



UNIVERSIDAD DE LOS ANDES  
FACULTAD DE INGENIERÍA  
POSTGRADO EN INGENIERÍA BIOMÉDICA

DISEÑO E IMPLEMENTACIÓN DE UN DISPOSITIVO PARA  
LA MEDICIÓN DE LA VELOCIDAD DE LA ONDA DE  
PULSO POR FOTOPLETISMOGRAFÍA

Ing. Hugo David Reyes Amell

UNIVERSIDAD DE LOS ANDES  
FACULTAD DE INGENIERÍA  
POSTGRADO EN INGENIERÍA BIOMÉDICA

DISEÑO E IMPLEMENTACIÓN DE UN DISPOSITIVO PARA  
LA MEDICIÓN DE LA VELOCIDAD DE LA ONDA DE  
PULSO POR FOTOPLETISMOGRAFÍA

Trabajo de Grado presentado como requisito parcial para optar al título de Magíster  
Scientiae en Ingeniería Biomédica.

Ing. Hugo David Reyes Amell

Tutor: Prof. Ruben Medina

Mérida, 24 de septiembre de 2016

C.C.Reconocimiento

# AGRADECIMIENTOS

A Dios Todopoderoso por acompañarme siempre y brindarme tantas bendiciones.

A mi madre, por ser el pilar fundamental para el logro de esta meta y de muchas más a lo largo de mi vida. Gracias por amarme tanto y por tu apoyo incondicional ante la lucha por el cumplimiento de mis sueños.

A mi padre, por tantas enseñanzas, consejos, asesorías y palabras de motivación a lo largo del camino.

A la ilustre Universidad de los Andes, por cambiarme la vida en muchos aspectos con todo lo que ofrece. Gracias por los mejores años de mi vida hasta ahora, tanto en el ámbito académico como en el personal.

A mi novia, por acompañarme siempre en los momentos malos y buenos. Por darme ánimos cuando estuve desmotivado, por cuidarme cuando estuve enfermo y por todos los lindos momentos que hemos compartido.

A mis amigos del postgrado, en especial a Alejandra Delgado por todos los consejos de vida que me diste y a Jerick Órdenes por tu buena disposición de ayudarme siempre, por tus enseñanzas en Latex y ayudarme con asuntos de programación en general. ¡A ambos muchas gracias!

A mi también amigo del postgrado Alberto Medrano por su asesoría en el diseño y construcción de los circuitos impresos.

A mi tutor Ruben Medina por sus oportunas y detalladas asesorías en el desarrollo del presente trabajo.

Al médico cariólogo Mario Perez por su asesoría en cuanto a la parte médica del presente trabajo.

Hugo David Reyes Amell

**Hugo David Reyes Amell.** Diseño e implementación de un dispositivo para la medición de la velocidad de la onda de pulso por fotopleletismografía. Universidad de Los Andes. Tutor: Prof. Ruben Medina. Septiembre, 2016.

## Resumen

En el presente trabajo se detalla el diseño e implementación de un dispositivo para la medición de la velocidad de la onda de pulso (VOP) por fotopleletismografía. El diseño del dispositivo incluyó etapas de filtrado, amplificación, muestreo y conversión de señales, así como el desarrollo de una interfaz gráfica de usuario que abarcó comunicaciones con un circuito externo y la recepción, procesamiento y almacenamiento de los datos. Para el cálculo de la VOP se propusieron tres algoritmos de identificación de puntos: puntos máximos, tangente máxima y segunda derivada máxima. La selección del algoritmo se realizó mediante la evaluación de la dispersión de los resultados de un total de treinta (30) mediciones hechas en tres (3) pacientes. Se seleccionó el algoritmo de segunda derivada máxima. Adicionalmente, se realizó un segundo lote de mediciones a 10 pacientes (3 mediciones por paciente) en manera de comparar los resultados de VOP con valores de referencia. Los valores obtenidos estuvieron dentro de los intervalos provistos en investigaciones antecedentes [1, 2].

**Descriptoros:** Instrumentación biomédica, Velocidad de onda de pulso (VOP), Rigidez arterial, Fotopleletismografía.



## ÍNDICE GENERAL

AGRADECIMIENTOS . . . . .	i
RESUMEN . . . . .	ii
ÍNDICE . . . . .	iii
ÍNDICE DE TABLAS . . . . .	vi
ÍNDICE DE FIGURAS . . . . .	vii
<b>Capítulo</b>	<b>pp.</b>
<b>1. PLANTEAMIENTO DEL PROBLEMA . . . . .</b>	<b>1</b>
1.1 Antecedentes . . . . .	3
1.2 Objetivos . . . . .	4
1.2.1 Objetivo General . . . . .	4
1.2.2 Objetivos Específicos . . . . .	4
1.3 Justificación e Importancia . . . . .	5
1.4 Estructura . . . . .	5
<b>2. MARCO TEÓRICO . . . . .</b>	<b>7</b>
2.1 Aparato circulatorio . . . . .	7
2.1.1 Anatomía y fisiología de las grandes arterias . . . . .	7
2.1.2 Propiedades mecánicas de las arterias elásticas . . . . .	8
2.1.3 Arterias coronarias y su función . . . . .	9
2.1.4 Fisiología del movimiento de las arterias elásticas (Onda de Pulso) . . . . .	10
2.2 Velocidad de la onda de Pulso . . . . .	11
2.3 Fotopletismografía . . . . .	13
2.4 El electrocardiograma . . . . .	15
<b>3. METODOLOGÍA . . . . .</b>	<b>18</b>
3.1 Diagrama de bloques general . . . . .	18
3.2 Selección de sensores por fotopletismografía . . . . .	19
3.3 Amplificador de transimpedancia . . . . .	21
3.4 Circuito de control de LEDs . . . . .	21
3.5 Filtros analógicos y amplificadores . . . . .	23
3.6 Microcontrolador . . . . .	24
3.7 Convertidor Digital Analógico . . . . .	25
3.8 Circuito Amplificador ECG . . . . .	27

3.9	Circuito General . . . . .	28
3.10	Circuito de aislamiento . . . . .	28
3.11	Diseño del PCB . . . . .	33
3.12	Programación del Arduino . . . . .	34
3.12.1	Subrutina de calibración . . . . .	36
3.12.2	Subrutina de Muestreo . . . . .	39
3.12.3	Comunicación serial (protocolo RS-232) . . . . .	40
3.13	Programación en el computador personal . . . . .	42
3.13.1	<i>Back-end</i> . . . . .	43
3.13.1.1	Comunicación serial (protocolo RS-232) . . . . .	44
3.13.1.1.1	<i>Handshake</i> . . . . .	45
3.13.1.1.2	Calibración . . . . .	45
3.13.1.2	Subrutina de Muestreo . . . . .	47
3.13.1.3	Manejo de archivos . . . . .	51
3.13.1.4	Detección de puntos máximos en la señal electrocardiográfica . . . . .	53
3.13.1.5	Determinación de puntos de medición en las señales de pulso . . . . .	59
3.13.1.5.1	Algoritmo de puntos máximos . . . . .	60
3.13.1.5.2	Algoritmo de tangente máxima . . . . .	61
3.13.1.5.3	Algoritmo de segunda derivada máxima . . . . .	65
3.13.2	<i>Front-end</i> (Interfaz gráfica de usuario) . . . . .	67
3.13.2.1	Ventana de solicitud de dirección de base de datos . . . . .	68
3.13.2.2	Ventana principal . . . . .	69
3.13.2.3	Ventana de formulario . . . . .	71
3.13.2.4	Panel de adquisición de datos . . . . .	73
3.13.2.4.1	Calibración . . . . .	74
3.13.2.4.2	Muestreo . . . . .	76
3.13.2.5	Cálculo de la VOP . . . . .	77
3.13.2.6	Almacenamiento de archivos . . . . .	78
3.13.2.7	Registro de pacientes . . . . .	79
3.14	Evaluación de los algoritmos de identificación de puntos de medición . . . . .	81
3.14.1	Comparación con resultados manuales . . . . .	83
3.14.2	Análisis de similitud de resultados de VOP por periodos . . . . .	85
3.14.3	Análisis de dispersión de VOP promedio . . . . .	86
3.14.4	Eliminación de valores atípicos . . . . .	86
3.15	Comparación de mediciones de VOP promedio con valores de referencia . . . . .	87
<b>4.</b>	<b>RESULTADOS Y ANÁLISIS DE RESULTADOS . . . . .</b>	<b>90</b>

4.1	Resultados de comparación de procedimiento manual con los algoritmos propuestos . . . . .	91
4.2	Resultados del análisis de similitud de resultados de VOP por periodos . .	93
4.2.1	Resultados de paciente 1 . . . . .	93
4.2.2	Resultados de paciente 2 . . . . .	94
4.2.3	Resultados de paciente 3 . . . . .	94
4.2.4	Resultados de dispersión con todos los pacientes . . . . .	95
4.3	Resultados del análisis por nivel de dispersión en mediciones promedio de VOP . . . . .	96
4.4	Comparación de las mediciones realizadas con valores de referencia . . . . .	98
4.5	Cálculo de VOP en interfaz gráfica . . . . .	104
	<b>CONCLUSIONES</b> . . . . .	106
	<b>RECOMENDACIONES</b> . . . . .	107
	<b>REFERENCIAS</b> . . . . .	109

[www.bdigital.ula.ve](http://www.bdigital.ula.ve)

