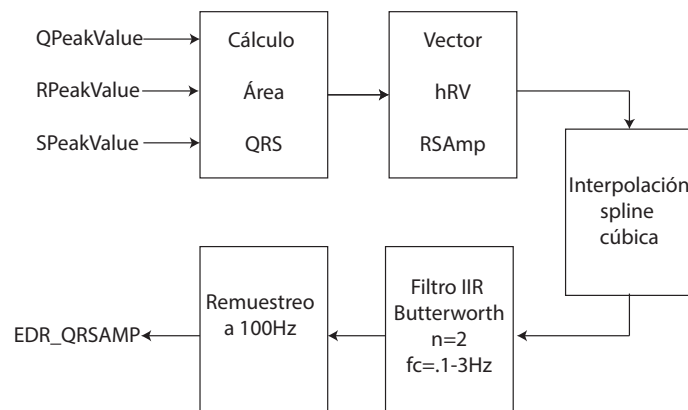


Figura 4.18: Diagrama general de procesamiento para EDR con área de QRS basado en [35].

## Función EDR\_RSA

El término “RSA” corresponde a la estimación de la respiración derivada del ECG, por medio de un fenómeno natural conocido como “Arritmia Sinusal Respiratoria”, en donde el intervalo de tiempo entre los picos R de la señal cardíaca normalmente oscilan de manera periódica, acortándose durante la inspiración y elongándose con la espiración, producido por el reflejo de Bainbridge, que relaciona variaciones en la presión intratorácica con el movimiento pulmonar y el volumen cardíaco de llenado, en donde al inspirar, la presión dentro del tórax disminuye y el retorno venoso aumenta, lo produce un estiramiento en la aurícula derecha y acortando los intervalos RR de la señal cardíaca, al espirar el proceso es inverso, por lo que se observa una duración mayor en los intervalos RR [14].

Para observar la acción de dicho reflejo en la señal electrocardiográfica es necesario realizar un tipo de filtrado específico a la señal directa, basado en [37], de acuerdo a lo mostrado en la figura 4.19 donde se utiliza un filtro de tipo FIR pasa-banda a frecuencias de paso de .2 a .4 Hz de un orden 20 y que se aplica a la señal original de ECG para obtener la señal envolvente equivalente al movimiento torácico de respiración.

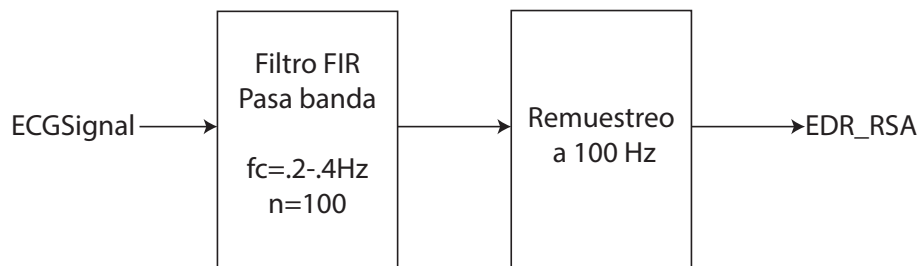


Figura 4.19: Diagrama general de procesamiento para EDR filtrado por Arritmia Sinusal Respiratoria (RSA) basado en [35].

## Función EDR\_PPG

Para la derivación de la respiración de la señal de la Oximetría de Pulso se aplicó un criterio similar a la función *EDR\_RS*, aplicándose el proceso descrito en la figura 4.20.

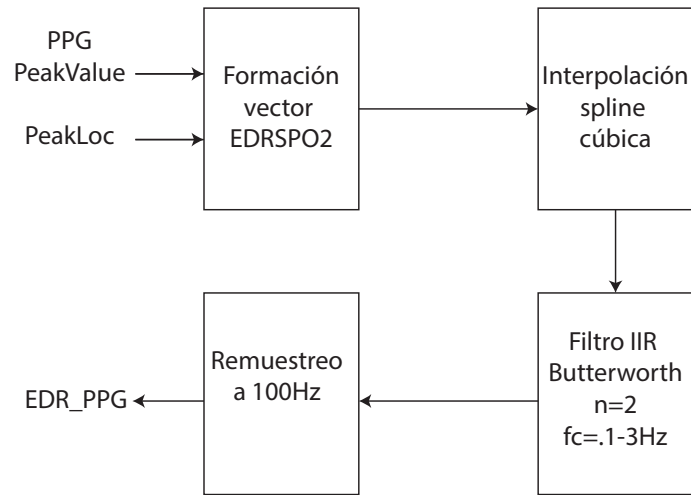


Figura 4.20: Diagrama general de procesamiento para EDR con picos de SpO2 basado en [35].

El resultado de los algoritmos descritos anteriormente es mostrado en la figura 4.21.

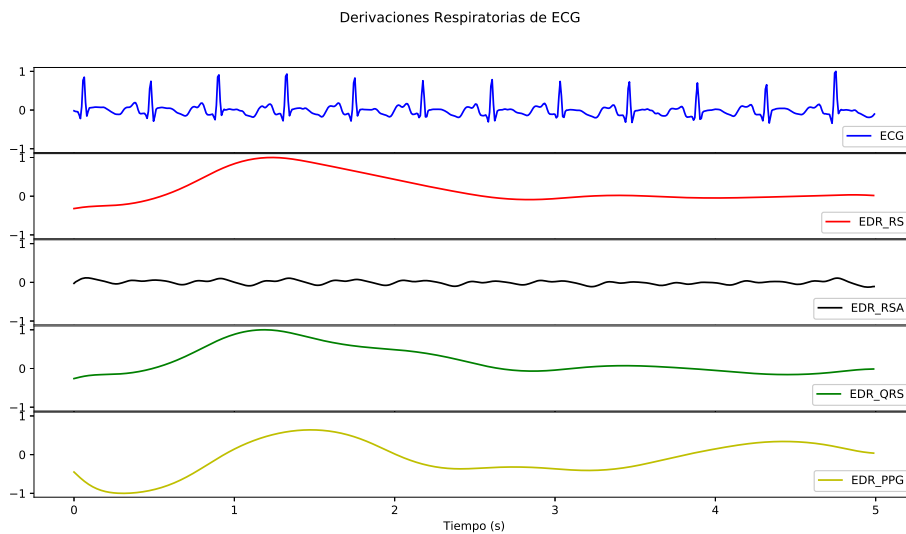


Figura 4.21: Señales Respiratorias derivadas de ECG [Fuente Propia].

### **Función RR\_Calculation**

Para el cálculo de la frecuencia respiratoria se siguió la metodología propuesta por Nemati en [38], en donde se utiliza un modelado autorregresivo de análisis espectral

(AR) aplicado a las estimaciones respiratorias obtenidas, siguiendo los siguientes pasos:

1. Decimado de la señal respiratoria a 5Hz.
2. Remoción de offset de la señal.
3. Selección de orden del modelo AR (orden 8 para este caso).
4. Selección de límites respiratorios, 4-70 respiraciones por minuto para neonatos.
5. Aplicación del modelo AR, utilizando la metodología de Arburg, para la obtención de los polos de la ecuación.
6. Exclusión de los polos que no entren dentro de los límites.
7. Selección de polos por encima del 95 % de la magnitud del polo con la magnitud máxima.
8. Selección del polo con la menor magnitud.
9. Cálculo de la frecuencia asociada con dicho polo como la Frecuencia Respiratoria.

### **Función $RR\_Fused$**

La función  $RR\_Fused$  abarca la fusión de las Frecuencias Respiratorias, es un proceso basado en Nemati et al. [38], donde propone un método de fusión de frecuencias respiratorias, basado en el índice de pureza de la señal, la aplicación de un filtro de estimación de estados denominado Filtro Kalman y un método de fusión de estimaciones independientes basado en la incertidumbre de la señal.

**Índice de pureza de la Señal(SQI)** Las señales respiratorias pueden ser caracterizadas en base a un índice de pureza de la señal; la fidelidad de la extracción de la frecuencia respiratoria de las señales derivadas de ECG, depende enteramente de la periodicidad de la señal EDR; en una respiración regular, la señal muestra un patrón sinusoidal con su frecuencia dominante (el pico más alto al analizar el espectro de frecuencias de la señal) en la frecuencia respiratoria del paciente [38].

El procedimiento para determinar el grado de oscilación de una señal, fue desarrollado para el ámbito de la Electroencefalografía (EEG) donde las ondas obtenidas son analizadas en base a la oscilación que presentan, el método para poder obtener el SQI de una señal respiratoria puede hacerse de dos maneras, con el análisis espectral directo de la señal o con la utilización de ciertos parámetros desarrollados por Hjorth, donde realiza la ubicación de tres características: Actividad, Movilidad y Complejidad, que caracterizan las señales en cuanto a su amplitud, escala de tiempo y la referencia a una curvatura sinusoidal, también conocidos como los descriptores de Hjorth [39, 40].

Los descriptores de Hjorth son obtenidos de acuerdo a los momentos espectrales 0, 2 y 4, que pueden ser analizados en base a una descomposición espectral utilizando la transformada de Fourier y la potencia de la señal o por medios matemáticos de acuerdo a las ecuaciones 4.4, 4.5 y 4.6, con un costo computacional menor; el índice de calidad de la señal senoidal es calculado de acuerdo a una relación entre los momentos espectrales de acuerdo a la ecuación 4.7 [38], la implementación se muestra en el código 4.6 con la función *sqiCalculator*.

$$m0 = 1/\tau \sum_{n=1}^{\tau} x^2(n) \quad (4.4)$$

$$m2 = 1/\tau \sum_{n=1}^{\tau} (x(n)')^2 \quad (4.5)$$

$$m4 = 1/\tau \sum_{n=1}^{\tau} (x(n)'' )^2 \quad (4.6)$$

$$\Gamma_s(k) = \frac{\bar{\omega}_2^2(k)}{\bar{\omega}_0(k)\bar{\omega}_4(k)} \quad (4.7)$$

```

1 def sqiCalculator (self, signal):
2     n = len(signal)
3     sg = np.array(signal)
4     '''
5     Momento espectral 0 = m0
6     Momento espectral 2 = m2
7     Momento espectral 4 = m4
8     '''
9     sgsq = sg**2
10    m0 = np.mean(sgsq)

```

```

11     dsg=[]
12     tempdsg=0
13     for i in range(1,len(signal)):
14         tempdsg = (signal[i]-signal[i-1])
15         dsg.append(tempdsg)
16     dsg = np.array(dsg)
17     gp =generalProcessing()
18     dsg = gp.Normalize(dsg)
19     dsgsq = dsg **2
20     m2 = np.mean(dsgsq)
21     d2sg=[]
22     tempd2sg= 0
23     for k in range(1,len(dsg)):
24         tempd2sg = (dsg[k]-dsg[k-1])
25         d2sg.append(tempd2sg)
26     d2sg = np.array(d2sg)
27     d2sg = gp.Normalize(d2sg)
28     d2sgsq= d2sg**2
29     m4= np.mean(d2sgsq)
30
31     num = m2**2
32     den = m0*m4
33
34     spi = num/den

```

Código 4.6: Cálculo del índice de Pureza de la señal

**Filtro Kalman** Tal y como se explicó en la introducción de esta sección, el proceso de fusión de las medidas respiratorias conlleva la utilización de un método de optimización en espacio de estados para señales estocásticas denominado Filtro Kalman, el cual, es un conjunto de ecuaciones matemáticas que implementan un estimador de predicción y corrección, que es óptimo en el sentido que minimiza la covarianza del error presentado a partir de ciertas condiciones conocidas, estima el estado de un proceso controlado  $x$  con una medida  $z$ , que están gobernados por las ecuaciones:

$$\begin{aligned}
 x_k &= Ax_{k-1} + Bw_{k-1} \\
 z_k &= Hx_k + v_k
 \end{aligned}$$

Donde  $w$  y  $v$  son variables con una distribución de probabilidad normal, representando al ruido del proceso y de la medición respectivamente, que presentan covarianzas de ruido  $Q$  y  $R$  mismas que pueden cambiar en cada paso o medición; en la implementación basada en [38] se toman como constantes, al igual que las



variables A y B; A que relaciona el estado anterior( $k - 1$ ) con el estado presente ( $k$ ) de la medición y B que relación la función de control con el estado  $x$ .

$$p(w) \sim N(0, Q)$$

$$p(v) \sim N(0, R)$$

Dichas ecuaciones dan origen a dos conjuntos de ecuaciones más tomando en cuenta que se denominará un estado *a priori* ( $\hat{x}_k^-$ ) y un estado *a posteriori* ( $\hat{x}_k$ ); un estado *a posteriori* es obtenido de acuerdo a un cálculo lineal de una estimación *a priori* y la innovación o residuo kalman, que es la diferencia entre la medida original y la predicción encontrada por el filtro, de acuerdo a la siguiente ecuación:

$$\hat{x}_k = \hat{x}_k^- + K(z_k - H\hat{x}_k^-) \quad (4.8)$$

la matriz  $K$  es la ganancia que minimiza error de covarianza *a posteriori*, que a su vez es dependiente de la ganancia de la covarianza del error estimado *a priori* ( $P_k^-$ ) de acuerdo a la ecuación 4.9, de modo que, cuando la covarianza de error de la medición  $R$  se aproxima a cero, la medición  $z_k$  será más confiable que la medida predecida, caso contrario cuando  $P_k^-$  se aproxima a cero, que se define de acuerdo a la ecuación 4.10.

$$K_k = \frac{P_k^-}{P_k^- + R} \quad (4.9)$$

$$P_k^- = P_{k-1} + Q \quad (4.10)$$

Dichas ecuaciones se clasifican en dos, ecuaciones de predicción y ecuaciones de corrección [41], en la figura 4.22 se observa la interacción de las mismas que fue implementado en el código 4.7, basado en los cambios propuestos por [38], donde  $R = R_k e^{SQI_k^2 - 1}$  y  $SQI_k = \Gamma(k)$ , para permitir al filtro kalman realizar un estimado variante de la covarianza del ruido usando estimados de la calidad de las señales basado en ECG y PPG, con valores iniciales de  $R=1$  y  $Q=5$ , la función donde fue

implementado dicho filtro retorna las innovaciones para cada una de las respiraciones que serán utilizadas en los pasos finales de la fusión [38].