## ÍNDICE DE TABLAS

<b>Tabla</b>	<b>pp.</b>
3.1 Valores de coeficientes del filtro FIR pasa bajas de 50 Hz . . . . .	56
3.2 Valores de coeficientes del filtro FIR pasa bajas de 30 Hz . . . . .	62
4.1 Resultados de comparación entre valores manuales y los provistos por algoritmos . . . . .	92
4.2 Resultados de experimentos de sección 3.14.2 con paciente 1 . . . . .	94
4.3 Resultados de experimentos de sección 3.14.2 con paciente 2 . . . . .	94
4.4 Resultados de experimentos de sección 3.14.2 con paciente 3 . . . . .	96
4.5 Resultados de dispersión con todos los pacientes . . . . .	96
4.6 Tabla de resultados de desviación en VOP promedios por paciente . . . . .	97
4.7 VOP (m/s) promedios por paciente . . . . .	97
4.8 Tabla de valores de referencia detallados en [2] . . . . .	100

www.bdigital.ula.ve

## ÍNDICE DE FIGURAS

<b>Figura</b>	<b>pp.</b>
2.1 Estructura de los grandes vasos [3] . . . . .	8
2.2 Arterias coronarias [4] . . . . .	10
2.3 Forma típica de la onda de pulso [1] . . . . .	12
2.4 Medición de la velocidad de la onda de pulso . . . . .	13
2.5 Fotopletismografía por reflexión . . . . .	14
2.6 Fotopletismografía por transmisión . . . . .	14
2.7 Diagrama de la absorción de luz en los tejidos [5] . . . . .	15
2.8 Diferentes puntos de medición de pulso por fotopletismografía. Puntos azules corresponden a la medición por transmisión. Puntos rojos corresponden a los sitios para la medición por reflexión . . . . .	16
2.9 Forma típica de la señal electrocardiográfica y sus componentes de ondas e intervalos [6] . . . . .	17
3.1 Diagrama de bloques general del circuito . . . . .	19
3.2 Sensor de oximetría por transmisión . . . . .	20
3.3 Sensor de oximetría por reflexión . . . . .	20
3.4 Amplificador de transimpedancia . . . . .	21
3.5 Circuito de control de LEDs . . . . .	22
3.6 Etapa de filtrado y amplificación . . . . .	23
3.7 Arduino uno Rv. 3 . . . . .	26
3.8 Bus de comunicación $I^2C$ . . . . .	27
3.9 Diagrama del circuito amplificador de ECG . . . . .	29
3.10 Diagrama general del circuito MVOP . . . . .	30
3.11 Protocolo SPI . . . . .	32
3.12 Optoacoplador 6N136 . . . . .	32
3.13 Diagrama del circuito de aislamiento . . . . .	33
3.14 Capa superior del circuito impreso principal . . . . .	35
3.15 Capa inferior del circuito impreso principal . . . . .	35
3.16 Circuito impreso de la etapa de aislamiento . . . . .	36
3.17 Diagrama de flujo general en programación del arduino . . . . .	37
3.18 Diagrama de flujo de la subrutina de calibración . . . . .	38
3.19 Diagrama de flujo de la subrutina de muestreo . . . . .	41
3.20 Diagrama de flujo del método <i>handshake</i> en Python . . . . .	46

3.21	Diagrama de flujo del método calibración en Python . . . . .	47
3.22	Diagrama de flujo de la subrutina de muestreo en Python . . . . .	48
3.23	Tiempos de muestreo . . . . .	50
3.24	Estructura organizacional del almacenamiento de los datos . . . . .	52
3.25	Diagrama de bloques de algoritmo de detección de picos R en señal ECG .	54
3.26	Respuesta en frecuencia de filtro FIR 50 Hz (31 coeficientes) . . . . .	55
3.27	Filtrado de la señal ECG . . . . .	55
3.28	Reducción de línea de base ( <i>baseline</i> ) de la señal ECG . . . . .	56
3.29	Diferenciación de la señal ECG . . . . .	57
3.30	Señal post-umbralizado . . . . .	58
3.31	Flancos de subida y bajada provenientes de la señal umbralizada . . . . .	58
3.32	Delimitación de rangos de búsqueda de puntos máximos en señal ECG . .	60
3.33	Identificación de puntos máximos en la señal ECG . . . . .	61
3.34	Diagrama de bloques de algoritmo de detección de puntos máximos . . . . .	61
3.35	Respuesta en frecuencia de filtro FIR 30 Hz (51 coeficientes) . . . . .	63
3.36	Identificación de puntos máximos . . . . .	64
3.37	Identificación de puntos con algoritmo de la primera derivada o tangente. .	64
3.38	Derivación de la señal de pulso . . . . .	65
3.39	Identificación de puntos con tangente máxima . . . . .	66
3.40	Puntos identificados con tangente máxima en señales de pulso . . . . .	67
3.41	Identificación de puntos con algoritmo de segunda derivada . . . . .	68
3.42	Segunda derivada de señal de pulso . . . . .	68
3.43	Identificación de puntos mínimos . . . . .	69
3.44	Puntos identificados con segunda derivada máxima en señales de pulso . .	70
3.45	Ventana de base de datos . . . . .	70
3.46	Ventana principal . . . . .	72
3.47	Ventana principal, formulario . . . . .	73
3.48	Ventana principal, validación en formulario . . . . .	74
3.49	Ventana principal, panel de adquisición de datos . . . . .	75
3.50	Ventana principal, calibración . . . . .	76
3.51	Ventana principal, muestreo . . . . .	77
3.52	Ventana principal, almacenamiento de datos . . . . .	78
3.53	Ventana principal, registro de pacientes . . . . .	79
3.54	Ventana principal, registro de consultas . . . . .	80
3.55	Ventana principal, registro de reportes . . . . .	81
3.56	Ventana de reporte . . . . .	82
3.57	Documento PDF, primera página . . . . .	83
3.58	Documento PDF, segunda página . . . . .	84

4.1	Resultados manuales y provistos por algoritmos . . . . .	92
4.2	VOP por periodos de paciente 1 . . . . .	94
4.3	VOP por periodos de paciente 2 . . . . .	95
4.4	VOP por periodos de paciente 3 . . . . .	95
4.5	Resultado de VOP promedios en mediciones paciente 1 . . . . .	98
4.6	Resultado de VOP promedios en mediciones paciente 2 . . . . .	99
4.7	Resultado de VOP promedios en mediciones paciente 3 . . . . .	99
4.8	Resultados de VOP en relación a la edad de los pacientes . . . . .	101
4.9	Comparación de resultados con valores de referencia para las distintas poblaciones . . . . .	102
4.10	Resultados de VOP en relación al índice de masa corporal del paciente . . . . .	103
4.11	Velocidad de pulso en relación con la presión arterial media de los pacientes . . . . .	103
4.12	Cálculo de velocidad de pulso en interfaz gráfica . . . . .	105

[www.bdigital.ula.ve](http://www.bdigital.ula.ve)

# Capítulo 1

## PLANTEAMIENTO DEL PROBLEMA

Las enfermedades cardiovasculares de acuerdo a la Organización Mundial de la Salud (O.M.S) <sup>1</sup> producen anualmente el mayor número de muertes por causa natural [7]. Varias de estas enfermedades, como la cardiopatía isquémica y la enfermedad cardíaca hipertensiva, poseen una estrecha relación con la rigidez arterial [8].

La rigidez en las arterias es una característica que provee mucha información sobre la fisiopatología de los vasos. Es considerada un biomarcador en la evaluación del riesgo cardiovascular y es utilizada en la evaluación de la eficacia terapéutica de determinados tratamientos [8]. La base de dichas afirmaciones está sustentada en diferentes estudios que detallan una relación entre el aumento de la rigidez arterial (diferente al producido por el envejecimiento de las arterias) con casos clínicos o patologías como la insuficiencia renal crónica terminal, la hipertensión arterial, la enfermedad cerebrovascular, la arterioesclerosis y la cardiopatía isquémica [9, 10].

En cuanto a su medición, el grado de rigidez de las arterias puede determinarse mediante procedimientos invasivos, sin embargo, debido a los riesgos inherentes, se han desarrollado métodos no cruentos que estiman cuanta rigidez presentan los vasos bajo estudio. La rigidez arterial puede entonces ser evaluada por: la medida de la presión de pulso (PP); mediante el análisis de los cambios en el diámetro arterial; por medio del análisis de la morfología de la onda de pulso o por la medida de la velocidad de la onda de pulso (VOP). El último método es por lo general, el más aceptado desde el punto de vista del valor pronóstico [9].

La medición de la rigidez arterial por medio de la VOP es, no obstante y desde el enfoque clínico, una variable en fase de investigación. Sin embargo, debido a la relevancia que representa la rigidez arterial sobre el riesgo cardiovascular, se está dando paso a la

---

<sup>1</sup>De acuerdo a datos del año 2012.



investigación clínica en esta área de estudio [11].

En el aspecto tecnológico, la medición de la VOP es realizada mayoritariamente con equipos comerciales (aunque no altamente difundidos). La existencia de diferentes fabricantes y métodos para su medición conllevan a una problemática: la dificultad para la estandarización debido a las diferencias entre las técnicas de adquisición que producen también diferencias entre mediciones. Adicionalmente, como las tecnologías comerciales no están altamente difundidas en el sector, no se han establecido normas y estudios estadísticos generalizados para establecer con rigurosidad el factor de riesgo en relación a la VOP.

Otro punto destacable en cuanto a las tecnologías existentes, es que algunas de ellas presentan algunos inconvenientes en determinadas situaciones, tal es el caso de la dificultad de obtener mediciones apropiadas ante la presencia de arritmias u obesidad extrema del paciente.

Otro aspecto en cuanto a la problemática del tópico, es el procedimiento realizado para la determinación del riesgo cardiovascular. Usualmente se determina mediante el sondeo de hábitos (consumo del alcohol, tabaco, ejercicio o sedentarismo), antecedentes clínicos o existencia de enfermedades agravantes (hipertensión arterial, diabetes, obesidad). De los cuales, buena parte de los mencionados no son representados de forma claramente cuantificable. La inclusión de la medición de la VOP agregaría un componente útil y más cuantitativo para el análisis y estimación del riesgo cardiovascular.

De manera similar, con el estudio de la VOP ante la presencia de enfermedades relacionadas con la rigidez arterial, pudieran realizarse mejores diagnósticos que establezcan de forma más clara la influencia de la enfermedad en la fisiopatología de los vasos.

En términos de funcionamiento, generalmente los equipos para la medición de la VOP obtienen dos señales de onda de pulso, o una señal y las ondas electrocardiográficas (ECG) del paciente, y se calcula el retardo entre las dos señales adquiridas. El retardo entre las ondas es usado en conjunto con la distancia entre los dos puntos de adquisición para calcular la velocidad. La obtención del tiempo de tránsito (retardo) se calcula identificando puntos de medición <sup>2</sup> en cada señal para luego determinar la diferencia de tiempo entre

---

<sup>2</sup>Entiéndase como puntos de medición aquellos utilizados como referencia para las mediciones, un punto de medición puede ser entonces el pico de una señal.

ellos.