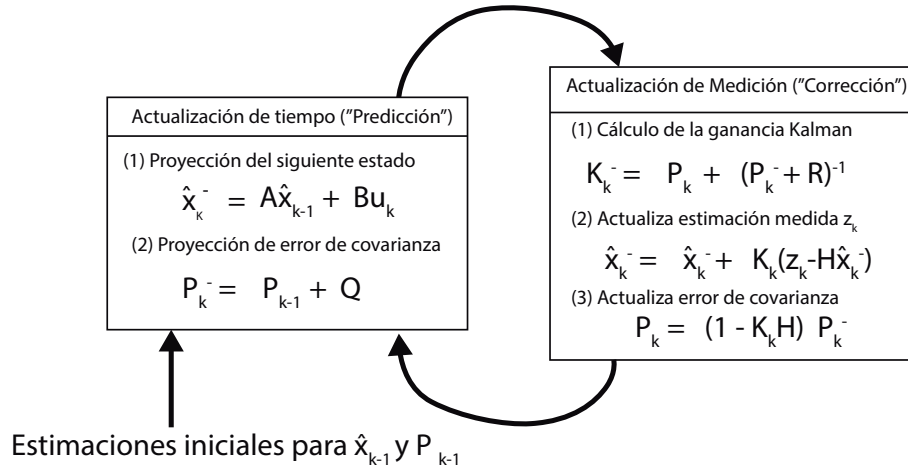


Figura 4.22: Diagrama de funcionamiento general de un filtro Kalman [41]

```

1 def kalmanFilter(self, signal, sqiIndex, RR = True):
2     n_iter = len(signal)
3     sz = n_iter # size of array
4     z=signal
5     Q = 5 # process variance
6     purityIndex = sqiIndex
7     #print "PI:", purityIndex
8
9     # allocate space for arrays
10    xhat=np.zeros(sz)      # a posteri estimate of x
11    P=np.zeros(sz)        # a posteri error estimate
12    xhatminus=np.zeros(sz) # a priori estimate of x
13    Pminus=np.zeros(sz)   # a priori error estimate
14    K=np.zeros(sz)        # gain or blending factor
15    R= 1
16
17    # intial guesses
18    xhat[0] = 0.0
19    P[0] = 10*Q
20    for k in range(1,n_iter):
21        if RR == False:
22            R = R*np.exp((purityIndex**-2)-1)
23        else:
24            if purityIndex == 0:
25                0.01
26            R = R*np.exp((purityIndex[k]**-2)-1)
27        # time update
28        xhatminus[k] = xhat[k-1]
29        Pminus[k] = P[k-1]+Q

```

```

30         # measurement update
31         K[k] = Pminus[k]/( Pminus[k]+R )
32         xhat[k] = xhatminus[k]+K[k]*(z[k]-xhatminus[k])
33         P[k] = (1-K[k])*Pminus[k]
34     kalmanResidue = z-xhat
35     gp = generalProcessing()
36     kalmanResidue = gp.Normalize(kalmanResidue)
37     return kalmanResidue, xhat
38

```

Código 4.7: Filtrado de tipo Kalman basado en [41].

**Fusión de Estimaciones** Para el cálculo de la fusión de las frecuencias respiratorias, las estimaciones producidas por el filtro Kalman deben ser fusionadas tomando en cuenta la incertidumbre asociada con cada estimación, para dicha tarea se siguió el procedimiento descrito en [38], así, para cada estimación final en algún instante de tiempo, será definido por:

$$X_k = \sum_{s=1}^s \left( \frac{\prod_{i=1, i \neq k}^s \sigma_{k,i}^2}{\sum_{s=1}^s \left( \prod_{j=1, j \neq i}^s \sigma_{k,i,j}^2 \right)} \cdot x_{k,s} \right) \quad (4.11)$$

donde  $X_{s,k}$  y  $\sigma_{k,s}$  son estimaciones independientes y están asociados a la incertidumbre del sensor  $s$  en el instante de tiempo  $k$ .  $\sigma_{k,s}$  es la innovación resultante del filtro Kalman, para lograr la fusión con respecto a la señal de ECG, los residuos deben relacionarse con el índice de calidad de la señal, de modo que:  $\sigma_{k,s}^2 = (r_{k,s}/SQI_{k,s})^2$ , se aplicaron dos tipos de fusión de acuerdo a la propuesta de [38], Fusión 0 (*rrF0*) y Fusión 1 (*rrF1*).

Fusión 0 toma en cuenta el índice de calidad de la señal como unitario, por lo que solamente dependerá la fusión de los resultados de las innovaciones del filtro Kalman, por otra parte, Fusión 1, si toma en cuenta el índice de calidad y permite una fusión dependiente de las señales producidas mediante los algoritmos propuestos.

Como validación y preparación para la formación del conjunto de entrenamiento para el algoritmo inteligente, se realizó el cálculo de la frecuencia respiratoria para 300 segundos de prueba, tomando en cuenta cada 5 segundos el cálculo de la frecuencia respiratoria, los resultados obtenidos se muestran en la figura 4.23, a su vez se realizó el mismo procedimiento para distintos conjuntos de 300 segundos de

señal obteniendo el porcentaje de error en cada una de las variables a computar, presentado en la tabla 4.1.

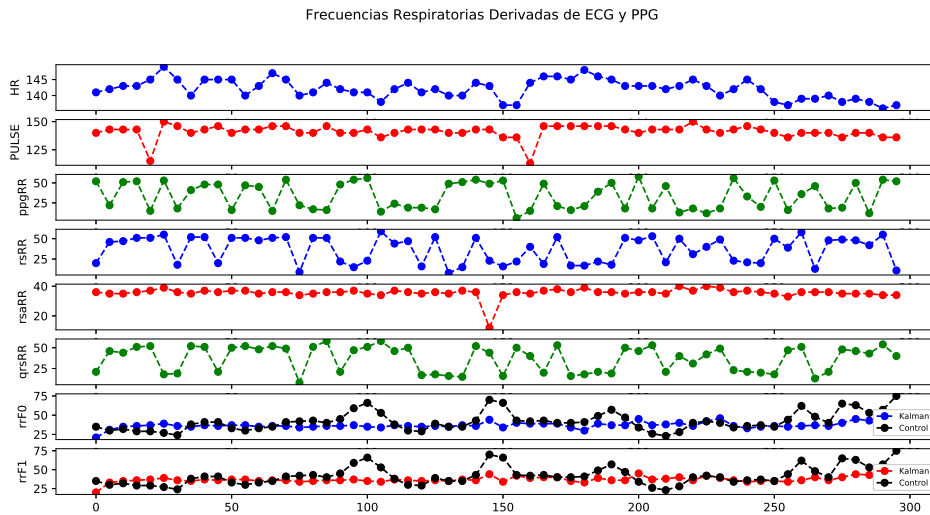


Figura 4.23: Frecuencias Respiratorias derivadas de ECG en un tramos de 300 segundos con subdivisión de 5 segundos [Fuente Propia].

n	ERROR HR	ERROR PR	ERRORPPG	ERRORRS	ERRORRSA	ERRORQRS	ERROR F0	ERROR F1
1	1 %	5 %	44 %	44 %	18 %	42 %	22 %	22 %
2	1 %	2 %	44 %	37 %	27 %	37 %	37 %	37 %
3	1 %	5 %	46 %	46 %	25 %	46 %	25 %	24 %
4	1 %	6 %	50 %	40 %	24 %	40 %	24 %	24 %
5	1 %	4 %	44 %	41 %	18 %	44 %	21 %	22 %
6	1 %	5 %	47 %	42 %	21 %	44 %	25 %	25 %
7	1 %	6 %	36 %	31 %	24 %	31 %	20 %	20 %
8	1 %	7 %	32 %	36 %	31 %	38 %	24 %	24 %
9	1 %	6 %	38 %	26 %	29 %	31 %	21 %	21 %
10	1 %	4 %	34 %	32 %	29 %	24 %	19 %	20 %
11	1 %	3 %	31 %	34 %	32 %	37 %	32 %	32 %
12	2 %	14 %	40 %	39 %	32 %	38 %	32 %	32 %
13	1 %	9 %	41 %	35 %	31 %	33 %	27 %	27 %
14	1 %	4 %	40 %	41 %	37 %	42 %	47 %	45 %
15	1 %	13 %	41 %	38 %	28 %	38 %	35 %	31 %
16	1 %	14 %	43 %	40 %	29 %	40 %	29 %	28 %
17	1 %	6 %	41 %	42 %	23 %	40 %	24 %	24 %
18	1 %	4 %	40 %	36 %	23 %	36 %	24 %	23 %
19	1 %	5 %	36 %	37 %	34 %	39 %	30 %	30 %
20	2 %	12 %	36 %	43 %	33 %	40 %	32 %	33 %
T	1 %	7 %	40 %	38 %	27 %	38 %	28 %	27 %

Tabla 4.1: Tabla de resultados porcentaje de error 20 lapsos de tiempo de 300 segundos [Fuente Propia].

## 4.6. Conjunto de Entrenamiento

En la presente sección se describe de manera detallada el proceso realizado para la creación del conjunto de entrenamiento que fue utilizado para el modelado del algoritmo inteligente para el desarrollo del sistema, abarcando la adquisición de los datos de los pacientes, la determinación de apneas en los datos, obtención de características y la selección de los conjuntos de características. Los datos utilizados para la creación del conjunto de entrenamiento fueron obtenidos a partir de una base de datos utilizada a nivel internacional llamada MIMIC III.

#### 4.6.1. MIMIC III Database Physionet

Como se mencionó anteriormente, para la creación del conjunto de entrenamiento se utilizaron los datos de un banco de señales fisiológicas llamado Physionet, donde están alojadas largas colecciones de datos, que pueden ser accedidas de manera gratuita y que es apoyado por el National Institute of General Medical Sciences (NIGMS) y el National Institute of Biomedical Imaging and Bioengineering (NIBIB) bajo la subvención NIH 2R01GM104987-09 [42].

Dentro de las colecciones se encuentra la colección MIMIC III, la cual contiene miles de grabaciones de múltiples señales fisiológicas y series temporales de señales vitales o numerales obtenidas a través de la utilización de monitores de signos vitales en pacientes adultos y neonatales en la Unidad de Cuidados Intensivos (UCI) [42].

Las señales registradas pueden variar dependiendo de las decisiones realizadas por el personal de cuidados intensivos. Dentro de los registros utilizados se tenían diversas curvas tanto de ECG, Presiones Arteriales, Fotopletismografía de Pulso y Respiración mismas que fueron descartadas de acuerdo a los datos que presentaban y los huecos y/o falta de información o de señales en los mismos.

También, se incluyen mediciones numéricas de Ritmos Cardíaco y Respiratorio, Saturación Parcial de Oxígeno y Presiones Arteriales, las cuales del mismo modo que las señales crudas fueron preprocesadas y evitar datos nulos o huecos en la medición.