Habitualmente los puntos de medición son los puntos mínimos de la onda. Sin embargo, se han realizado estudios donde se utilizan aquellos donde la derivada de la onda de pulso es mayor [12]. Y por otra parte, se ha detallado el uso de la transformada Wavelet para determinar los puntos donde ejercen más influencia los componentes de más alta frecuencia de la señal [13]. En consecuencia, es necesario un análisis para identificar la técnica de procesamiento más robusta y que habilite la medición de la VOP ante la presencia de arritmias.

Finalmente, con el desarrollo de un equipo de medición de VOP se buscaría reducir los costos de producción y adquisición en relación a las marcas existentes con el fin de favorecer la presencia de equipos para la medición de VOP en el ámbito clínico y de investigación en dicha área de estudio.

## 1.1 Antecedentes

El desarrollo de dispositivos para la medición de la VOP involucra necesariamente filtros para obtener la señal de pulso. Lopez [14] por ejemplo, implementó una etapa de filtrado compuesta por cinco filtros, cuatro de ellos pasivos y uno activo para la amplificación de la señal de onda de pulso. El ancho de banda de toda la etapa de filtrado fue diseñado para el rango de 0.8-4.8 Hz.

En el uso de sensores por fotoplethismografía predominó la utilización de un diodo emisor de luz NIR 940 nm y un fotodiodo. Para la adquisición de la señal con el fotodiodo se utiliza generalmente el integrado OPA380 como amplificador de transimpedancia diseñado para aplicaciones con fotodiodos.

El procesamiento de la señal se realiza usualmente en un computador personal y los datos se transmiten vía puerto serial. Autores como Lopez [14] utilizaron placas de desarrollo como la TWR-K53N512 de la familia K53 de los MCUs Kinetics para la adquisición, procesamiento local y envío al ordenador.

Con respecto al procesamiento de las señales adquiridas en el computador, Gonzalez y Col. [12], y por otra parte, Kazanavicius y Col. [15], realizaron pruebas para la obtención de la velocidad de onda de pulso mediante la identificación de los puntos de medición. Los

métodos propuestos involucraron la detección de puntos máximos, mínimos y de tangente máxima.

De acuerdo con Kazanavicius y Col. [15], el método para la detección de puntos mínimo es el más aceptado y también el más óptimo. Consecuentemente en dicha investigación, se desarrollaron metodologías para identificar del pie de la señal en las señales de pulso.

Por su último, otros autores como Jeong y col. [13] realizaron la identificación de los puntos por medio de la transformada Wavelet, obteniendo resultados satisfactorios.

En relación al entorno de trabajo en el computador, autores como Gozalez y Col. [12] realizaron la adquisición y el procesamiento matemático en el ordenador con programación orientada a objetos en C++. Sin embargo, la herramienta de software mayormente utilizada por los investigadores fue Matlab, utilizada por múltiples autores [16–18].

De acuerdo a los trabajos precedentes, existen diferentes métodos para obtener la medición de la VOP. Algunos autores como Jeong y Col. [13] plantean en su diseño la medición de la VOP con la adquisición de la señal electrocardiográfica (ECG) y una señal por fotopleletismografía. Bajo esta técnica generalmente se realiza la asunción de que el pico R de la onda ECG representa instantáneamente el inicio de la contracción ventricular, esto genera imprecisiones porque en realidad existe un retardo para que la acción mecánica se inicie (período de pre-ejección). En consecuencia la mayoría de los autores plantean la medición de la VOP mediante la adquisición de dos señales de pulso y en caso de utilizar la señal ECG, lo hacen como señal de referencia.

## 1.2 Objetivos

### 1.2.1 Objetivo General

- Diseñar un equipo de medición de la rigidez arterial basado en la adquisición de la velocidad de onda de pulso por fotopleletismografía.

### 1.2.2 Objetivos Específicos

- Estudiar las tecnologías existentes para la adquisición de la señal de onda de pulso por fotopleletismografía.

- Seleccionar el tipo de sensor por fotopleletismografía que facilite la adquisición de la onda de pulso en diferentes partes del cuerpo.
- Diseñar el circuito de adquisición, amplificación y filtrado de dos señales de onda de pulso de forma simultánea.
- Desarrollar un sistema para la digitalización simultánea de las dos señales de onda de pulso y la señal ECG.
- Estudiar los diferentes algoritmos para la identificación de puntos utilizados para el cálculo de tiempo de tránsito entre dos señales de onda de pulso.
- Implementar el algoritmo más estable y robusto para la identificación de puntos para el cálculo del tiempo de tránsito entre dos señales de onda de pulso y una ECG.

### 1.3 Justificación e Importancia

El uso de sensores de fotopleletismografía para el diseño e implementación de un equipo de medición de VOP económicamente asequible habría de reducir los costos de fabricación y adquisición de dichos equipos, lo cual fomentaría la investigación en cuanto a la relación del riesgo cardiovascular y la rigidez arterial, en el ámbito clínico.

Por otra parte, la medición de la VOP es un método *Gold Standard* no invasivo para la rigidez arterial, pero su uso clínico ha sido limitado debido a la carencia de normativas y referencias para instrumentos de medición específicos. En Europa por ejemplo, se ha tratado de estandarizar la evaluación del riesgo cardiovascular por medio de la medición de la VOP, con modelos de regresión y con el uso de un sistema *SphygmoCor* (equipo comercial) [19]. De manera que con la existencia de equipos para la medición de VOP se pudieran hacer estudios similares pero enfocados a la población latinoamericana.

### 1.4 Estructura

El presente trabajo consta de cuatro (4) partes principales. En el capítulo 2 se detalla una recopilación de bases teóricas que involucran al sistema arterial, tales como: la

anatomía de las grandes arterias, las coronarias y su irrigación al corazón, la fisiología del movimiento de las arterias elásticas y la velocidad de la onda de pulso y su relación con la rigidez arterial. Aunado a ello, se detalla la técnica de fotoplethismografía y la onda electrocardiográfica.

Posteriormente en el capítulo 3 se presenta un trabajo técnico para el diseño e implementación de la etapa de hardware que incluye: selección del sensor por fotoplethismografía, diseño del circuito de adquisición y filtrado de la señales analógicas de las ondas de pulso, y finalmente el diseño y programación del circuito para la digitalización, muestreo y transferencia de los datos vía puerto USB hacia un computador personal. Por otra parte, también está incluido un desarrollo teórico a nivel de software, abarcándose entre ello, la escogencia del mejor algoritmo para la identificación de puntos de medición en las señales en base a la robustez y estabilidad que presenten.

A continuación en el Capítulo 4 se detalla un análisis exhaustivo de los resultados obtenidos.

Por último, se presentan las conclusiones sobre los resultados y algunas recomendaciones para mejoras e investigaciones posteriores.

# Capítulo 2

## MARCO TEÓRICO

### 2.1 Aparato circulatorio

El aparato circulatorio está compuesto por la sangre, el corazón y los vasos sanguíneos. En el aspecto fisiológico, es el encargado de suministrar nutrientes, oxígeno y hormonas a los tejidos y recoger los productos de desecho para ser eliminados.

Dentro del aparato circulatorio los vasos sanguíneos son en conciso conductos que distribuyen la sangre bombeada desde el ventrículo izquierdo del corazón a los órganos y tejidos del cuerpo humano, para luego regresarla al corazón nuevamente [20].

#### 2.1.1 Anatomía y fisiología de las grandes arterias

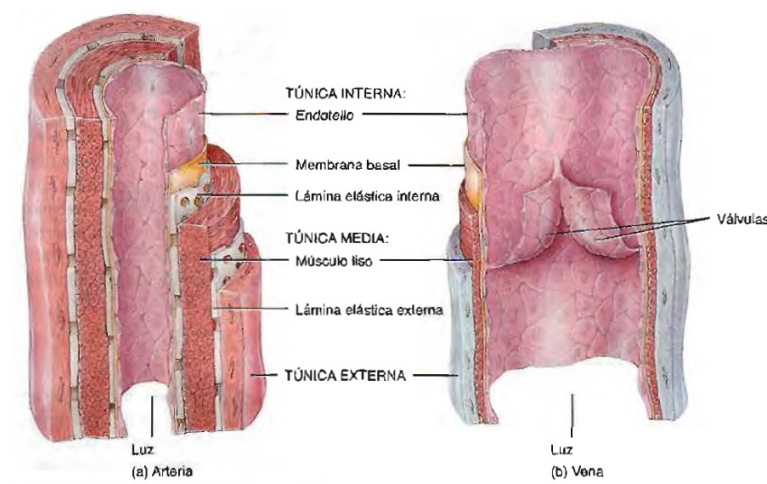
Dentro de la clasificación de los vasos se distinguen cinco tipos: las arterias, las arteriolas, los capilares, las vénulas y las venas. Particularmente, las arterias son las que conducen la sangre desde el corazón hacia otros órganos.

Entre las arterias se pueden enumerar varios tipos clasificadas de acuerdo a su calibre y tamaño: las grandes arterias elásticas, las arterias musculares de mediano calibre, las de pequeño calibre y finalmente las arteriolas. Partiendo del corazón hasta los diferentes tejidos, las arterias se van dividiendo sucesivamente hasta ser capilares <sup>1</sup> [3].

La pared de una arteria posee tres capas o tunicas: una túnica interna, una media y una externa. La capa más interna o íntima comprende un revestimiento de epitelio pavimentoso simple llamado endotelio, una membrana basal y una capa de tejido elástico llamada lámina elástica interna. La capa media está constituida por fibras elásticas y musculares lisas que se extienden en forma circular alrededor de la luz. La túnica media también posee una lámina elástica externa compuesta por tejido elástico. Finalmente, la

---

<sup>1</sup>Vasos diminutos donde se produce el intercambio de sustancias entre la sangre y los tejidos corporales.



**Figura 2.1. Estructura de los grandes vasos [3]**

capa más externa está compuesta en su mayor parte por fibras elásticas colágenas <sup>2</sup> [3]. Ver figura 2.1 para más detalles.

En el aspecto fisiológico, las arterias cumplen dos funciones: distribuir la sangre proveniente del corazón hacia todas las partes del cuerpo para irrigarlo, y convertir el flujo intermitente del corazón en un flujo aproximadamente continuo en los capilares. Como podrá pensarse, la primera función está relacionada de forma directa con la anatomía de las estructuras arteriales, y la segunda con las propiedades mecánicas de las arterias [1].