La duración obtenida de los datos fue muy variada teniendo registros de minutos, horas o hasta días. Para un mejor manejo de los datos se estableció un control de tiempo tomando en cuenta los segundos que habían transcurrido en la adquisición de los datos, misma que fue realizada a partir de dos maneras, con la utilización de una librería multi-entorno llamada *WFDB* o descargados a partir de la interfaz grafica del *Physiobank ATM* la cual se pesenta en la figura 4.24

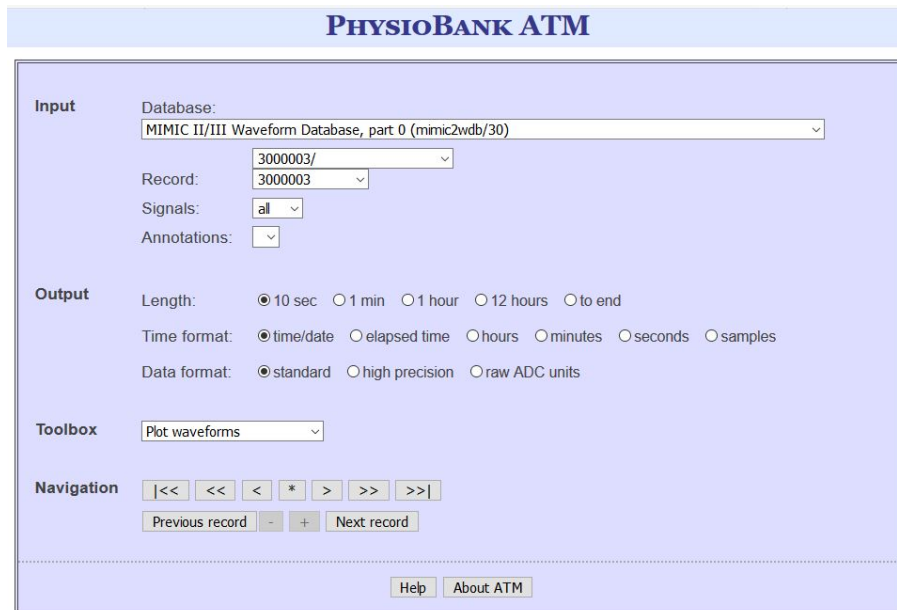


Figura 4.24: Interfaz Gráfica Physiobank ATM para obtención de datos MIMIC III.

Debido a diversas limitaciones encontradas al momento de descargar los datos a partir del ATM, se optó por la utilización de la librería *WFDB* en un entorno virtual de Linux por su facilidad de uso, ya que, al utilizar la función *rdsamp* fue posible la selección de los canales de datos que eran convenientes para el desarrollo del proyecto, así como el tiempo transcurrido para la descarga de datos de principio a fin, los datos se obtuvieron en un formato de valores separados por comas (CSV) mismos que fueron importados por medio de un algoritmo de pre-procesamiento de los datos crudos y alojados en tablas dentro de una base de datos denominada *ApnoeaCE* en SQL Server.

#### 4.6.2. Detección de casos de Apnea

Los registros utilizados para la creación del conjunto de entrenamiento, se diferenciaban entre los que contenían los numerales de las mediciones obtenidas a partir del monitor de signos vitales con una “n” al finalizar el identificador, p.e. “3000544n”, Éstos valores numéricos de medición procesada fueron clave para la detección de posibles casos de apnea.

La clasificación y obtención de características para el conjunto de entrenamiento

se realizó siguiendo el ciclo descrito en la figura 4.25, en donde se obtienen los datos sin procesar a partir de *ApnoeaCE* se realiza el procesamiento de datos para validación y de control y se obtienen las características a utilizar para el conjunto de entrenamiento que son ingresadas en una tabla dentro de *ApnoeaCE* denominada *ApnoeaFeatures*. En los próximos puntos se expondrá más a detalle el funcionamiento del proceso de detección de apneas y el cálculo de las características utilizadas.

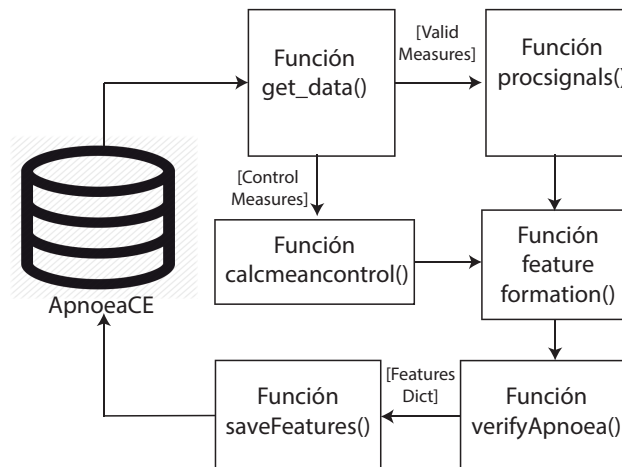


Figura 4.25: Proceso de detección Apneas y cálculo de características [Fuente Propia].

### Descripción y Obtención de Características

Se analizaron los valores en tiempo cuando la  $SpO_2$  era menor a 90 %, posteriormente se buscaron los casos dentro de esos tiempos en donde la diferencia entre cada uno era mayor a 300 segundos, se obtuvieron un total de 982 casos, en registros de 19 pacientes neonatales. Se realizó una función para la obtención de la información alojada en SQLServer con datos correspondientes a las señales crudas y los numerales de acuerdo a cada uno de los casos identificados. Una vez obtenidas las señales en forma de vector de datos se dio paso al procesamiento para la obtención de las siguientes características vitales de los pacientes:

- Ritmo Cardíaco (*HR*).
- Pulso Cardíaco (*PR*).
- Porcentaje de Saturación de Oxígeno (*SpO2*).



- Ritmos Respiratorios Derivados de ECG y PPG.
  - Índices de Calidad de las señales (SQI)

Dichas características corresponden al principio de experimentación de este proyecto, debido a que al momento de presentarse una apnea, el diafragma dejará de moverse por unos instantes por lo que la señal cardíaca no presentará los cambios habituales presentes en la modulación del pico del ciclo QRS, mismo que se puede observar en las señales respiratorias derivadas o en la señal de Impedancia Torácica incluida en los datos obtenidos de Physionet.

Posterior a la disminución de la actividad respiratoria se observa la presencia de Bradicardia o disminución de los Frecuencia Cardíaca y la actividad Electrocardiográfica, que precede a la disminución total de la SpO<sub>2</sub>, siendo esta la característica principal por la que la apnea es detectada a nivel empírico y físico con el cambio en coloración del neonato.

En la figura 4.26 se presenta un caso de apnea en donde se observan los cambios secuenciales descritos anteriormente en los diversos sistemas fisiológicos que intervienen en el proceso respiratorio en una ventana de 300 segundos previos a la caída de la SpO<sub>2</sub> en el paciente por debajo de un 90 % de concentración sanguínea, por lo que se considera un posible caso de apnea.

Para la determinación de apneas en el Conjunto de Entrenamiento, se tomó como base el fenómeno de disminución cardíaca debido a la apnea, considerando un caso verdadero de apnea a aquel caso que, en un lapso de 300 segundos previos a la caída de SpO<sub>2</sub> menor a 90 %, presente una disminución de HR mayor a 10 Latidos Por Minuto (LPM), que tenga como mínimo en HR un valor que sea menor a 130 LPM y otro factor a considerar en la clasificación de apnea para el conjunto de entrenamiento, es la presencia de ruido o el grado de calidad o fidelidad que tiene la señal, la cuál será indicativo del estado estático o dinámico del paciente, el cuál es reflejado directamente como un ruido externo a la señal provocando movimientos en la línea basal de la misma, en caso de no cumplirse las tres condiciones anteriores, el caso es considerado una falsa alarma.

Una vez obtenidos los datos se procede a obtener las características a utilizar en el Conjunto de Entrenamiento para la RNA. A partir de las mediciones obtenidas

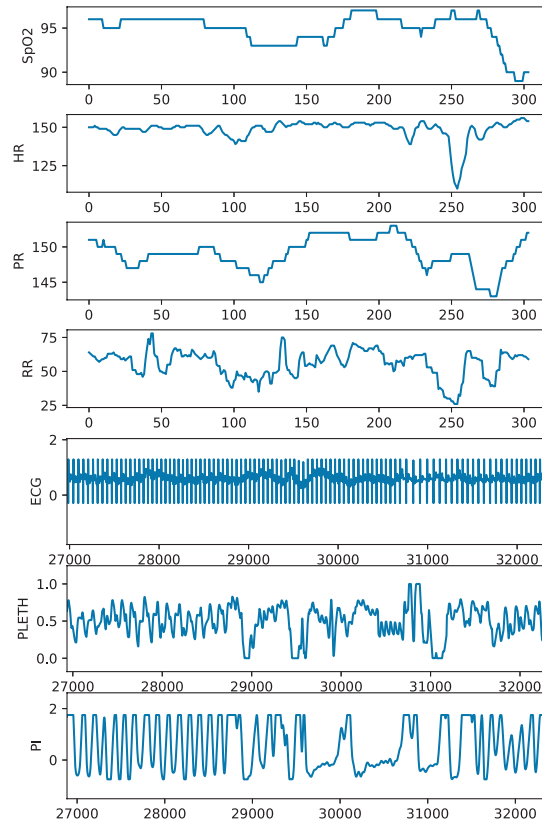


Figura 4.26: Descripción de Apnea en forma gráfica con fenómenos fisiológicos notables [Fuente Propia].

para las señales de los pacientes, se calculan las características de acuerdo a la tabla 4.2, considerando los valores mínimos y pendientes en ventanas de tiempo dentro de los 300 segundos de análisis, lo que generaría un panorama de las condiciones límite antes de la detección de la apnea, éste proceso se detallará en las secciones siguientes.

Variable	Características a utilizar
ECG	minHR, slopeHR
PPG	minPR, slopePR, sqi_ppg, slopeO2
EDR	minEDR/PEDR, sqi_EDR, minF0/1

Tabla 4.2: Tabla de Características por variable para Conjunto de Entrenamiento [Fuente Propia].

### Obtención de datos

Para la realización del proceso descrito de clasificación de los casos de apnea en el Conjunto de Entrenamiento, se creó la función *get\_data*, dicha función ejecuta una serie de Procedimientos Almacenados en SQL Server para la adquisición de los datos, se tienen dos diccionarios con un query de ejecución de un procedimiento por tabla para los datos de señal original y datos numéricos, una vez adquiridos se generan marcos de datos y se regresan vectores con cada una de las series de datos para señales sin procesar y valores numerales de control, misma que se describe en el código 4.8 Para la creación del Conjunto de Entrenamiento se utilizaron los siguientes pacientes registrados en la base de datos MIMIC III de physionet:

- 3000544
- 3000775
- 3001049
- 3001703
- 3001788
- 3002559
- 3003030
- 3005344
- 3005424
- 3006179
- 3006240
- 3006599
- 3008278
- 3003482
- 3005290
- 3009183
- 3009639
- 3009655
- 3010009

```

1 def get_data(self, rn_name, time_init, engine):
2     path = "C:\Users\danos\source\repos\CETesis"
3     data_queries = {
4         3000544:"EXEC get_Apnoea_time_0544 {}".format(time_init),

```

```

5     3000775:"EXEC get_Apnoea_time_0775 {}".format(time_init),
6     ...
7     3009655:"EXEC get_Apnoea_time_9655 {}".format(time_init),
8     3010009:"EXEC get_Apnoea_time_9655 {}".format(time_init)
9 }
10
11 n_data_queries = {
12     3000544: "EXEC get_Apnoea_ntime_0544 {}".format(time_init),
13     3000775: "EXEC get_Apnoea_ntime_0775 {}".format(time_init),
14     ...
15     3009655: "EXEC get_Apnoea_ntime_9655 {}".format(time_init),
16     3010009: "EXEC get_Apnoea_ntime_9655 {}".format(time_init)
17 }
18
19 vs_monitor_measure_df = pd.read_sql_query(n_data_queries[rn_name],engine)
20 all_measure_df = pd.read_sql_query( data_queries[rn_name],engine)
21
22 # organizes data
23 baby_hr_control = vs_monitor_measure_df['HR'].values
24 baby_pr_control = vs_monitor_measure_df['PULSE'].values
25 baby_rr_control = vs_monitor_measure_df['RESP'].values
26 baby_sp_o2_control = vs_monitor_measure_df['SpO2'].values
27 baby_ecg_valid = all_measure_df['ECG'].values
28 baby_pleth_valid = all_measure_df['PLETH'].values
29 baby_resp_valid = all_measure_df['RESP'].values
30
31 return baby_ecg_valid, baby_pleth_valid, baby_hr_control, baby_pr_control,
32     baby_rr_control, baby_sp_o2_control, baby_resp_valid

```

Código 4.8: Función `get_data()`

## Cálculo de características

El cálculo de las características mencionadas en la tabla 4.2 es realizado a través de tres funciones principales: la función `procsignals()`, la función `calcmeancontrol()` y la función `featureformation()`.

Una vez obtenidas las señales a partir de la función `get_data` se procede al procesamiento de las mismas en la función `procsignals` de acuerdo al diagrama presentado en la figura 4.25.

**Función `procsignals()`** La función `procsignals()` está compuesta por distintos pasos a realizar con la finalidad de arrojar los numerales de medición necesarios para la formación de las características a utilizar en la RNA.

En la figura 4.27 se presenta el algoritmo comenzando con la división de los 300 segundos de señal obtenidos en *get\_data* en 20 partes de 15 segundos que a su vez volverán a ser divididos para dejar como unidades de medición para procesamiento lapsos de 5 segundos de las distintas señales a utilizar, cabe mencionar que para la obtención de las características para la RNA solamente se utilizan las señales de ECG y Fotopletismografía de Pulso.

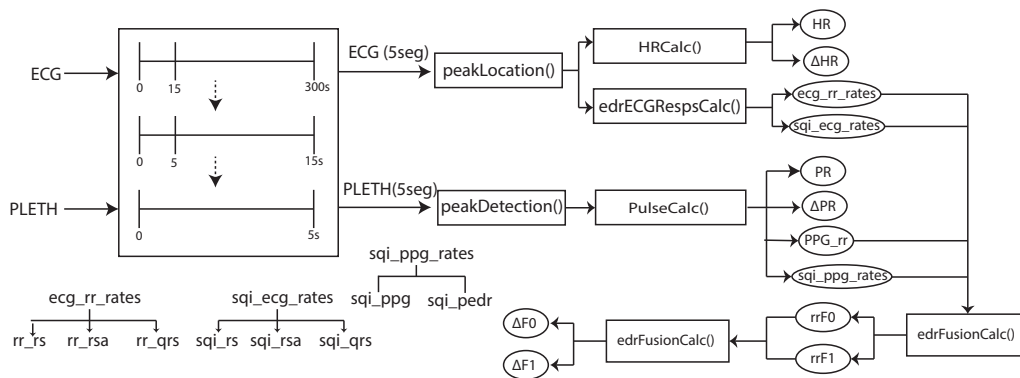


Figura 4.27: Diagrama Función *procsignals()* [Fuente Propia].

Una vez que son divididas cada una de las señales tiene su propio procesamiento de acuerdo a lo presentado en la figura 4.9 arrojando como resultados un valor para HR y otro para la pendiente de HR ( $\Delta HR$ ) en el lapso de 5 segundos. Para el cálculo de los Ritmos Respiratorios e Índices de calidad de las señales se realizó la función *edrECGRespCalc()*, la cual considerando los procesos de obtención de Frecuencia Respiratoria por medio de señales EDR se calculó el conjunto *ecg\_rr\_rates* que está conformado por [*rr\_rs*, *rr\_rsa*, *rr\_qrs*], del mismo modo se calculan los valores para los SQI de las señales respiratorias, al aproximarlas a una señal senoidal mismos que son almacenados en la variable *sqi\_ecg\_rates*.

De manera similar, para la señal de Fotopletismografía se realizó la adquisición de los valores de Pulso (PR),  $\Delta PR$ , y las variables de *sqi\_ppg*, *sqi\_pedr* y la frecuencia respiratoria *ppg\_rr*.

**Función *calcmeancontrol()*** Del mismo modo a la división de datos que se realizó en la función *procsignals()*, se realiza la división de los numerales obtenidos en la consulta, y se calcula el promedio para *HR*, *PR*, *RR*, *SpO2* y a su vez se calcula la

pendiente para cada uno de los numerales adquiridos en los 300 segundos de lectura previos a la caída de la SpO2, de acuerdo al código 4.9.

```

1  def calcmeancontrol(self, baby_hr_control, baby_pr_control, baby_rr_control,
2      baby_spo2_control):
3
4      step1 = 20
5      step2 = 5
6
7      for i in range(0, len(baby_hr_control), step1):
8          window_ecg_20sec_control = baby_hr_control[i:(i + step1)]
9          window_ppg_20sec_control = baby_pr_control[i:(i + step1)]
10         window_rr_20sec_control = baby_rr_control[i:(i + step1)]
11         window_o2_20sec_control = baby_spo2_control[i:(i + step1)]
12
13         for x in range(0, len(window_ecg_20sec_control), step2):
14             low = x
15             high = x + step2
16
17             window_ecg_3sec_control = window_ecg_20sec_control[low:high]
18             self.mean_hr_control.append(int(np.round(np.mean(window_ecg_3sec_control), 0)))
19
20             window_ppg_3sec_control = window_ppg_20sec_control[low:high]
21             self.mean_pr_control.append(int(np.round(np.mean(window_ppg_3sec_control), 0)))
22
23             window_rr_3sec_control = window_rr_20sec_control[low:high]
24             self.mean_rr_control.append(int(np.round(np.mean(window_rr_3sec_control), 0)))
25
26             window_o2_3sec_control = window_o2_20sec_control[low:high]
27             self.mean_o2_control.append(int(np.round(np.mean(window_o2_3sec_control), 0)))
28
29             xreg_hr_control = np.arange(0, len(window_ecg_3sec_control))
30             xreg_hrc = np.vstack([xreg_hr_control, np.ones(len(xreg_hr_control))]).T
31             m_hr = np.linalg.lstsq(xreg_hrc, window_ecg_3sec_control, rcond=-1)[0]
32             self.slope_hr_control_windows.append(np.round(m_hr[0], 4))
33
34             xreg_pr_control = np.arange(0, len(window_ppg_3sec_control))
35             xreg_prc = np.vstack([xreg_pr_control, np.ones(len(xreg_pr_control))]).T
36             m_pr = np.linalg.lstsq(xreg_prc, window_ppg_3sec_control, rcond=-1)[0]
37             self.slope_pr_control_windows.append(np.round(m_pr[0], 4))
38
39             xreg_rr_control = np.arange(0, len(window_rr_3sec_control))
40             xreg_rrc = np.vstack([xreg_rr_control, np.ones(len(xreg_rr_control))]).T
41             m_rr = np.linalg.lstsq(xreg_rrc, window_rr_3sec_control, rcond=-1)[0]
42             self.slope_rr_control_windows.append(np.round(m_rr[0], 4))
43
44             xreg_o2_control = np.arange(0, len(window_o2_3sec_control))
45             xreg_o2_c = np.vstack([xreg_o2_control, np.ones(len(xreg_o2_control))]).T
46             m_o2 = np.linalg.lstsq(xreg_o2_c, window_o2_3sec_control, rcond=-1)[0]
47             self.slope_o2_windows.append(np.round(m_o2[0], 4))

```

Código 4.9: Función calcmeancontrol()

**Función featureformation()** La función *featureformation()* es la encargada de la formación de un diccionario, que será almacenado en una tabla de características y resultados para ser utilizada en el entrenamiento de la RNA.

En la figura 4.28 se muestra la formación del Diccionario de características para

el conjunto de entrenamiento y la verificación de las apneas realizado de acuerdo a lo descrito por [30], tomando lapsos de 300-15k segundos teniendo como inicio el instante de tiempo donde comienza la desaturación menor a 90 % en el paciente, buscando el valor el valor mínimo en el lapso y analizano el área bajo la curva en una prueba unitaria con las medidas de control y las registradas.

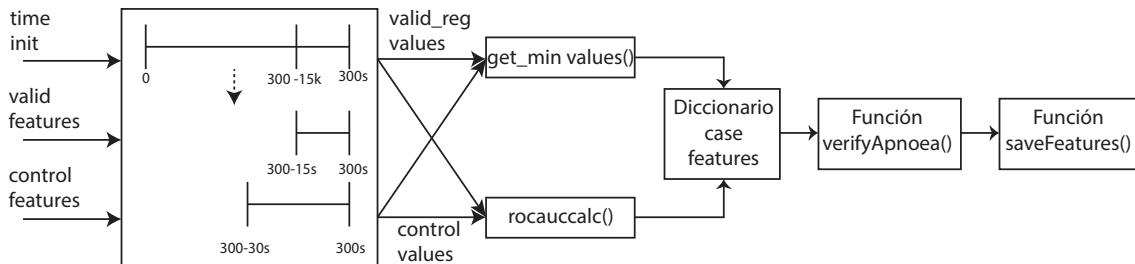


Figura 4.28: Diagrama Función featureformation() [Fuente Propia].

Para la verificación de la apnea se consideran las condiciones descritas al inicio de esta sección a través de la función verifyApnoea() (véase cod. 4.10) en donde al cumplirse ambas condicionales el algoritmo grafica las señales adquiridas tanto de manera cruda como numerales y permite al usuario el análisis visual de las señales, junto con el diccionario de características obtenido en el caso de apnea analizado y que solicita una clasificación al usuario de acuerdo a sus observaciones.

Esta clasificación puede ser '1' en caso de no ser una falsa alarma, '2' en caso de una falsa alarma o '3' en caso de indecisión, cabe mencionar que solamente se utilizarán los valores nominales de 0 y 1 para la clasificación de una falsa alarma o un caso de apnea, finalmente se guarda el diccionario en un marco de datos para ser ingresado en la tabla de la base de datos utilizada.

```

1  def verifyApnoea(self,init_time, hr_control):
2      # hr_control = self.calculated_hr
3      low_hr_flag = False
4      diff_hr = False
5      low_hr_detected = []
6      Apnoea = 0
7      for i in range(len(hr_control)):
8          if hr_control[i] <= 130:
9              low_hr_detected.append(i)
10             low_hr_flag = True
11  if np.max(hr_control)-np.min(hr_control) >10:
12      diff_hr = True

```

```

13     print("HR less than 130: {}".format(low_hr_flag), "\n10 bpm hr drop: {}".format(
diff_hr))
14     plt.show()
15     if low_hr_flag and diff_hr:
16         print("Possible Apnoea check signals")
17         print("verify HR at: {}".format(low_hr_detected))
18         plt.show()
19         det_apnea = input("Apnoea case?" + "\n1: Positive case, 2: negative, 3:
unsure\n")
20         if det_apnea == '1':
21             Apnoea = 1
22         elif det_apnea == '2':
23             Apnoea = 0
24         elif det_apnea == '3':
25             Apnoea = 3
26     else:
27         plt.show()
28         print("No Possible Apnoea detected")
29         plt.close()
30     return Apnoea
31

```

Código 4.10: Función verifyApnoea()

**Función rocauccalc()** Una vez que la información del lapso es extraída se registra el valor mínimo de las variables analizadas y se realiza un Análisis Unitario de Área Bajo la Curva, con la finalidad de obtener la medida con la mayor Área Bajo la Curva que garantice el mejor intervalo evaluado de la variable analizada, tomando como base los registros de control de los numerales obtenidos y comparandolos con los registros calculados a partir del procesamiento de las señales.

El área de la curva ROC calculada para cada una de las comparaciones es almacenada en un vector para posteriormente seleccionar el lapso de mayor área, después se selecciona como valor de la variable dentro del diccionario al valor mínimo correspondiente en dicho lapso, el código para la realización de este procedimiento se presenta en el código 4.11.

```

1     def rocauccalc(control_vales, registered_values, limit, sqi=False):
2         global roc_auc_p
3         control_values = control_vales
4         reg_values = registered_values
5         binreg = np.ones(len(control_values))
6         if sqi:
7             for i in range(len(control_vales)):
8                 if reg_values[i] < control_vales[i] - limit:

```



```

9         binreg[i] = 0
10     try:
11         roc_auc_p = mt.roc_auc_score(binreg, control_vales)
12         roc_auc_p = np.round(roc_auc_p, 2)
13     except:
14         fpr, tpr, threshold = mt.roc_curve(binreg, reg_values, pos_label=1)
15         if len(fpr) == 1 or len(tpr) == 1:
16             if np.isnan(fpr).any():
17                 roc_auc_p = 1
18             if np.isnan(tpr).any():
19                 roc_auc_p = 0
20         else:
21             if np.isnan(fpr).any():
22                 roc_auc_p = 1
23             elif np.isnan(tpr).any():
24                 roc_auc_p = 0
25             else:
26                 roc_auc_p = mt.auc(fpr, tpr)
27         roc_auc_p = np.round(roc_auc_p, 2)
28     else:
29         for i in range(len(control_values)):
30             if reg_values[i] < control_values[i] - limit or reg_values[i] >
control_values[i] + limit:
31                 binreg[i] = 0
32             fpr, tpr, threshold = mt.roc_curve(binreg, reg_values, pos_label=1)
33             if len(fpr) == 1 or len(tpr) == 1:
34                 if np.isnan(fpr).any():
35                     roc_auc_p = 1
36                 if np.isnan(tpr).any():
37                     roc_auc_p = 0
38             else:
39                 if np.isnan(fpr).any():
40                     fps = np.zeros(len(fpr))
41                     roc_auc_p = mt.auc(fps, tpr)
42                 elif np.isnan(tpr).any():
43                     tps = np.zeros(len(tpr))
44                     roc_auc_p = mt.auc(fpr, tps)
45                 else:
46                     roc_auc_p = mt.auc(fpr, tpr)
47             roc_auc_p = np.round(roc_auc_p, 2)
48     return roc_auc_p
49

```

Código 4.11: Función rocauccalc()

### 4.6.3. Selección de características

Una vez obtenidas las mejores características en el conjunto de entrenamiento se realizó el orden de las mismas de acuerdo a su relevancia y redundancia. Tomando como base el análisis realizado por [30] se utilizó la herramienta de Aprendizaje Automático WEKA [43] para la realización de esta distribución utilizando un método de filtrado para realizar una organización de todas las características en el conjunto de entrenamiento mostrado en la tabla 4.3.