### 2.1.2 Propiedades mecánicas de las arterias elásticas

- **Elasticidad:** La elasticidad de un material es la capacidad para sufrir deformaciones reversibles cuando se encuentra sometido a fuerzas exteriores y recuperar su forma original cuando dichas fuerzas se retiran. La elasticidad se cuantifica con el módulo de Young. El módulo de Young es la relación entre el esfuerzo  $\sigma$  y el coeficiente de deformación  $\epsilon$ , ver ecuación 2.1. En los materiales inertes,  $E$  es constante dentro de ciertos límites. En el caso de las arterias, en la parte inicial de la curva el coeficiente de deformación versus esfuerzo es aplanada y se debe a las fibras de elastina que se distienden con facilidad, pero a medida que se incrementa la deformación, comienzan a estirarse las fibras colágenas (mil veces más rígidas) y en consecuencia, el coeficiente de esfuerzo crece rápidamente con pequeños aumentos de deformación [21, 22].

<sup>2</sup>Debido a la presencia de contenido elástico las arterias normalmente poseen alta distensibilidad, por lo que sus paredes se estiran con facilidad o se expanden sin romperse ante cambios de presión.

$$E = \frac{\sigma}{\varepsilon} \quad (2.1)$$

- Compliance (C): es una propiedad estructural de un material que relaciona el cambio de volumen con un cambio en la presión transmural, ver ecuación 2.2. La distensibilidad arterial disminuye con la edad debido a un aumento de la rigidez de la pared arterial causado por la sustitución de las fibras de elastina por fibras de colágeno [21, 22].

$$C = \frac{\Delta V}{\Delta P} \quad (2.2)$$

### 2.1.3 Arterias coronarias y su función

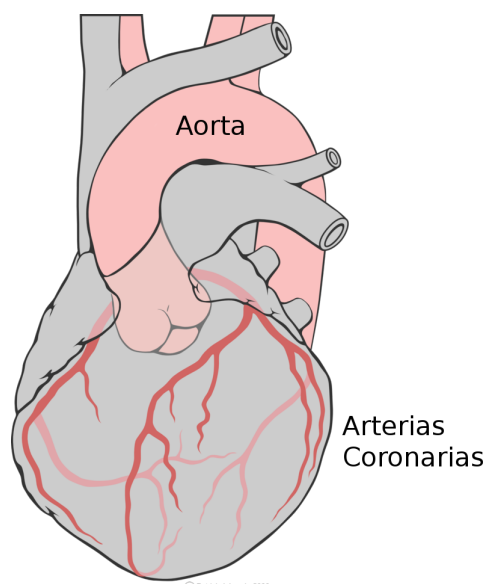
Como en el corazón los nutrientes no pueden difundirse lo suficientemente rápido desde la sangre contenida en las cámaras cardíacas a todas las etapas de la pared cardíaca, el miocardio posee su propia vascularización [4]: la circulación coronaria o cardíaca. Las arterias coronarias nacen de la aorta ascendente y rodean al corazón para irrigarlo.

Cuando el corazón se contrae, fluye poca sangre por dichas arterias porque son comprimidas hasta cerrarse. No obstante, cuando el corazón está en la diástole, la elevada presión en la aorta permite la circulación de la sangre a través de ellas, y luego hacia los capilares y venas coronarias [3]. Ver figura 2.2.

El funcionamiento de las arterias coronarias está influenciado por las presiones presentes en el arco aórtico en las diferentes etapas del ciclo cardíaco y éstas a su vez, dependen de las propiedades mecánicas de los grandes vasos. Particularmente, la rigidez arterial (de la cual la VOP es dependiente) es una propiedad mecánica que tiene una estrecha relación con la presión variable en los grandes vasos.

Puede entonces inferirse que la VOP ejerce influencia sobre la efectividad de la irrigación coronaria. Esto se fundamenta en que, cuando el corazón expulsa la sangre a la aorta, se genera una onda que viaja por los grandes vasos hasta llegar a bifurcaciones del árbol arterial. El choque de la onda con dichas bifurcaciones produce otra en dirección opuesta a la original. Consecuentemente, dependiendo de la rigidez que presentan los vasos, la onda reflejada llegará durante o después del período diastólico del corazón.





**Figura 2.2. Arterias coronarias [4]**

Naturalmente, una VOP baja (condición óptima) produce que la onda reflejada llegue al arco aórtico justo en la diástole, aspecto que favorece a la irrigación del corazón porque aumenta la presión diastólica. Por el contrario, cuando la VOP es elevada a causa de un aumento en la rigidez arterial por envejecimiento de los vasos o debido a determinadas patologías, la onda de pulso puede desplazarse de una manera lo suficientemente veloz para que la llegada al arco aórtico de la onda reflejada coincida con la sístole ventricular, lo que dificulta la expulsión de sangre por parte del corazón y disminuya la irrigación miocárdica.

#### **2.1.4 Fisiología del movimiento de las arterias elásticas (Onda de Pulso)**

A lo largo del ciclo cardíaco se producen cambios de volumen en las arterias, que se originan a consecuencia de las contracciones del corazón, que producen la expulsión de sangre de forma intermitente.

Particularmente, durante la sístole ventricular, la sangre expulsada produce un incremento de presión en el arco aórtico, el cual debido a las propiedades mecánicas de los vasos, es transmitido en forma de onda a lo largo del árbol arterial.

En base a lo anterior, la onda de pulso es el registro de presión o volumen en un punto arterial específico a lo largo del tiempo. El pulso puede palparse en cualquier arteria que se encuentre próxima a la superficie del cuerpo y que pueda ser comprimida contra un hueso

u otra estructura firme [3]. Es más fuerte en las arterias cercanas al corazón, se vuelve más débil en las arteriolas y prácticamente desaparece en los capilares.

La morfología de la señal del pulso consta básicamente de una onda sistólica y una diastólica separadas por una incisura aguda, ver figura 2.3. Durante la sístole ventricular la sangre es expulsada hacia la aorta produciendo un incremento de presión que genera la onda sistólica. Cuando la sístole finaliza se cierra la válvula aórtica y se produce una bifurcación en la señal de pulso denominada incisura aguda, que establece el límite entre la sístole y la diástole [1].

Posterior al cierre de la válvula aórtica se produce un pequeño incremento de presión debido a la fuerza que ejercen los vasos para reducir su distensión.

Adicionalmente, la forma de onda puede ser distorsionada por la presencia de una onda de reflexión sistólica [23] que se origina cuando la onda de pulso llega a bifurcaciones importantes creando un fenómeno de rebote.

En condiciones normales la onda de reflexión llega al arco aórtico durante la diástole, favoreciendo la irrigación coronaria. Sin embargo, cuando la VOP es mayor, la onda de reflexión sistólica viaja tan rápido que llega a la válvula aórtica durante la sístole, la presión sistólica por ende aumenta y la diastólica disminuye. Consecuentemente, se incrementa el riesgo cardiovascular porque el corazón necesita realizar mayor esfuerzo para expulsar la sangre y recibe menos irrigación por las arterias coronarias en su período diastólico [23,24].

## 2.2 Velocidad de la onda de Pulso

La velocidad de la onda de pulso (VOP) es la rapidez con que viaja la onda de presión transmitida desde la aorta a través del árbol arterial [24].

Por medio del estudio de la onda de pulso puede determinarse el estado del sistema arterial. De hecho, sólo con la morfología de la onda puede denotarse un envejecimiento de las arterias, o la influencia de patologías como arterioesclerosis, estenosis aórtica, insuficiencia aórtica, entre otras patologías [1].

Las alteraciones en la rigidez arterial son consecuencia de cambios físicos producidos por alguna condición patológica o fisiológica. De manera similar, la VOP conserva una estrecha relación con la rigidez arterial. Un endurecimiento arterial por ejemplo, produce

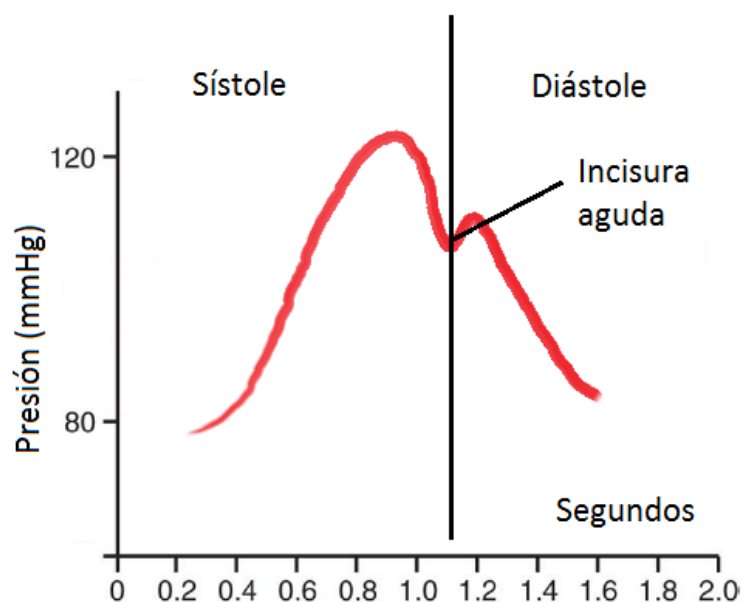


Figura 2.3. Forma típica de la onda de pulso [1]

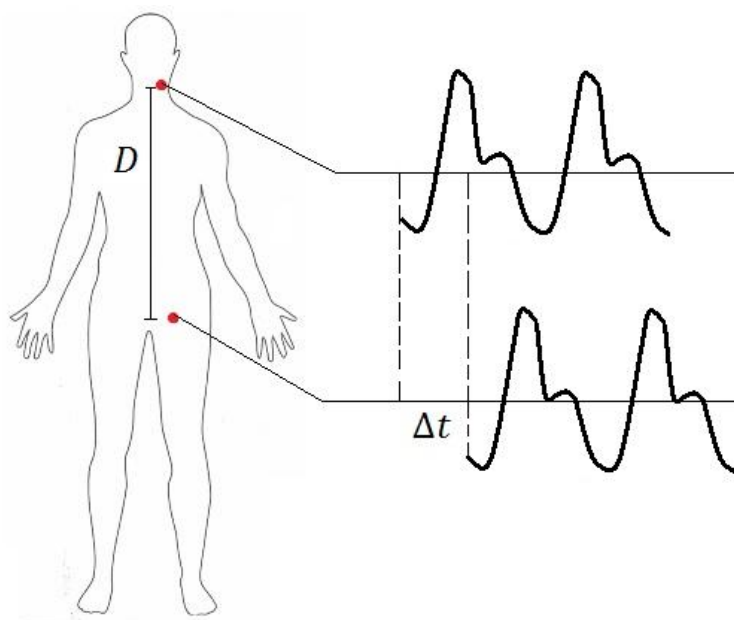
un incremento en la velocidad del pulso. Por el contrario, un aumento hipotético en la elasticidad de las arterias disminuiría la VOP. Dicha relación está fundamentada por la ecuación de Moens-Korteweg:

$$VOP = \sqrt{\frac{hE}{2r\rho}} \quad (2.3)$$

Donde  $h$  es el grosor de toda la pared arterial,  $E$  el módulo de Young,  $r$  es el radio interno y  $\rho$  es la densidad de la sangre.