S	Variable	S	Variable
S1	sqi_ppg	S10	slopeO2
S2	min_hr	S11	min_rs
S3	slopePR	S12	min_qrs
S4	slopeHR	S14	slopeF0
S5	min_f0	S15	sqi_rs
S5	min_f1	S16	slopeF1
S6	min_rsa	S17	sqi_pedr
S7	min_pr	S18	sqi_rsa
S8	min_o2	S19	min_ppg
S9	sqi_qrs		

Tabla 4.3: Tabla de organización de características de acuerdo a método de Filtrado [Fuente Propia].

## 4.7. Red Neuronal para clasificación de Apneas

Formado el Conjunto de Entrenamiento y realizado el ordenamiento de las características, se procedió al diseño de la RNA a utilizar para la clasificación final de las apneas. Para dicha tarea se implementó un modelo utilizando una biblioteca de software libre que facilita los cálculos numéricos mediante diagramas de flujo de datos llamada TensorFlow [44]. A su vez se utilizó una API de RNA de alto nivel, escrita en Python y que es capaz de utilizar TensorFlow como su base de procesamiento llamada Keras[45].

Es importante señalar que para la creación, entrenamiento y validación del modelo de RNA que se utilizó en este experimento contó con un hardware especializado que consistía en una computadora con un Sistema Operativo en Ubuntu, 16 GB en memoria RAM y una tarjeta de video GeForce GTX 1070 de Nvidia con una Unidad de Procesamiento de Gráficos de 8GB. El diseño, implementación y modelado de la RNA, se desarrolló en un entorno Jupyter que permite un mejor manejo del código implementado al hacerlo de manera modular.

Normalmente al diseñar un modelo de RNA se deben tomar en cuenta factores de diseño como el número de neuronas y capas a utilizar entre otros denominados hiperparámetros, el tipo de Red Neuronal ya sea Secuencial, Convolutiva o Recursiva. Dentro de los hiperparámetros a considerar se encuentra el número de épocas con las que se entrenará a RNA, el tamaño del lote de entrada de datos de alimentación de la RNA, el tipo de optimizador a utilizar, la activación de las neuronas que conformarán al modelo entre otros.

Normalmente una red neuronal tiene un proceso para su creación, comenzando con el diseño y la selección de valores para los hiperparámetros, posteriormente se realiza la compilación del modelo seleccionado, después se entrena con la finalidad de cargar las capas con los pesos correspondientes para la generalización del fenómeno estudiado en el conjunto de entrenamiento, dentro de esta fase se realiza una validación del entrenamiento para el ajuste de los hiperparámetros.

Seguido del entrenamiento se realiza una evaluación del nivel de aprendizaje de la RNA con valores desconocidos y se analiza la respuesta generada a partir del conocimiento obtenido, existen diferentes métricas que permiten conocer el desempeño de la RNA, mismas que serán descritas en las secciones siguientes.

Para este experimento se decidió la utilización de una RNA de tipo Secuencial, con aprendizaje Supervisado y un Conjunto de Entrenamiento que constaba de 981 casos de apnea con un total de 257 casos positivos y 724 casos negativos que fue almacenado en un archivo de valores separados por comas para su utilización de una manera rápida en el proceso de la RNA.

Las variaciones numerales del Conjunto de Entrenamiento tuvieron que ser preprocesadas para su utilización dentro del modelo de la RNA, por lo que se utilizan diversos métodos de procesamiento y normalización que se explicarán en las siguientes

secciones.

### 4.7.1. Preprocesamiento y creación de vectores de datos

Como se mencionó con anterioridad el Conjunto de Entrenamiento fue almacenado en un archivo .CSV, mismo que fue leído en un marco de datos denominado *train\_dataset* en formato *float32*, posteriormente se utilizan los escaladores *StandardScaler* y *MinMaxScaler* de la librería *sklearn*, el primer normalizador se encarga de garantizar un promedio igual a 0 y una varianza = 1, mientras que el segundo normalizador se encarga de tomar los valores límites y dejar los datos en un rango de 0 a 1, escalando las columnas distintas a 'apnoea'. Después, el Conjunto de Entrenamiento es separado en dos vectores, *targets* que corresponden a la columna 'apnoea' y el resto en una matriz *train\_dataset\_final*, mismos que se separarán en las distintas variables del Conjunto de entrenamiento para finalmente reorganizarlas en un vector *sort\_cv*, dicho algoritmo se describe en el código 4.12

```
1 train_dataset = pd.read_csv('dataset.csv',header = 0)
2 train_dataset = train_dataset.astype(dtype=np.float32)
3 std_scaler = StandardScaler()
4 min_max_scaler = MinMaxScaler()
5 for col in train_dataset.columns:
6     if col != 'apnoea':
7         train_dataset[col] = std_scaler.fit_transform(train_dataset[col].values.
8             reshape(-1,1))
9         train_dataset[col] = min_max_scaler.fit_transform(train_dataset[col].values.
10             reshape(-1,1))
11 targets= train_dataset['apnoea'].values
12 train_dataset_final = train_dataset.drop(['apnoea'],axis=1)
13 td = train_dataset_final.values
14 min_hr = td[:,0]
15 min_pr = td[:,1]
16 min_o2 = td[:,2]
17 min_rs = td[:,3]
18 min_rsa = td[:,4]
19 min_qrs = td[:,5]
20 min_ppg = td[:,6]
21 min_f0 = td[:,7]
22 min_f1 = td[:,8]
23 slopeHR = td[:,9]
24 slopePR = td[:,10]
25 slopeO2 = td[:,11]
26 slopeF0 = td[:,12]
27 slopeF1 = td[:,13]
28 sqi_rs = td[:,14]
```

```

27     sqi_rsa = td[:,15]
28     sqi_qrs = td[:,16]
29     sqi_pedr = td[:,17]
30     sqi_ppg = td[:,18]
31     sort_cv = [sqi_ppg, min_hr, slopePR, slopeHR, min_f0, min_f1, min_rsa, min_pr, min_o2,
32                sqi_qrs, slopeO2, min_rs, min_qrs, slopeF0, sqi_rs, slopeF1, sqi_pedr, sqi_rsa,
33                min_ppg]
34

```

Código 4.12: Código de Preprocesamiento de Conjunto de Entrenamiento

## 4.7.2. Modelo de RNA

Para la realización del modelo de la RNA se realizó una función que automatiza la creación del modelo (véase código 4.13), que recibe los hiperparámetros para las capas, nodos y dimensiones de entrada retornando el modelo creado para ser utilizado en el algoritmo de entrenamiento y validación de la RNA.

```

1     def create_model():
2         global inputs
3         model = Sequential()
4         for i, nodes in enumerate(layers):
5             if i == 0:
6                 model.add(Dense(nodes, input_dim= inputs, activation = 'relu'))
7             else:
8                 model.add(Dense(nodes, activation = 'relu'))
9                 if nodes > 100:
10                    model.add(Dropout(.5))
11         model.add(Dense(1, activation = 'sigmoid'))
12         model.compile(optimizer = Nadam(lr=0.0001), loss = 'binary_crossentropy', metrics= [
13             'accuracy'])
14         return model

```

Código 4.13: Función de creación de modelos RNA automatizado()

Una vez que se realizó el preprocesamiento de los datos del Conjunto de Entrenamiento se realizó el proceso de utilización de la RNA, que constaba de un entrenamiento, validación y evaluación de desempeño de la RNA. Recordando que en total se tienen 19 características en el Conjunto de Entrenamiento, se realizó una evaluación incremental en el número de características como entrada del modelo.

El modelo utilizado cuenta con una capa de entrada con un número de entradas 100 neuronas, otra capa Densa de 1000 neuronas, seguida de una capa de acondicionamiento tipo Dropout, otra capa Densa de 100 neuronas y una capa de salida

con una neurona. Para las capas de entrada y ocultas se utilizó un rectificador de tipo ‘relu’ mientras que para la capa de salida se utilizó un activación de tipo ‘sigmoid’ para garantizar una salida entre 0 y 1, que será el indicador de la presencia de una apnea o una falsa alarma, con un optimizador de tipo Nadam y una razón de aprendizaje de 0.0001, la estructura de la RNA se presenta en la tabla 4.4.

Capas(tipo)	Salida Forma	# Parámetros
Dense	(None,100)	200
Dense	(None,1000)	101000
Dropout	(None,1000)	0
Dense	(None, 100)	100100
Dense	(None, 1)	101

Tabla 4.4: Estructura de Modelo de RNA utilizado.

### 4.7.3. Métricas de Evaluación de RNA

Para la evaluación de la RNA se utilizaron diversas métricas, las cuales fueron calculadas utilizando la librería *sklearn*. Dentro de las métricas de evaluación se tomaron en cuentaa la Exactitud, Precisión, Sensibilidad, Especificidad y Valor F1. El diagnóstico de enfermedades y/o clasificación es un proceso que depende de la probabilidad de acertar a un resultado esperado. Para la obtención de las métricas se utilizaron las funciones *accuracy\_score* y *classification\_report*, la cual entrega un diccionario con los resultados obtenidos para cada una de las clases de clasificación, en este caso 1 o 0, la medida *recall* en la clase positiva, es considerada la sensibilidad y en la clase negativa la Especificidad de la clasificación[46].

El proceso de entrenamiento y evaluación de la RNA se llevó a cabo utilizando el código 4.14, en el que se especifica la utilización de la Unidad de Procesamiento Gráfico (GPU), se especifican los hiperparámetros de conjunto de aplicación o batch, época, las capas y sus nodos, se realiza el análisis incremental con el número de variables de entrada que se mencionó con anterioridad.

El conjunto de entrenamiento es dividido en dos partes, una con el 90 % de las muestras acomodadas de manera aleatoria y otra con el 10 % de las muestras para

la validación final del modelo, se utilizó la función *EarlyStopping* que permite al modelo detenerse en su entrenamiento cuando el valor de pérdida en validación sea aproximadamente igual al valor de pérdida del entrenamiento lo que indicará que se puede evitar el sobre entrenamiento al detener el proceso.

Se realiza el entrenamiento de la RNA utilizando la función *fit*, dicha función se encarga de entrenar y acondicionar los pesos de las neuronas a modo de realizar la generalización del fenómeno utilizando, aprendizaje Supervisado con el conjunto de entrenamiento y un grupo de validación del 10 % del conjunto utilizado en la función *fit*, el valor inicial de los pesos tiene un formato aleatorio sin embargo se fija un valor de raíz para tener siempre la repetibilidad del experimento.

Una vez terminado el entrenamiento, se procede a la evaluación del desempeño de la red utilizando la función *predict\_classes*, dicha función arrojará la clase de pertenencia de acuerdo a la probabilidad obtenida en la predicción utilizando un subconjunto del conjunto para el entrenamiento y el conjunto de validación separado al principio del proceso.

```

1   with tf.device('GPU:0'):
2       batch = 32
3       epochs = 1000
4       sort_train = np.array(sort_cv)
5       history_df = pd.DataFrame()
6       for input in range(1, len(sort_cv)+1):
7
8           train_features = sort_train[0:input].T
9           inputs = train_features.shape[1]
10          INPUT_NAME = 'Sort: {}, Input:s{}'.format('cv', input)
11          print('\n', INPUT_NAME)
12          layers = [100,1000,100]
13          model = create_model()
14          print("Model Fitting in Process....")
15          x_main, x_holdout, y_main, y_holdout = train_test_split(train_features, targets,
16          test_size = .1)
17          monitor = EarlyStopping(monitor= 'val_loss', min_delta=1e-3, patience=30, verbose
18          =1, mode = 'auto')
19          history = model.fit(x_main,y_main,batch_size = batch,callbacks=[monitor],
20          epochs = epochs, validation_split=.1,verbose=0)
21
22          print("Model Fitting Finished, evaluating model in progres.... ")
23          x_res, x_fit, y_res, y_fit = train_test_split(x_main,y_main,test_size=.2)
24
25          ts_pred = model.predict_classes(x_fit)
26          ev_pred = model.predict_classes(x_holdout)

```

```

26     print("Evaluation Completed, printing Results....")
27
28     fit_results = classification_report(y_fit, ts_pred, output_dict= True)
29     fit_prec = fit_results['1.0']['precision']
30     fit_sens = fit_results['1.0']['recall']
31     fit_spec = fit_results['1.0']['recall']
32     fit_f1 = fit_results['1.0']['f1-score']
33
34     eval_results = classification_report(y_holdout, ev_pred, output_dict= True)
35     eval_prec = fit_results['1.0']['precision']
36     eval_sens = fit_results['1.0']['recall']
37     eval_spec = fit_results['0.0']['recall']
38     eval_f1 = fit_results['1.0']['f1-score']
39
40     history_dict = {
41         'sort': 'cv',
42         'input': input,
43         'acc': np.round(np.mean(history.history['acc']),2),
44         'val_acc': np.round(np.mean(history.history['val_acc']),2),
45         'loss': np.round(np.mean(history.history['loss']),2),
46         'val_loss': np.round(np.mean(history.history['val_loss']),2),
47
48         'test_accuracy': np.round(accuracy_score(y_fit,ts_pred),2),
49         'test_precision': np.round(fit_prec,2),
50         'test_sensibility': np.round(fit_sens,2),
51         'test_specificity': np.round(fit_spec,2),
52         'test_f1': np.round(fit_f1,2),
53
54         'eval_accuracy': np.round(accuracy_score(y_holdout, ev_pred),2),
55         'eval_precision': np.round(eval_prec,2),
56         'eval_sensibility': np.round(eval_sens,2),
57         'eval_specificity': np.round(eval_spec,2),
58         'eval_f1': np.round(eval_f1,2),
59     }
60     for k,v in history_dict.items():
61         print(k,v)
62     temp_res_df = pd.DataFrame([history_dict], columns=history_dict.keys())
63     history_df = pd.concat([history_df, temp_res_df], axis =0, sort= False)
64

```

Código 4.14: Código de Aplicacion de la RNA al Conjunto de Entrenamiento.