La VOP es una variable aceptada para evaluar la rigidez arterial porque además de su relación matemática, su medición es relativamente sencilla. Se obtiene por medio de la medición simultánea de dos señales de pulso en lugares distintos e identificables del árbol arterial, la distancia entre los puntos de medición ( $d$ ) y el retardo entre las señales ( $t$ ). Obteniéndose la velocidad ( $v$ ) mediante la ecuación  $d = (v)(t)$ , ver figura 2.4 como referencia.

Otra opción viable para la medición de la VOP es por medio de solo una medición de una onda de pulso y la señal ECG. Se identifican por ejemplo, los puntos máximos de la señal ECG y los puntos correspondientes en la señal de pulso. Se determina la diferencia de tiempo entre ellos y la distancia para obtener la velocidad [12]. La limitante de este



**Figura 2.4. Medición de la velocidad de la onda de pulso**

método es el error presente bajo la consideración que la señal ECG (una señal eléctrica) es simultánea con el fenómeno físico de flujo sanguíneo en el inicio de la aorta, cuando en realidad existe un tiempo de retardo entre ellos.

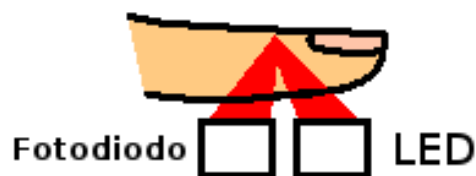
Por otra parte, es importante destacar que la adquisición de la señal ECG es algunas veces utilizada para aumentar el grado de robustez en la medición de la VOP. Es especialmente útil ante la presencia de arritmias, condición en que las mediciones de la VOP de forma estándar pueden generar resultados erróneos.

En cuanto a las tecnologías para la medición de la VOP, varían principalmente de acuerdo a los tipos de sensores utilizados. La medición de la onda de pulso de forma no invasiva puede realizarse por medio de sensores con efecto piezo eléctrico, por tonometría por aplanación, por ultrasonido, por pletismografía o por fotopletismografía.

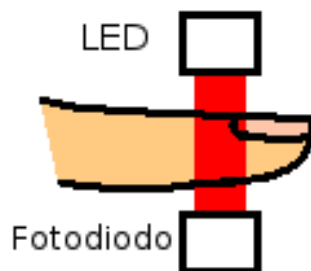
### 2.3 Fotopletismografía

La pletismografía es la medición volumétrica de alguna parte del cuerpo. Esta medición puede realizarse con distintos métodos tales como: por impedancia, capacitancia, por strain gauges o también por medio de sensores fotoeléctricos.

La fotopletismografía es una medición de volumen que se realiza de forma óptica par-



**Figura 2.5. Fotopleletismografía por reflexión**



**Figura 2.6. Fotopleletismografía por transmisión**

tiendo del hecho que cuando la luz incide sobre la piel, parte de ella es reflejada, otra parte es absorbida y la restante es transmitida. Si se mide bien sea la luz reflejada o la transmitida durante un período de tiempo determinado se obtendría una señal de fotopleletismografía.

El proceso para la medición por fotopleletismografía puede observarse de forma más ilustrativa en las figuras 2.5 y 2.6. Un diodo LED es utilizado para dirigir un haz de luz hacia los tejidos y la luz que no es absorbida es registrada por un foto sensor.

La luz emitida tiene longitudes de onda específicas en manera de optimizar el nivel de absorción para el flujo sanguíneo. Para aplicaciones de fotopleletismografía como pulsioxímetros y medidores de onda de pulso sanguíneo es común el uso de LED rojos e infrarrojos<sup>3</sup> con longitudes de onda de 660 nm y 940 nm.

Un punto destacable de la medición del pulso por fotopleletismografía es que la señal pulsátil obtenida tiene un nivel DC correspondiente a las absorciones fijas de tejidos inertes. Por otra parte, la variabilidad de la señal es provocada por los cambios de volumen en los vasos sanguíneos, ver figura 2.7.

Para la medición del pulso cardíaco por fotopleletismografía resulta relevante el tipo de técnica a realizar y el sitio de adquisición de la señal. La medición por transmisión

<sup>3</sup>Estos últimos en su mayoría son LEDs NIR (siglas en inglés "near infrared" o cercano al infrarrojo).

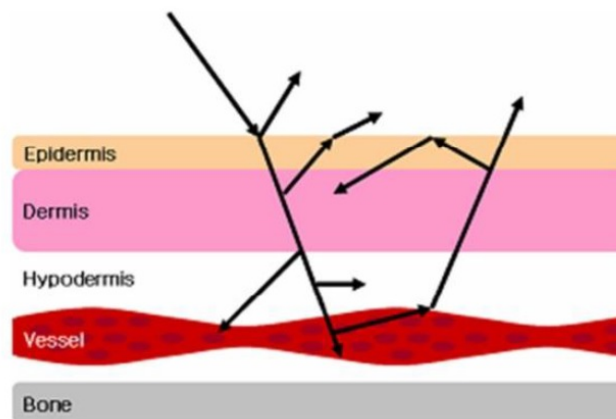


Figura 2.7. Diagrama de la absorción de luz en los tejidos [5]

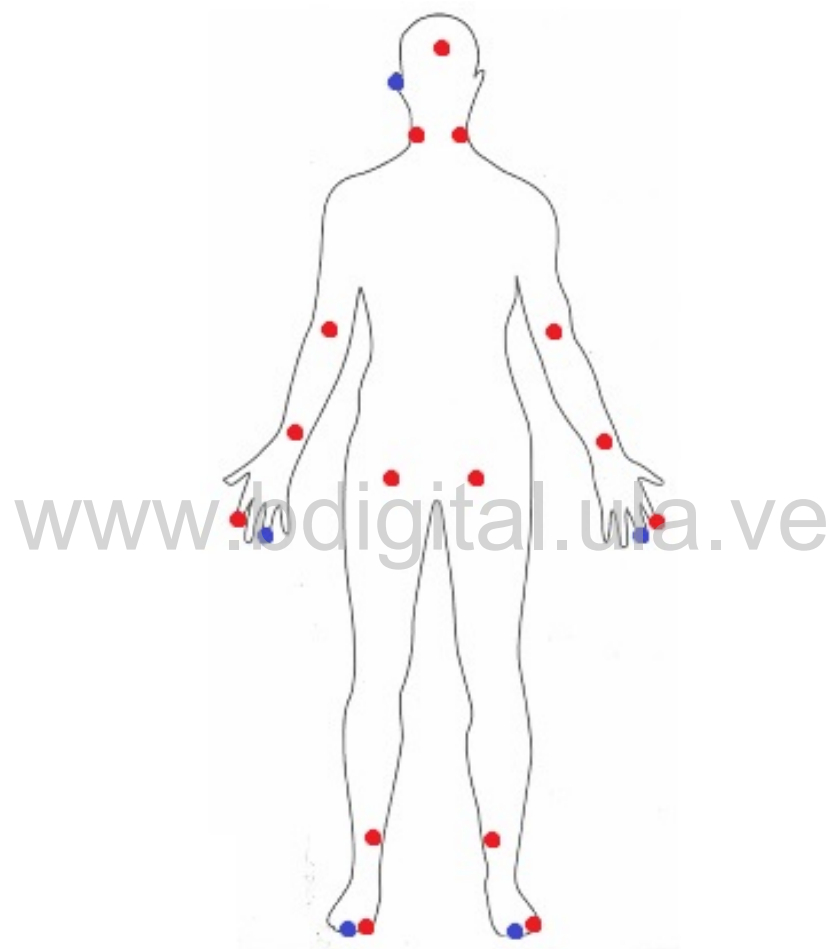
es posible por ejemplo, solo en lugares lo suficientemente delgados para que la luz no sea totalmente absorbida y pueda registrarse la información en el sensor (figura 2.6). La medición por reflexión es entonces más versátil, pues habilita la adquisición de la señal del pulso en casi cualquier lugar donde se pueda palpar el pulso con el tacto, pero es más sensible a los movimientos del paciente. En la figura 2.8 se observan los lugares de medición posibles del pulso por fotopleletismografía.

## 2.4 El electrocardiograma

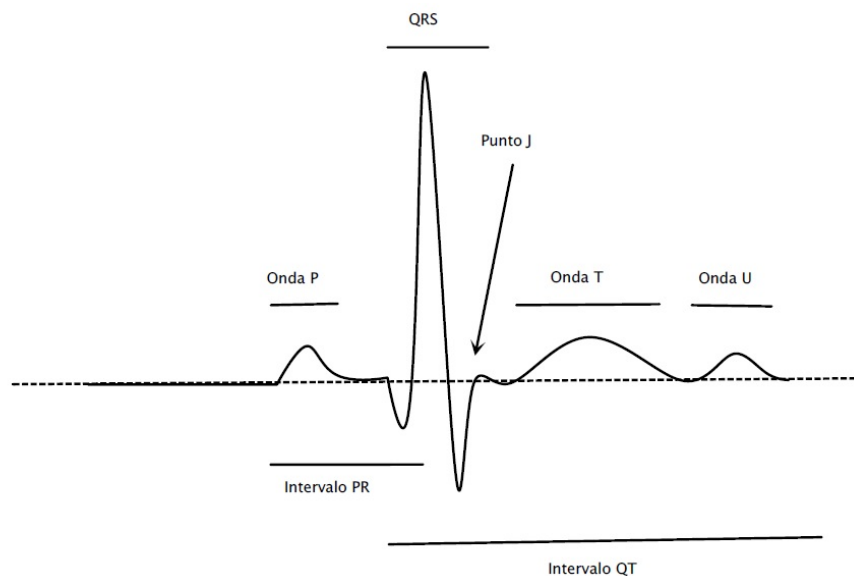
La señal electrocardiográfica corresponde al registro gráfico de la actividad eléctrica del corazón. Fue observado por primera vez por Einthoven en 1901 [6]. Es una señal que se obtiene por medio de electrodos colocados en forma superficial sobre ciertos puntos del tórax y/o las extremidades. La diferencia de potencial entre los electrodos da origen al electrocardiograma [20].