Se utiliza las funciones para las métricas que se guardan en un diccionario que posteriormente pasar a un marco de datos, donde se guardarán los 19 resultados correspondientes a las 19 corridas que realiza el proceso de clasificación con la RNA. Las tablas 4.5, 4.6, 4.7 contienen los resultados para el proceso de entrenamiento con la métrica de Exactitud para el entrenamiento y la validación de entrenamiento, los resultado de la evaluación del entrenamiento utilizado la predicción de clases con un



subconjunto del conjunto de entrenamiento y la última tabla contiene los resultados para la evaluación de la validación con un conjunto de datos desconocido para el modelo entrenado, cuyos resultados se muestran en la tabla 4.5.

<b>Entrada</b>	<b>Exactitud</b>	<b>Val. Exac.</b>
s1	0.75	0.79
s2	0.79	0.78
s3	0.8	0.69
s4	0.8	0.79
s5	0.8	0.84
s6	0.83	0.77
s7	0.8	0.79
s8	0.8	0.81
s9	0.83	0.78
s10	0.81	0.85
s11	0.83	0.8
s12	0.86	0.81
s13	0.84	0.79
s14	0.84	0.81
s15	0.83	0.82
s16	0.85	0.82
s17	0.84	0.81
s18	0.83	0.78
s19	0.88	0.78

Tabla 4.5: Tabla resultados exactitud de entrenamiento.

Con el fin de evaluar el entrenamiento que obtuvo la RNA y verificar si la misma generó un conocimiento que le permita realizar la clasificación de los casos de apnea en base a la información que se le entregó, se realizó la evaluación utilizando un subconjunto del mismo conjunto de entrenamiento obteniendo los resultados mostrados en la tabla 4.6.

Emtrada	Exactitud	Precisión	Sensibilidad	Especificidad	F1
s1	0.7	1	0.69	0.98	0.17
s2	0.83	0.63	0.91	0.86	0.69
s3	0.78	0.71	0.80	0.89	0.61
s4	0.75	0.56	0.81	0.91	0.53
<b>s5</b>	<b>0.87</b>	<b>0.81</b>	<b>0.89</b>	<b>0.94</b>	<b>0.77</b>
s6	0.76	0.52	0.82	0.94	0.48
s7	0.83	0.72	0.85	0.90	0.6
s8	0.85	0.52	0.97	0.88	0.65
s9	0.78	0.69	0.80	0.91	0.5
<b>s10</b>	<b>0.87</b>	<b>0.83</b>	<b>0.88</b>	<b>0.97</b>	<b>0.75</b>
s11	0.8	0.59	0.86	0.93	0.57
s12	0.83	0.86	0.82	0.99	0.69
s13	0.76	0.55	0.84	0.93	0.57
s14	0.76	0.55	0.84	0.96	0.57
s15	0.86	0.82	0.86	0.93	0.56
s16	0.79	0.35	0.88	0.94	0.36
s17	0.77	0.7	0.79	0.95	0.62
s18	0.82	0.86	0.81	0.91	0.67
s19	0.8	0.62	0.86	0.97	0.62

Tabla 4.7: Tabla de Resultados Evaluación de Validación de RNA.

Como última parte del proceso de Diseño e Implementación de la RNA, se realizó una evaluación con un subconjunto de entrenamiento que no se utilizó para entrenar la RNA, fue separado a modo de que pueda fungir como un tipo examen de conocimiento para el fenómeno estudiado utilizando el modelo previamente entrenado. Se realizaron las mismas métricas que en la evaluación del entrenamiento de la RNA obteniendo los resultados mostrados en la tabla 4.7.

<b>Entrada</b>	<b>Exactitud</b>	<b>Precisión</b>	<b>Sensibilidad</b>	<b>Especificidad</b>	<b>F1</b>
s1	0.79	0.50	0.08	0.98	0.14
s2	0.77	0.54	0.48	0.86	0.51
s3	0.79	0.65	0.53	0.89	0.58
s4	0.81	0.73	0.58	0.91	0.65
s5	0.85	0.77	0.60	0.94	0.68
s6	0.88	0.81	0.71	0.94	0.76
s7	0.86	0.72	0.75	0.90	0.73
s8	0.84	0.56	0.70	0.88	0.62
s9	0.82	0.76	0.64	0.91	0.69
s10	0.88	0.89	0.65	0.97	0.75
s11	0.89	0.76	0.74	0.93	0.75
s12	0.96	0.94	0.87	0.99	0.9
s13	0.9	0.78	0.84	0.93	0.81
s14	0.9	0.88	0.73	0.96	0.8
s15	0.88	0.78	0.72	0.93	0.75
s16	0.9	0.81	0.77	0.94	0.79
s17	0.88	0.79	0.65	0.95	0.71
s18	0.86	0.64	0.66	0.91	0.65
s19	0.96	0.92	0.94	0.97	0.93

Tabla 4.6: Tabla de Resultados Evaluación de Entrenamiento de RNA.

# Capítulo 5: Resultados y Análisis

En esta sección se tomarán en cuenta los resultados obtenidos y mostrado en el capítulo 4, se realizará una análisis de los mismos, de acuerdo al orden en que fueron presentados, desde la adaptación del hardware hasta los resultados finales del modelo de RNA utilizado para la clasificación de los casos de apnea.

## 5.1. Adaptación del Hardware

Una parte importante de la implementación y desarrollo del proyecto fue la adaptación de sensores capaces de obtener señales fidedignas y con la menor cantidad de ruido posible, recordando al capítulo anterior, se utilizaron dos tipo de sensores para la adquisición de la señal de ECG y SpO<sub>2</sub>.

Era importante la correcta adaptación del sensor de ECG, debido a la importancia que esta refleja en el procesamiento de las características ya que de ahí se obtienen la mayoría de los resultados de Frecuencia Respiratoria por lo que una señal de buena calidad es más que necesaria para dicha tarea.

Se realizó una prueba de adquisición de la señal cardíaca en electrodos convencionales de AgCl y los electrodos mínima invasión, se obtuvo la señal mostrada en la figura 4.5. Si se observa el despliegue de ambas señales, la morfología de la señal es completamente igual, lo cual indica que la adquisición de la misma tiene un grado mínimo de ruido con la presencia de rizo de movimiento entre la onda T y la onda P.

Debido al análisis que se realiza en este proyecto donde solamente se interesa en el complejo QRS, dicho rizo no representa una afectación severa al desempeño del algoritmo para la adquisición de las señales, sin embargo se debe tomar en

consideración que los electrodos secos al ser textiles, pueden llegar a presentar un deterioro dependiendo de la cantidad y el proceso de limpieza y lavado que se le da a la prenda diseñada.

## 5.2. Procesamiento Digital de las Señales

### 5.2.1. Procesamiento ECG

Tal y como se mencionó en la implementación del sistema, se aplicaron los procesos de procesamiento para las señales de ECG y SpO<sub>2</sub>, en la figura 4.12, se observa el resultado del procesamiento de la señal cardíaca utilizando el algoritmo implementado, se observa que la detección de picos Q, R y S, se realiza de manera efectiva siempre y cuando la calidad de la señal cardíaca no presente un ruido excesivo, por lo que es necesario el refinamiento de la implementación realizada para distinguir los picos de subida y bajada de la señal ante la presencia de ruido, se observó que con una señal con ruido abundante, se detectaba en algunos casos un pico falso en el que tomaba en cuenta como subida el inicio de la etapa de ruido y como bajada el final de dicha etapa.

Al estar en una posición intermedia del lapso de tiempo se tenían fallas en la duración de los lapsos RR reflejándose en la medida del Ritmo Cardíaco calculado, se recomienda agregar una etapa de verificación correcta de latidos en la cual se realiza la medición entre las subidas y bajadas de la pendiente de la derivada para identificar los lapsos con ausencia de complejos QRS y tener un estimado de la medida del Ritmo Cardíaco.

Se enfrentó ante un caso específico en el que el ruido presente en la señal no permite la detección de los picos del segmento QRS y por lo tanto alarga los segmentos y disminuye el cálculo de la Frecuencia Cardíaca, se revisó pero tendrá que ser refinado para garantizar una discriminación de dichos lapsos y poder generar una medición cien por ciento confiable, se intentó hacer la medición de manera consecutiva con lapsos de 1 segundo, obteniendo mediciones erróneas por lo que se concluye que es necesario un lapso mayor a 2 segundos para una correcta medición de la Frecuencia Cardíaca tomando como base el promedio de la duración de los segmentos RR.

El tipo de ruido presente en esos casos, corresponde a movimiento del paciente, se espera que existan ese tipo de señales debido a la manipulación de los pacientes neonatales por parte del personal de salud, siendo un fenómeno estocástico para el procesamiento de la señal pero que pueden ser abordado utilizando algún tipo de filtro adaptativo o verificando la ausencia de picos QRS.

### 5.2.2. Procesamiento SpO2

Tras la aplicación del procesamiento descrito para la señal de SpO2, se presentan los cambios graduales de la señal en la figura 4.14, en donde se observa como la señal debido a la forma de trabajo del sensor MAX30102 y la adquisición por medio de interrupciones se despliega como la resultante de una digitalización simple, el hecho del uso de las interrupciones significó una gran ventaja ya que permitió la armonización de la adquisición de la señal de SpO2 en conjunto con la señal cardíaca y la utilización del MCP3008, sin embargo, el entendimiento del funcionamiento del buffer de datos y la lectura en base a interrupción es algo confusa debido al movimiento de los apuntadores de lectura de la memoria del dispositivo, debido a que la cuenta de las muestras depende del número de bits en la adquisición es necesario la normalización de la señal para evitar fallos al trabajar con un rango acotado.

Al realizar el procesamiento de la señal de SpO2, es necesaria la inversión de la misma para tener una morfología natural representativa de la señal pulsátil que corresponde al movimiento arteriovenoso. Se obtiene el índice de índices que, por su naturaleza no tiene un significado numérico concreto, se realizó un proceso de caracterización del sensor, con más de 100 muestras y diversos sujetos de prueba, lo que permitió linealizar el comportamiento del sensor para obtener un valor de 0 a 100 % correspondientes a la concentración del oxígeno en la sangre, siendo necesario en un futuro al momento de realizar la conjunción de los distintos módulos la revisión del orden del filtro utilizado debido a la diferencia en la frecuencia de muestreo que hubo en la prueba de adquisición de datos y la frecuencia de muestreo final de 200 Hz y la frecuencia de muestreo de los datos obtenidos de Physionet.

Este proceso es importante debido a que la caracterización dependerá entera-

mente de la calibración y estado del sensor base en el que se realiza la relación de porcentaje de oxigenación con la cantidad RR presentada en la implementación de dicha parte del código, para este proceso se utilizó un dispositivo nuevo para evitar riesgos en medición por desgaste o trabajo anterior. La estructura física del sensor de SpO2 incluye la adaptación del actuador vibratorio, el cual cuenta con la potencia suficiente para transmitir la vibración hacia el paciente, se considera que este puede ser mejorado e incluso adaptar un modelo de sensor basado el polimero elástico para evitar algún tipo de roce por la banda que sostiene el sensor a la extremidad del paciente neonatal.

### **5.2.3. Procesamiento Señales Respiratorias EDR**

Una parte importante y que tiene gran relevancia en el cálculo de las características utilizadas en la formación del Conjunto de Entrenamiento, es la robustez de los algoritmos desarrollados para la detección de picos, ya que las medidas respiratorias dependen enteramente de la calidad de las señales de ECG y SpO2; se decidió realizar este tipo de aproximaciones a la medición de la Frecuencia Respiratoria debido a la peligrosidad de aplicación de una corriente eléctrica externa para la medición de la Impedancia Torácica.

En la implementación de esta parte del código se tuvo especial atención debido al tipo de interpolación en conjunto con la aplicación del filtro IIR, verificando el ancho de banda y el orden del filtro a utilizar ya que la morfología de la señal resultante se veía afectada en su totalidad teniendo en algunos casos una línea recta debido a un filtro de un orden muy alto, en el caso de la versión RSA de la señal respiratoria obtenida se obtuvo un mejor desempeño utilizando un filtro FIR de un orden alto para evitar los de recursividad de la función de transferencia natural de los filtros digitales.