El electrocardiograma resulta una medición de alta importancia en la cardiología porque provee mucha información acerca del funcionamiento o estado del corazón. En la figura 2.9 se observa una onda ECG típica, la cual denota una onda P, el complejo QRS, la onda T y la onda U.

La onda P corresponde a los potenciales generados por la despolarización auricular. El complejo QRS trae consigo toda la información de la despolarización ventricular, y además coincide con la repolarización auricular, pero como los potenciales de las aurículas



**Figura 2.8.** Diferentes puntos de medición de pulso por fotopletismografía. Puntos azules corresponden a la medición por transmisión. Puntos rojos corresponden a los sitios para la medición por reflexión



**Figura 2.9.** Forma típica de la señal electrocardiográfica y sus componentes de ondas e intervalos [6]

son mucho menores que los de los ventrículos, esta información se pierde. La onda T por su parte corresponde al proceso de repolarización ventricular. Finalmente la onda U, que no aparece en la gran mayoría de los ECGs, representa fenómenos de repolarización ventricular tardía [6].



# Capítulo 3

## METODOLOGÍA

### 3.1 Diagrama de bloques general

El esquema de funcionamiento se basa en la adquisición de dos señales de pulso y la recepción de la señal ECG.

Las señales de pulso se obtienen mediante el uso de sensores por fotopleletismografía que reciben los cambios luminosos y los transforman en una señal o corriente eléctrica. Las señales de corriente son convertidas en señales de tensión por amplificadores de transimpedancia que luego son procesadas por filtros analógicos. Posteriormente, dichas señales son digitalizadas por el convertidor AD (Análogo Digital) del microcontrolador (Arduino Uno rev 3), para ser enviadas a un computador personal donde se realiza el procesamiento y cálculo de la VOP.

Los sensores son comandados por circuitos manejadores de LEDs y éstos a su vez por un convertidor digital analógico (DAC). El DAC controla también el nivel cero de las señales en los amplificadores y filtros activos que se alimentan con  $0 - 5V$ .

En términos generales, el diseño puede observarse en el diagrama de bloques de la figura 3.1.

Dentro del circuito digital el microcontrolador ejerce múltiples funciones: digitaliza las señales de pulso, comanda al convertidor digital analógico, recibe la señal electrocardiográfica del circuito externo, recibe parámetros del computador personal y envía todos los datos adquiridos de las señales al puerto serial.

Finalmente, dentro de la circuitería está incluido el uso de un amplificador ECG, y el diseño e implementación de un circuito que digitaliza la señal electrocardiográfica y provee un aislamiento eléctrico entre el resto del circuito y el paciente.

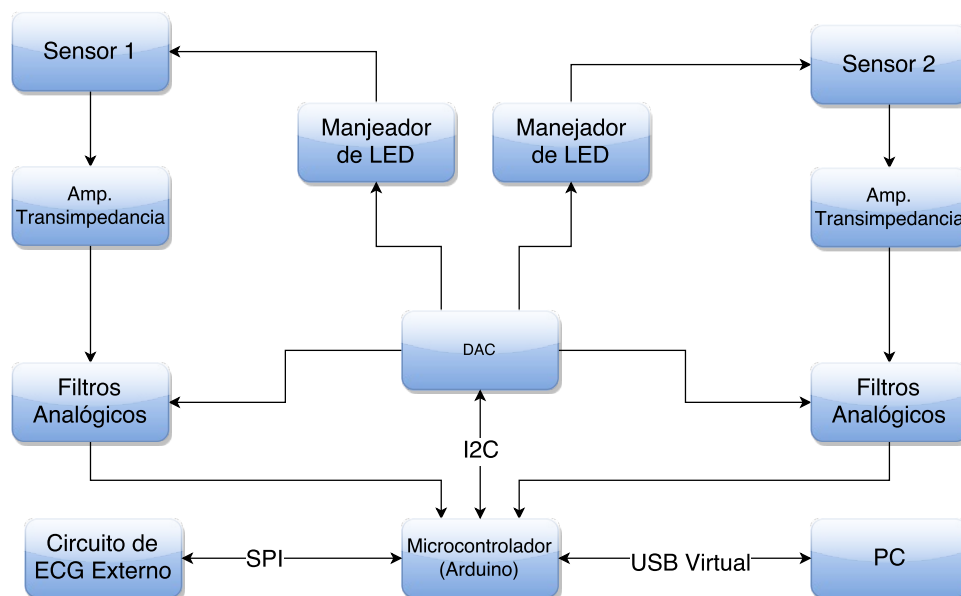


Figura 3.1. Diagrama de bloques general del circuito

### 3.2 Selección de sensores por fotopleletismografía

Como se ha detallado en la sección 2.3, la adquisición de las señales por fotopleletismografía puede realizarse por la técnica de transmisión (se emite un haz de luz a través de los tejidos y la luz que logra atravesar completamente el tejido es medida con un foto sensor) o por la técnica de reflexión (se la mide la luz reflejada por los tejidos y no la transmitida).

Entre las diferencias se encuentra el hecho que la medición por transmisión sólo puede ser realizada en pocas zonas, pero en general la señal obtenida es de mayor amplitud que la proveniente por reflexión o reflectancia.

Se han seleccionado para la aplicación sensores comerciales compatibles con la tecnología Nellcor (ver figuras 3.2 y 3.3), marca de fabricantes de oxímetros altamente difundida. Los sensores tienen la ventaja de tener incluido un conector macho DB-9 en uno de sus extremos y una distribución y conexión interna conocida. Esto último resulta favorable suponiendo una posible fabricación artesanal de dichos sensores para su utilización en el dispositivo de VOP, cuyo requisito sería la correcta conexión y disposición de los componentes para que sean compatibles con un sensor Nellcor.



**Figura 3.2. Sensor de oximetría por transmisión**

[www.bdigital.ula.ve](http://www.bdigital.ula.ve)



**Figura 3.3. Sensor de oximetría por reflexión**

C.C.Reconocimiento

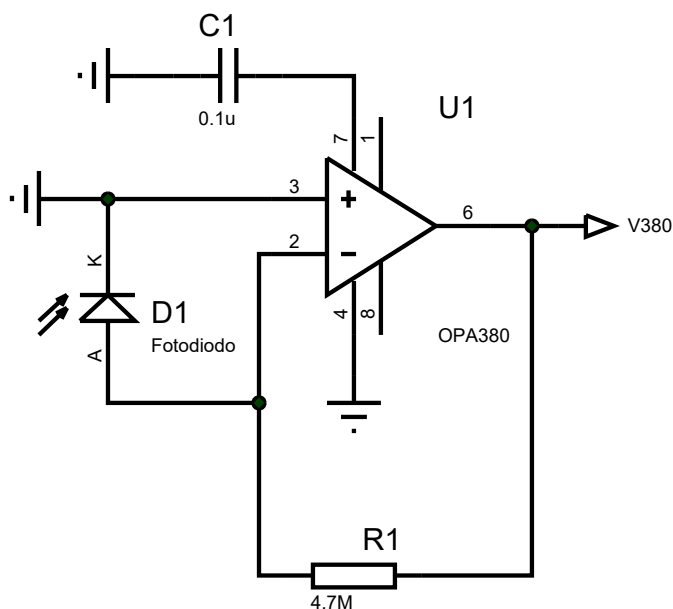


Figura 3.4. Amplificador de transimpedancia

### 3.3 Amplificador de transimpedancia

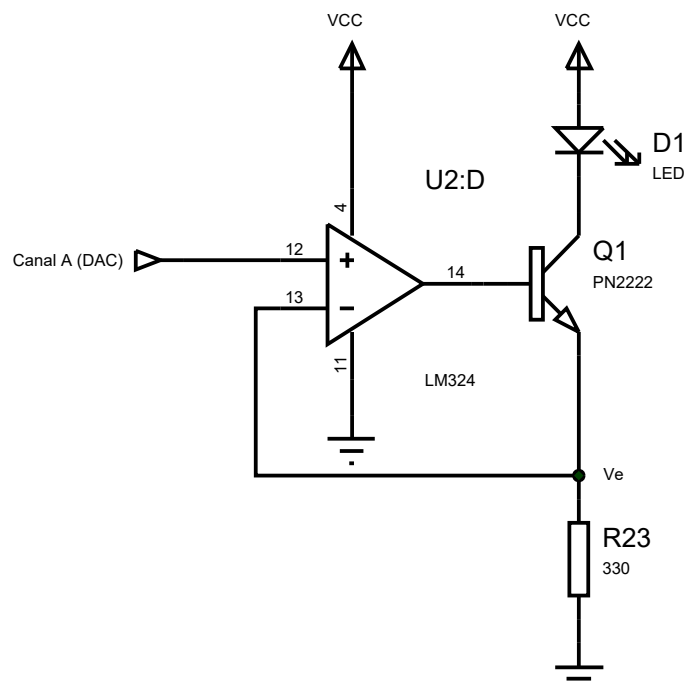
Como los sensores por fotopleletismografía poseen fotodiodos que perciben los cambios de luz y generan una corriente eléctrica. Esta corriente debe ser recibida por un amplificador de transimpedancia para su amplificación y transformación a una señal de tensión.

Se seleccionó el OPA380, diseñado para aplicaciones con fotodiodos. Este amplificador es de alta precisión, provee baja corriente de polarización (menor a 50pA) y requiere alimentación en el rango de tensión de 0-5 V.

En la figura 3.4 se observa la conexión para la conversión de corriente-voltaje o amplificación de transimpedancia. La corriente  $I$  es generada por el fotodiodo y fluye por la resistencia  $R_1$ . El voltaje  $V$  es entonces obtenido por ley de Ohm:  $V = R_1 I$ . El resultado es una señal de tensión originada por una señal de corriente amplificada  $R_1$ .

### 3.4 Circuito de control de LEDs

Debido a que los sensores por fotopleletismografía disponen de LEDs para emitir los haces de luz, se hace necesario un circuito de control para establecer un grado de potencia de los LEDs óptimo para la medición de los cambios de luz incidente o reflejada a través



**Figura 3.5. Circuito de control de LEDs**

de los tejidos. Cabe destacar que los tejidos así como sus volúmenes y consistencias varían de acuerdo al lugar de registro del pulso y también entre pacientes. Esto hace imperante la implementación de un circuito que gobierne la luminosidad de los LEDs en los sensores. Como regla general en dicho contexto, se observa que a mayor nivel de luminosidad por parte de los LEDs mayor es la amplitud de la señal obtenida.

En la figura 3.5 se observa el circuito de control de LEDs implementado. La tensión en la entrada positiva del amplificador está dada por el Canal A del DAC. Este voltaje es asignado en el emisor del transistor por medio de la entrada negativa del amplificador operacional, entonces  $V_e = V_{dacA}$ .

Como existe una estrecha relación entre la corriente del emisor y la del colector en los transistores ( $I_c \simeq I_e$ ), la intensidad de corriente y en consecuencia la luminosidad del LED es controlada por los cambios de voltaje provenientes del DAC.

Podrá observarse también que la corriente en el emisor y por tanto en el colector, está limitada a  $I_e = 5V/330\Omega = 15.1515 \text{ mA}$  para evitar exceder los límites de corriente permitidos por el LED.

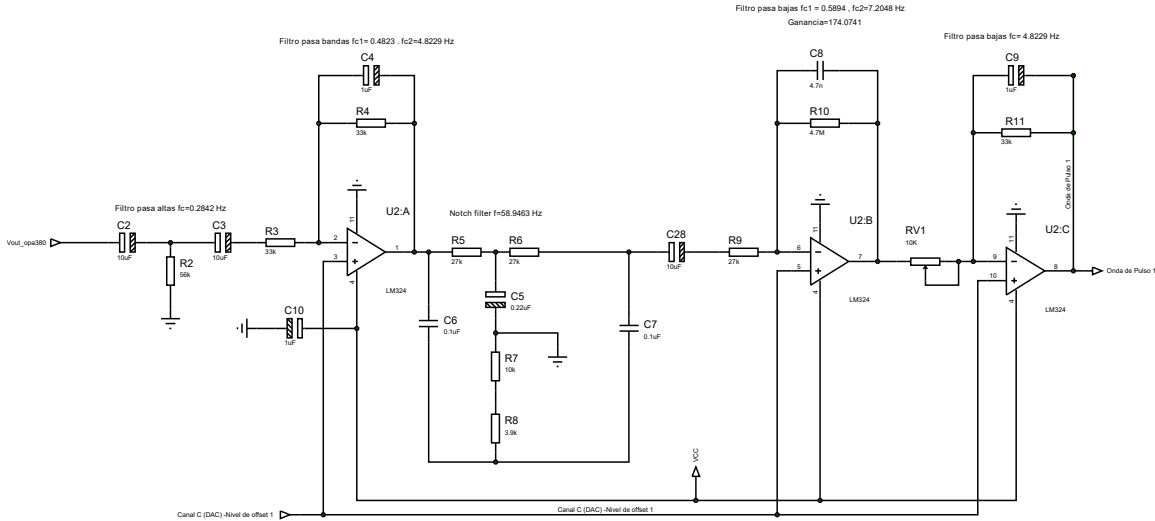


Figura 3.6. Etapa de filtrado y amplificación

### 3.5 Filtros analógicos y amplificadores

Posterior a la conversión corriente-tensión del OPA380, se implementaron filtros analógicos y de amplificación para el tratamiento de las señales de pulso.

La etapa de filtrado completa tiene una banda de paso aproximada de 0.28 – 4.28 Hz. Cabe destacar que el rango de frecuencia de la señal de pulso se ubica entre los 0.04 – 5 Hz [25–27], con una frecuencia fundamental de aproximadamente 1 Hz [25].

En cuanto al diseño circuital el circuito analógico consta de varias etapas de filtros pasivos y activos, ver figura 3.6. La primera etapa corresponde a un filtro pasa altas pasivo de  $f_c = 0.2842$  Hz con el fin de atenuar la componente DC de la señal. La segunda consiste en un filtro activo pasa banda con amplificación  $G_1 = \frac{R_4}{R_3} = 1$  y frecuencias de corte:  $f_{c1} = \frac{1}{2\pi C_3 R_3} = 0.4823$  Hz,  $f_{c2} = \frac{1}{2\pi C_4 R_4} = 4.8229$  Hz.

A continuación se dispuso un filtro Notch de frecuencia de corte  $f_{c3} = 56.2031$  Hz y  $f_{c4} = 58.0919$  Hz, regido por las ecuaciones 3.1 y 3.3 con el fin de atenuar la componente de 60 Hz de la red eléctrica.

$$f_{c3} = \frac{1}{2\pi} \sqrt{\frac{\frac{1}{C_6} + \frac{1}{C_7}}{C_5 R_5 R_6}} = \frac{1}{2\pi} \sqrt{\frac{\frac{1}{0.1\mu F} + \frac{1}{0.1\mu F}}{(0.22\mu F)(27k\Omega)(27k\Omega)}} = 56.2031 \text{ Hz} \quad (3.1)$$

$$f_{c4} = \frac{1}{2\pi} \sqrt{\frac{1}{C_6 C_7 (R_7 + R_8)(R_5 + R_6)}} \quad (3.2)$$

$$f_{c_4} = \frac{1}{2\pi} \sqrt{\frac{1}{(0.1\mu F)(0.1\mu F)((10k\Omega) + (3.9k\Omega))((27k\Omega) + (27k\Omega))}} = 58.0919 \text{ Hz} \quad (3.3)$$

Posterior al filtro Notch se implementaron dos etapas de amplificación y filtrado adicionales, ambas de primer orden. La primera es un filtro pasa banda con amplificación  $G_2 = \frac{R_{10}}{R_9} = 174.074$  y frecuencias de corte  $f_{c_5} = \frac{1}{2\pi C_{28} R_9} = 0.5894 \text{ Hz}$  y  $f_{c_6} = \frac{1}{2\pi C_8 R_{10}} = 7.2048 \text{ Hz}$ . Finalmente, la segunda corresponde a un filtro pasa bajas activo de  $f_{c_7} = \frac{1}{2\pi C_8 R_{10}} = 7.2048 \text{ Hz}$  con ganancia mínima de  $G_3 = \frac{R_{11}}{RV_1} = \frac{33k\Omega}{10k\Omega} = 3.3$ , aunque variable gracias al trimmer  $RV_1$ .

Finalmente para los filtros activos ha de detallarse la utilización de amplificadores operacionales LM324 en base a los siguientes puntos:

- Cada integrado posee cuatro amplificadores operacionales (menor número de componentes).
- Pueden trabajar en escala de tensión (0 – 5V).
- Bajo costo y alta disponibilidad comercial.

### 3.6 Microcontrolador

Para la selección del microcontrolador es necesario detallar los requerimientos mínimos de acuerdo a la aplicación. Siguiendo el orden de ideas expuesto, es imprescindible que el microcontrolador disponga de:

- Puertos de comunicación serial síncrona  $I^2C$  y SPI.
- Un receptor transmisor universal síncrono asíncrono  $USART$ .
- Convertidor analógico digital (ADC) de 8 bits de dos canales.<sup>1</sup>
- Frecuencia de muestreo para las señales de pulso mayor a 20 Hz<sup>2</sup>.

<sup>1</sup>Este requerimiento se basa en experiencias de investigaciones antecedentes [13, 28].

<sup>2</sup>Como el rango de frecuencias del pulso cardíaco oscila entre 0.04-5 Hz [25–27] y es necesario el muestreo de dos señales de pulso simultáneamente, entonces la frecuencia mínima de muestreo es  $f_s = 2((2)f_{max}) = (2)(2)(5) = 20 \text{ Hz}$ .

Los requerimientos mencionados son necesarios para que el microcontrolador pueda:

- Recibir instrucciones de un computador personal.
- Comandar a un convertidor digital analógico vía  $I^2C$  para configurar varias partes del circuito.
- Realizar el muestreo simultáneo de dos señales de pulso.
- Recibir una señal electrocardiográfica por un convertidor digital analógico proveniente de un circuito externo vía SPI.
- Enviar todos los datos adquiridos a un computador personal.

En base a las exigencias puntualizadas se ha seleccionado para la presente investigación el uso de un dispositivo Arduino Uno rev 3.

Arduino es una plataforma *open source* basada en una placa con un microcontrolador y un entorno de desarrollo, diseñada para facilitar el uso de la electrónica en proyectos multidisciplinarios.

El modelo de placa Arduino Uno rev 3 (ver figura 3.7) posee un microcontrolador Atmel AVR (Atmega328), con 14 pines de entrada/salida digital, 32 KB memoria flash, un conector USB y un resonador cerámico de 16 MHz. El microcontrolador posee 6 canales de conversión analógica digital con una resolución de 10 bits y con una tasa máxima de lectura de aproximadamente 10000 veces por segundo ( $f_s = 10$  KHz). Además está habilitado para comunicaciones seriales con los protocolos SPI,  $I^2C$  y un puerto UART - TTL 5V que permite la comunicación vía puerto USB con el computador personal.

### 3.7 Convertidor Digital Analógico

Puede constatar en las figuras 3.5 y 3.6 que el convertidor digital analógico (DAC) cumple dos funciones principales. La primera, relacionada con la figura 3.6 está enfocada en la asignación del cero de las señales tratadas, esto debido a la escogencia de los amplificadores LM324 que trabajan con alimentación en el rango de  $0 - 5V$ . De esta manera el



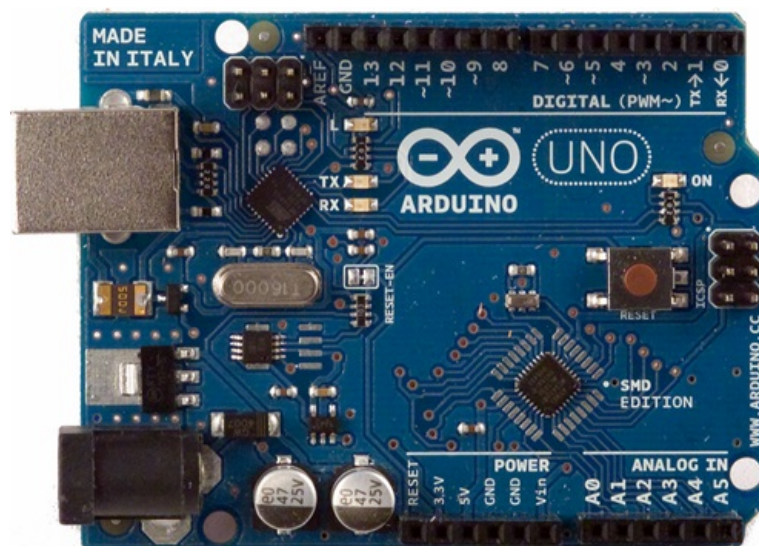


Figura 3.7. Arduino uno Rv. 3

convertidor digital analógico (DAC) asigna el nivel de referencia deseado para las señales en la etapa de filtrado.