En la figura 4.21, se tiene el despliegue de las 4 señales respiratorias derivadas calculadas, se observa como la forma de la señal es uniforme para las versiones en RS, QRS y PPG, debido a que la amplitud de las señales originales se encontraba modulada y se usan los picos en contrario con la señal RSA que utiliza un filtrado directo sobre la señal de ECG para obtener una envolvente pero que no cuenta con

una interpolación como las señales restantes y que tendrá una morfología distinta pero una periodicidad parecida a las otras calculadas; dicha periodicidad es la obtenida con el procesamiento del polo menor en el cálculo de la Frecuencia Respiratoria, importante mencionar que este cálculo se realizó en las medidas de 1 segundo de señal en los lapsos de 5 segundos debido a que el tramo corresponderá a una ecuación de una aproximación de una onda senoidal, para su posterior fusión en el Filtro Kalman.

El filtro Kalman como se describió en la implementación necesita de un conjunto de valores y de entradas a partir de las cuales se utiliza una metodología estadística basado en las ecuaciones presentadas y la covarianza entre las diferentes mediciones de Frecuencia Respiratoria en el lapso de 5 segundos que se analizó; inicialmente se consideraron los lapsos enteros de 5 segundos pero se observó que era necesario la segmentación del lapso para el uso apropiado del Filtro Kalman y la generación de los residuos y la señal calculada.

En la figura 4.23 se observa el resultado gráfico de los cálculos para las estimaciones de la Frecuencia Respiratoria posteriores a la Fusión de la señal, así como los resultados de las frecuencias calculadas en el filtro kalman y las medidas observando que las variaciones entre cada una en  $rrFO$  y  $rrF1$ , son mínimas teniendo casos en donde se cumple el cálculo debido a la variación que se tiene en el cálculo de la Frecuencia Respiratoria en RSA con respecto a las demás como lo que se observa alrededor del segundo 150, donde se tiene una caída de la estimación de la Frecuencia Respiratoria de RSA con respecto a las demás.

De manera numérica se hizo el análisis del error presente en la aplicación del procesamiento enteros de las señales con los resultados presentados en la tabla 4.1, en donde se observa que el algoritmo de cálculo de la Frecuencia Cardíaca tiene un error aproximado de un 1%, el Pulso Cardíaco calculado tiene un error del 7% mientras que se observa una disminución en el error al realizar la fusión de los valores calculados para la Frecuencia Respiratoria, llegando a 27 y 28%, debido a lo susceptible que es el cálculo derivado del ruido presente en la señal de origen se considera que la fusión de los valores es necesaria ya que disminuye el error cerca de un 10% y arroja un resultado más confiable con respecto a los valores de control en las tablas de numerales para el paciente de prueba.



#### 5.2.4. Conjunto de Entrenamiento y RNA

Durante la formación del Conjunto de Entrenamiento se tienen diversos casos detectados de Apnea, como el presente en la figura 4.26, analizando 5 minutos previos a la caída de la SpO<sub>2</sub>, y se observa que existe una disminución del Ritmo Cardíaco un poco después del segundo 250, el cual es corroborado por la caída del Pulso Cardíaco y una disminución en la Frecuencia Respiratoria, detectada posterior a una interrupción de la onda natural de respiración descrita en PI, la cual origina el alargamiento de la duración de los segmentos RR, que ocasionará una disminución del Ritmo Cardíaco previo y la falta de saturación de oxígeno en el cuerpo, es importante denotar este fenómeno, debido a que es necesaria la selección de casos donde se esté detectando una falsa alarma o si el caso posible de apnea, realmente sea un caso de apnea.

Esta preclasificación basada en los hechos descritos para el caso presentado en la figura así como también para lo descrito en las condiciones para la clasificación de una apnea, y que en caso positivo debe ser corroborado por el usuario, se realizó de manera local, se precisa que para un mejor desempeño como valoración inicial de la clasificación se utilice la opinión de un personal experto en Neonatología o bien que pueda arrojar un mejor entendimiento de las señales ya que se podía considerar una Falsa Alarma, una Apnea Verdadera o dejar un indicativo de decisión insegura para su valoración posterior. Finalmente no se tomaron en cuenta los casos inseguros en la clasificación y la jerarquía de las características utilizadas en la entrada del modelo inteligente.

Antes del ordenamiento se consideró a *sqi\_ppg* uno de los principales componentes debido a la susceptibilidad del sensor de SpO<sub>2</sub> de registrar fallos o ruido en la transmisión de la señal, esto debido a que, como ya se explicó en la descripción del sensor MAX30102, la utilización de un medio óptico, puede verse interrumpido y presentar ruido ante el movimiento o la distancia de la fuente de luz con el cuerpo que interactúa.

En segundo lugar tiene al mínimo en HR, tal y como es la naturaleza del fenómeno en estudio el ritmo cardíaco juega un papel determinante debido a las variaciones que tiene en la presencia de una apnea con la presencia de bradicardia, las pendientes del

Pulso y Ritmo Cardíaco también serán un indicativo de cómo ha sido la variación ya que al tener una pendiente negativa se estaría considerando una caída en la medición de ambas características; las Fusiones Respiratorias tal y como se observó en la tabla 4.1 minimizó el error obtenido en el cálculo de la Frecuencia Respiratoria por lo que se esperaría que tuviera una relevancia interesante en los primeros puestos del orden de características.

Una vez realizado el entrenamiento y la validación del modelo de la RNA, en la tabla 4.5 se muestran los resultados en Exactitud de Entrenamiento y la Validación de dicha Exactitud, donde se observa un máximo en validación de entrenamiento del 85 % en la medida número 10, por lo que se podría considerar como el subconjunto de características que tiene un mejor desempeño en la clasificación de los casos de apnea, seguido el subconjunto con 5 características, y teniendo el conjunto más bajo con 3 características; es importante recordar la naturaleza aleatoria de los datos, ésto debido a que la combinación entre el acomodo aleatorio y el orden obtenido en la selección de características permite una generalización de los casos más no completamente exacta.

En la tabla 4.6 se observa que la red tiene un mejor desempeño con la utilización de 12, 13, 14, 16, y 19 características con las que tuvo una exactitud de 90 % o superior, se puede observar como la red tiene una especificidad buena ante casos de Falsa alarma pero una sensibilidad a casos de Apneas Verdaderas con un desempeño máximo de un 94 % utilizando todas las características seguido de un 87 % utilizando solamente 12 características, se espera que tenga buenos resultados en la clasificación de estos casos, ya que el conocimiento adquirido por la RNA tiene como base las características y los casos que ya ha observado con anterioridad.

El valor F1 permite observar la correlación existente entre la precisión y a sensibilidad ante la presencia de un conjunto de entrenamiento desbalanceado, es decir que no se tenga el mismo número de casos positivos como negativos, se observa que los valores más altos corresponden a los valores más altos en dichas métricas, con un valor cercano a 1 o 100 %.

En la tabla 4.7, se observa que el modelo tiene una mayor exactitud al utilizar 5 y 10 de las características previamente ordenadas con un valor de 0.87 u 87 %; se tiene el valor mínimo en exactitud solamente utilizando la primer característica con

un valor de 0.7 o 70 %, la precisión del model otiene un valor máximo en .86 con la utilización de 12 y 18 variables, una sensibilidad máxima de un 0.97 con 8 variables y una especificidad máxima de .99 para 12 variables, mientras que para los valores seleccionados (5 y 10) con mayor exactitud corresponden a .81 y .83 en Precisión,.89 y .88 en Sensibilidad, .94 y .97 en Especificidad y un valor de .77 y .75 en la relación de Precisión - Sensibilidad o Valor F1.

Considerando que tienen la misma Exactitud utilizando 5 y 10 características y que la diferencia entre las demás métricas obtenidas en la evaluación de la RNA con un valor menor a un .5 o 5 %, se puede considerar válido el uso de cualquiera de estas dos opciones. De acuerdo a lo reportado por Monasterio et al.[30] se tiene una diferencia en Exactitud de un 3 % al reportar 90.2 % , una diferencia en Sensibilidad superior de un 2 y 3 % al reportar 86 %, una Especificidad mayor en 4 % al reportar 91 % y una Precisión superior al reportar un 75 % de Precisión. Estas diferencias pueden ser atribuibles a la naturaleza de los propios datos utilizados en su formación del Conjunto de Entrenamiento, la Selección de Características y en el método de IA utilizado para la clasificación de los casos de Apnea al implementar una Máquina de Soporte Vectorial en comparación con lo realizado en esta tesis en la implementación de una RNA.

De acuerdo a los resultados la utilización exclusivamente de la señal más significativa (S1:sqi\_ppg) solamente permite la clasificación de un 70 % de los casos utilizados para la evaluación con una Precisión máxima y un alto grado de Especificidad, si se agregará la segunda característica esencial para la determinación de la apnea neonatal (S2:min\_hr) se lograría aumentar la exactitud del algoritmo pero se perdería campo en las métricas de Precisión y Especificidad, ganando terreno en la Sensibilidad aumentando el valor de la misma.

Del mismo modo se observa cómo los valores de medición de la Frecuencia Respiratoria para la primer mejor combinación de características no tienen una gran relevancia ya que solamente se utilizan los valores de las fusiones realizadas de los valores de EDR RR; sin embargo, para aumentar la Exactitud y la Especificidad si se considerara utilizar más de 5 características, sería necesario utilizar parte de estas Frecuencias Respiratorias. Si se analiza los resultados presentados en la tabla 4.1, se observa como el error disminuye al momento de realizar la Fusión de las

Respiraciones, lo que garantizará un mejor cálculo de dicha variable.

# Capítulo 6: Conclusiones y Trabajo a Futuro

## 6.1. Conclusiones de la Tesis

El proyecto desarrollado en la presente Tesis consistió en el diseño e implementación de un Sistema de Monitoreo Inteligente para la Detección de Apneas Neonatales en pacientes con gestación completa y pre término, teniendo consideraciones especiales tales como la construcción del hardware necesario para que fuera lo menos invasivo posible, así como, que tuviera la capacidad de tomar una decisión ante la presencia de un posible caso de apnea. Se desarrolló la problemática y se entendieron las variables clave para la detección y clasificación de los eventos de apnea, así como la propuesta de solución basada en la literatura disponible.

Se realiza el modelado global del sistema siguiendo el modelo de 4+1 vistas con la descripción de los Diagramas de Contexto nivel 0 y 1, Diagrama de Casos de Uso con descripción, Diagrama de Actividades, Diagrama de Clases, Diagrama de Componentes y Despliegue, agregando un modelo de datos y la presentación gráfica de la Arquitectura del Sistema.

Se decide utilizar una Raspberry Pi 3 como entorno de desarrollo para el proyecto debido a las aplicaciones conocidas con diversos sistemas embebidos utilizando dicha tecnología y la capacidad de la utilización de sensores. Durante el desarrollo del hardware se utilizó un sensor de oxígeno con conexión de tipo I2C que permitió la interacción con el sensor de ECG de una manera favorable debido a su funcionamiento por medio de interrupciones que fue aprovechado para la adquisición simultánea de las señales Electrocardiográficas a la misma frecuencia de muestreo.

Se hace una aportación pública al desarrollar una librería para el uso del sensor MAX30102 con la Raspberry Pi 3 en lenguaje Python. La adquisición de la señal fue correcta y permitió un procesamiento sencillo, sin embargo, se tiene la necesidad del diseño de un formato de presentación específico para el sensor debido a la gran facilidad de contaminación de la señal con movimientos corporales y la adquisición por medio óptico. No se considera que tenga mucha relevancia el tipo de sensor utilizado en comparación su otra versión de tipo clip para pacientes neonatales.

Para la adquisición de las señales cardíacas se desarrolló un método sin pegamento, es decir no se utilizan electrodos convencionales, si no que, se desarrolla un sensor vestible para la toma de señal del corazón, al realizar las pruebas de funcionamiento despues de la creación de dicho sensor, se verifica la fidelidad de la señal obtenida antes y despues del procesamiento digital para garantizar una señal con la forma natural del ECG y que no presenta ruido del sensor, se concluye que la utilización de ese sensor para la adquisición de la señal cardíaca es favorable debido a la gran calidad con la que se adquiere la señal y que cumple con la especificación original de no utlizar pegamento, sin embargo, puede llegar a ser ineficaz dependiendo del tamaño del paciente a analizar y el estado de la prenda modificada.

Posterior al desarrollo del hardware se realizaron los algoritmo procesamiento digital de las señales cardíacas, en donde se realiza la adaptación de un algortimo capaz de realizar al detección de picos, que tuvo un buen desempeño en la gran mayoría de los casos. Para el cálculo del porcentaje de Saturación Parcial de Oxígeno, fue necesario analizar la señal en sus componentes transitorios y estacionarios pero fue necesaria la utilización de un sensor externo para la determinación de la concentración de Oxígeno diluido en la sangre, concluyendo que, el proceso de obtención de la medida de porcentaje de oxígeno debe tomarse con respecto a varias mediciones con un sistema de control y con distintos sujetos ya que no todas las personas tendrán cantidad exactamente iguales debido a la carga química sanguínea de cada uno.

La aproximación de las señales respiratorias a partir de señales secundarias como el ECG o PPG, es una tarea complicada y de mucho análisis, se tuvieron problemas con las aproximación en señales con mucho ruido por lo que se concluye que no siempre será la opción más favorable de aproximación debido a la dependencia que

tiene con el índice de calidad o la cantidad de ruido presente en las señales de donde fue derivada, los índices de señal de calidad no son meramente exclusivos de este tipo de señales si no que se pueden adaptar a distintas formas como el Electro Encéfalograma, este índice de calidad arroja una buena aproximación de acuerdo a que tan cercano a una señal senoidal es la morfología de la señal analizada, convirtiéndose en una de las características esenciales en el desarrollo del proyecto.

Siguiendo con el desarrollo del Sistema Inteligente se encuentra la formación del Conjunto de Entrenamiento, este proceso fue en punto de convergencia de los distintos algoritmos desarrollados durante la adquisición y procesamiento de las señales, se concluye que la correcta conformación de un conjunto de entrenamiento es una tarea clave en el desarrollo de un sistema inteligente, ya que se debe tomar en cuenta las fuentes de la información, los pacientes y la manera de procesar la información obtenida,

Las características incluidas en este conjunto de entrenamiento, fueron seleccionadas basándose en la literatura disponible, el proceso de formación del conjunto de entrenamiento fue en su totalidad una implementación original, dentro del trabajo a futuro para este algoritmo se considera la verificación de sus componentes y la comparación con algoritmos o procesos parecidos, para su mejora, esencialmente en el procesamiento de las señales y el cálculo del análisis de Curva ROC, finalmente se desarrolla un conjunto de entrenamiento que no era balanceado en sus clasificaciones, situación que fue planeada con la finalidad de obtener un tipo de respuesta de respues a partir del algoritmo inteligente.

Por último, la implementación de un modelo de Red Neuronal Artificial no es un proceso sencillo, debido a que la naturaleza del fenómeno estudiado no permite una generalización de una estructura neural para su análisis, se decide la utilización de un modelo Secuencial debido a la naturaleza de los datos discretos, se considera y se tiene evidencia que es posible la solución del problema utilizando otro tipo de modelos, como las máquinas de soporte vectorial y las Redes Neuronales Convolucionales, siendo el uso de la Red Neuronal Secuencial la forma distintiva de este proyecto para generar una solución; se realizaron pruebas con distintos modelos, concluyendo que, un modelo de RNA puede ser tan simple o tan complejo como el diseñador lo prefiera sin embargo se tienen distintas limitantes, ya que mientras más complejo

sea el modelo y menos general sea la información se corre el riesgo de sobre entrenar el modelo o bien, que falte información para que genere el conocimiento necesario para su utilización final.

La respuesta obtenida fue cercana a lo reportado por las fuentes citadas en una métrica pero cercanamente superior en distintas más se concluye que un modelo de análisis inteligente puede tener distintas aproximaciones para su desarrollo y evaluación. a su vez que dependiendo de la naturaleza, la conformación del Conjunto de Entrenamiento y de la selección de las características en conjunto con el número de dimensiones de entrada, se verá afectadas las métricas que pueden ser significativas como en el caso de la exactitud y el valor calculado F1 para Conjuntos de Entrenamiento no balanceados.

De manera general se concluye que el proyecto tuvo un resultado satisfactorio ya que se obtuvo un resultado con una variación de menos del 5% con respecto a lo reportado por la experimentación base, superando a su vez las evaluaciones en otro tipo de métricas.

También que el modo de aproximación a la problemática con la utilización de señales secundarias y sensores vestibles, dan al proyecto una innovación que permite que el paciente resulte con menos daños corporales o fatales y que permite la reutilización de sensores con lo que se disminuiría el costo por operación del sistema, y con la utilización de sistemas inteligentes se tendrá una mayor seguridad al momento de el diagnóstico de la apnea neonatal lo que permitirá que el personal médico pueda confiar en la tecnología desarrollada y que el desarrollo de la Ingeniería Biomédica avance con la confianza de sus usuarios.



## 6.2. Trabajo a Futuro

Como trabajo a Futuro se contempla el refinar la sensibilidad de los algoritmos para buscar una manera de contrarrestar el efecto que tienen los ruidos y perturbaciones externas en el procesamiento de las señales así como la conjunción de el hardware desarrollado con el sistema inteligente y el diseño general del software, para proceder a la realización de pruebas ya se tiene un avance en la implementación del sistema en un ambiente embebido pero deberá continuar hasta que pueda ser liberado a una fase de pruebas y la mejora de la presentación de los sensores y mejoramiento de la implementación de hardware, en caso necesario posterior a la fase de pruebas la realización de cambios necesarios en el algoritmo o modelo inteligente.

## Bibliografía

- [1] O. M. de la Salud, “Reducción de la mortalidad de recién nacido,” 2016.
- [2] C. T. Tolomeo, *Nursing Care Nursing Care in Pediatric Respiratory Disease*. Wiley.Blackwell, 2012.
- [3] J. G. Webster, J. W. Clark, M. R. Neuman, W. H. Olson, R. A. Peura, F. P. Primiano, M. P. Siedband, and L. A. Wheeler, *Medical instrumentation: Application and design*. 2016-10-25: John Wiley & Sons, 2010.
- [4] I. M. d. S. Social, *Diagnóstico y Tratamiento de la Apnea del Prematuro*. Instituto Mexicano del Seguro Social, 2015.
- [5] G. Salas, G. Pérez, G. Centeno, and M. Alvarez, “Consenso sobre la monitorización del recién nacido internado. Parte 1: Monitorización no invasiva del recién nacido,” *Archivos Argentinos de Pediatría*, vol. 111, no. 4, pp. 353–359, 2013.
- [6] R. Aggarwal, “Apnea in the newborn,” *Indian Journal of Pediatrics*, vol. 68, no. 10, pp. 959–962, 2001.
- [7] CONSEJO DE SALUBRIDAD GENERAL, “Diagnóstico Y Tratamiento De Apnea Del Prematuro.” *Cenetec*, 2014. [Online]. Available: [www.cenetec.salud.gob.mx](http://www.cenetec.salud.gob.mx)
- [8] J. P. Rodríguez, M. C. Lafuente, and A. M. S. Torres, “Apnea en el periodo neonatal,” *Junta Directiva de la Asociación Española de Pediatría*, p. 37, 2002.
- [9] T. Sadler, “Embriología Médica,” 2012.

- [10] J. S. Martín and D. S. Caussade, “Evaluación funcional de la vía aérea,” *Neumologia Pediatrica*, vol. 7, no. 2, pp. 61–66, 2012.
- [11] G. A. C. Hall, John, *Tratado de Fisiología Médica*, 12th ed. Ibiza, España: Elsevier Inc., 2016, vol. 31, no. 2.
- [12] J. Keener and J. Sneyd, *Mathematical Physiology Systems Physiology*. Springer, 2009, vol. 8/II.
- [13] R. M. Rangayyan, *Biomedical signal analysis*. IEEE Press Series, 2015.
- [14] G. Clifford, *Advanced method and tools for ECG data analysis*. Artech House, 2006.
- [15] S. Lopez, “Pulse Oximeter Fundamentals and Design,” Tech. Rep., 2012.
- [16] R. Khandpur, *Biomedical Instrumentation, Technology and Application*, 2nd ed. Mc Graw Hill, 2003.
- [17] D. S. Pupo, “ECG-DERIVED RESPIRATION IN ATRIAL FIBRILLATION,” 2007.
- [18] S. Cerutti and C. Marchesi, *Advanced Methods of Biomedical Signal Processing*. IEEE Press Series, 2011.
- [19] J. Semmlow, *Cicuits, Signals and Systems for Bioengineers*, 2nd ed. Elsevier, 2005.
- [20] A. y.k. chan, *Biomedical Device Technology Principle and Design*, 2008.
- [21] T. Li, *Digital Signal Processing Fundamentals and Applications*.
- [22] A. Devices, “Single-Lead , Heart Rate Monitor Front End AD8232 \* Product Page Quick Links,” Norwood, MA, 2013.
- [23] M. Integrated, “High-Sensitivity Pulse Oximeter and Heart-Rate Sensor for Wearable Health MAX30102 High-Sensitivity Pulse Oximeter and Heart-Rate Sensor for Wearable Health Absolute Maximum Ratings,” 2017.

- [24] R. M. M. José Tomás Palma Méndez, *Inteligencia artificial*, 2013.
- [25] D. L. Hudson and M. E. Cohen, *Neural Networks and Artificial Intelligence for Biomedical Engineering*. IEEE Press Series, 1999, vol. XXXIII, no. 2.
- [26] S. Raschka, *Python Machine Learning*. PACKT publishing, 2015, no. 1.
- [27] E. d. J. Rendón Luna, “Entrenamiento Evolutivo De Redes Neuronales Dinámicas Para La Detección De Apnea-Bradycardia En Neonatos Prematuros,” Ph.D. dissertation, UNIVERSIDAD SIMON BOLIVAR, 2013.
- [28] M.-m. Nano, “Automated Detection of Central Apnea in Preterm Infants,” Ph.D. dissertation, Delft University of Technology, 2015.
- [29] R. D. Shirwaikar, U. D. Acharya, K. Makkithaya, M. Surulivelrajan, and L. E. S. Lewis, “Machine learning techniques for neonatal apnea prediction,” *Journal of Artificial Intelligence*, vol. 9, no. 1-3, pp. 33–38, 2016.
- [30] V. Monasterio, F. Burgess, and G. D. Clifford, “Robust classification of neonatal apnoea-related desaturations,” *Physiological Measurement*, vol. 33, no. 9, pp. 1503–1516, 2012.
- [31] L. Hattersley, *The Official Raspberry Pi Beginner’s Book*, 2017.
- [32] S. Bouwstra, W. Chen, and L. Feijs, “Smart Jacket Desing for Neonatal Monitorin with Wearable Sensors,” *IEEE computer society*, pp. 162–169, 2009.
- [33] M. T. Rose-Gómez, César Enrique Serna-Encinas, “Procesamiento del Electrocardiograma para la Detección de Cardiopatías,” *Encuentro Nacional de Computación*, pp. 1–4, 2014.
- [34] W. Zong, T. Heldt, G. Moody, and R. Mark, “An open-source algorithm to detect onset of arterial blood pressure pulses,” *Computers in Cardiology, 2003*, pp. 259–262, 2003. [Online]. Available: <http://ieeexplore.ieee.org/document/1291140/>

- [35] C. O'Brien and C. Heneghan, "A comparison of algorithms for estimation of a respiratory signal from the surface electrocardiogram," *Computers in Biology and Medicine*, vol. 37, no. 3, pp. 305–314, 2007.
- [36] D. Widjaja, J. Taelman, S. Vandeput, M. A. Braeken, R. a. Otte, B. R. V. D. Bergh, and S. V. Huffel, "ECG-derived respiration: Comparison and new measures for respiratory variability," *2010 Computers in Cardiology*, pp. 149–152, 2010.
- [37] G. B. Moody, R. G. Mark, A. Zoccola, and S. Mantero, "Derivation of respiratory signals from multi-lead ECGs," *Computers in Cardiology 1985*, vol. c, pp. 113–116, 1985. [Online]. Available: <http://www.physionet.org/physiotools/edr/cic85/>
- [38] S. Nemati, A. Malhotra, and G. Clifford, "Data Fusion for Improved Respiration Rate Estimation," *EURASIP Journal on Advances in Signal Processing*, vol. 2010, no. 1, p. 926305, 2010.
- [39] E. Gil, J. María Vergara, and P. Laguna, "Detection of decreases in the amplitude fluctuation of pulse photoplethysmography signal as indication of obstructive sleep apnea syndrome in children," *Biomedical Signal Processing and Control*, vol. 3, no. 3, pp. 267–277, 2008.
- [40] B. Hjorth, "EEG analysis based on time domain properties," *Electroencephalography and Clinical Neurophysiology*, vol. 29, no. 3, pp. 306–310, 1970.
- [41] G. Welch and G. Bishop, "An introduction to the Kalman Filter," *IEEE Power and Energy Magazine*, vol. 13, no. 6, pp. 50–59, 2015.
- [42] A. Goldberger, L. Amaral, L. Glass, J. Hausdorff, P. Ivanov, R. Mark, J. Mietus, G. Moody, C. kang Peng, and H. Stanley, "PhysioBank, PhysioToolkit, and PhysioNet: Components of a New Research Resource for Complex Physiologic Signals," *Circulation Electronic Pages*, vol. 101, no. 23, pp. 215–220, 2000.

- [43] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten, “The WEKA data mining software: an update,” *SIGKDD Explorations*, vol. 11, no. 1, pp. 10–18, 2009.
- [44] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng, “TensorFlow: Large-scale machine learning on heterogeneous systems,” 2015, software available from tensorflow.org. [Online]. Available: <https://www.tensorflow.org/>
- [45] F. Chollet *et al.*, “Keras,” <https://keras.io>, 2015.
- [46] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, “Scikit-learn: Machine learning in Python,” *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.