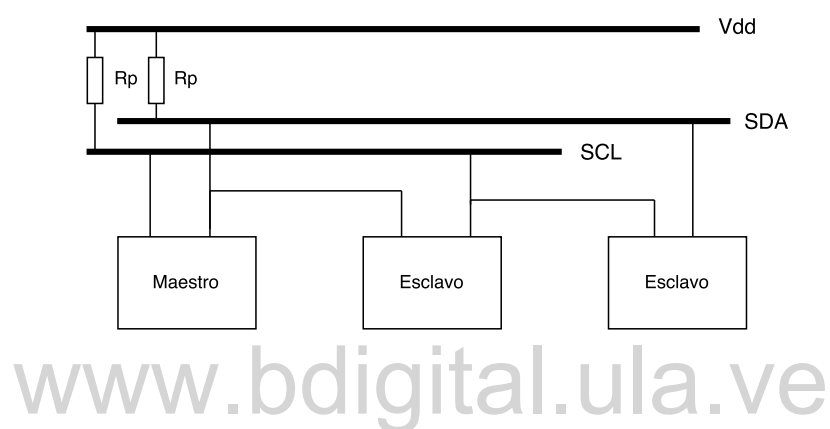
La segunda función está relacionada con el control de la luminosidad de los LEDs en los sensores fotoplestismográficos. En la figura 3.5 y sección 3.4 se ha detallado que la tensión en la entrada positiva del amplificador operacional influye en la intensidad del LED del sensor.

Por otra parte, se seleccionó el convertidor digital analógico (DAC) MCP4728 porque posee 4 canales internos de 12 bits en su entrada y cuatro salidas independientes, puede trabajar con alimentación asimétrica (0 – 5V) y soporta comunicación serial por el protocolo  $I^2C$ .

La implementación de la comunicación se realizó atendiendo a lo siguiente:

- El protocolo  $I^2C$  (*Inter-Integrated Circuit*) es un bus de comunicación serial que consta de dos líneas de bidireccionales: una línea de datos (SDA o *Serial Data Line*) y una línea de reloj (*Serial Clock Line* (SCL)). El bus requiere también una conexión a masa en caso que los circuitos bajo comunicación no estén alimentados bajo la misma fuente. Las líneas SDA y SCL necesitan resistencias de pull-up, ver figura 3.8.
- En en el bus  $I^2C$  los dispositivos conectados pueden ser maestros o esclavos y cada uno posee una dirección única. La comunicación (en su forma más simple) comienza

cuando el dispositivo maestro envía un bit de inicio (*Start*). Esto prepara a los esclavos para recibir una dirección y un bit de escritura o lectura, que serán enviados por el maestro en la línea SDA. La dirección debe corresponder a la dirección de uno de los esclavos conectados al bus. Posterior a ello, el esclavo correspondiente emite un bit de reconocimiento (ACK) y envía o recibe la información esperada. Finalmente, el esclavo emite otro bit de reconocimiento (ACK) y el maestro transmite un bit de parada para habilitar al bus de datos nuevamente.



**Figura 3.8. Bus de comunicación I<sup>2</sup>C**

Consecuentemente, en base al procedimiento recién mencionado, el Arduino es capaz de comunicar al DAC los niveles de tensión requeridos en las diferentes etapas de filtrado y sensores del circuito.

### 3.8 Circuito Amplificador ECG

Aunque el circuito amplificador de la señal electrocardiográfica no fue diseñado en la presente investigación, resulta de importancia ilustrar su diseño con la finalidad de clarificar el funcionamiento de éste en el resto del circuito.

El diseño del circuito involucra el uso de amplificadores diferenciales INA326 de Texas Instruments y amplificadores operacionales OPA2335A.

En la figura 3.9 se observa el circuito en cuestión. En el conector J1 se observan las entradas provenientes de los electrodos, las resistencias  $R_1$ ,  $R_2$  y  $R_3$  disminuyen el riesgo

de riesgo del paciente al limitar la corriente en sus respectivas líneas. Adicionalmente los amplificadores operacionales  $U1 : A$  y  $U1 : B$  proveen un incremento adicional de impedancia de entrada a las líneas de dos de tres electrodos conectados al paciente.

Por otra parte, el amplificador INA326 rechaza la señal en modo común y amplifica la diferencial. Esta última es filtrada y amplificada por un circuito pasa bajas de frecuencia 150 Hz y ganancia 200.

Finalmente se debe detallar la implementación de un circuito de retroalimentación activa constituido por los amplificadores operacionales  $U2 : A$  y  $U2 : B$ . En esta etapa el promedio de la señal (nivel DC) en modo común es invertida y amplificada para su posterior introducción al paciente con el fin de disminuir las interferencias de modo común.

Por último, cabe destacar que el circuito amplificador de ECG dispone de un regulador de tensión 78L05 para la estabilización del nivel de tensión de 5V. El circuito fue diseñado para ser alimentado por una batería de 9V.

### 3.9 Circuito General

Atendiendo a todas las consideraciones mencionadas desde la sección 3.1 hasta la 3.7 se presenta en la figura 3.10 el diagrama general del circuito, exceptuando la etapa de aislamiento y la conexión con el amplificador ECG.

### 3.10 Circuito de aislamiento

El circuito amplificador de ECG externo detallado en la sección 3.8 provee en su salida una señal analógica que debe ser digitalizada por un ADC.

El microcontrolador de la placa Arduino Uno lleva incluido un convertidor ADC de 6 canales, pero no puede ser empleado para dicha finalidad porque ello implica una conexión directa entre el circuito principal y el paciente. Ha de detallarse que la alimentación del circuito general y la del microcontrolador provienen de un puerto USB de un computador personal y como el circuito externo de ECG necesita la disposición de electrodos en la piel del paciente, la conexión mencionada incrementaría significativamente el riesgo de

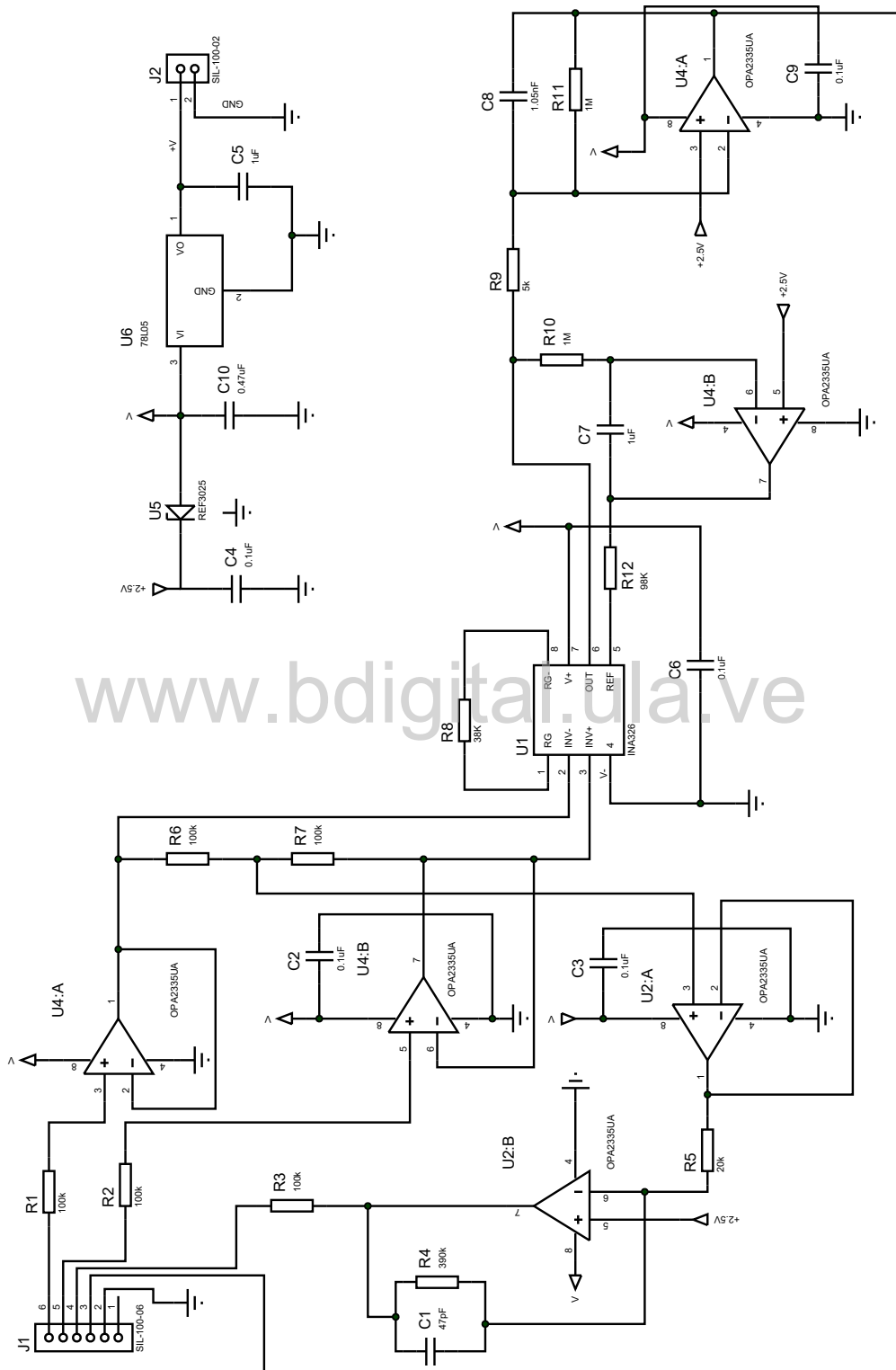


Figura 3.9. Diagrama del circuito amplificador de ECG

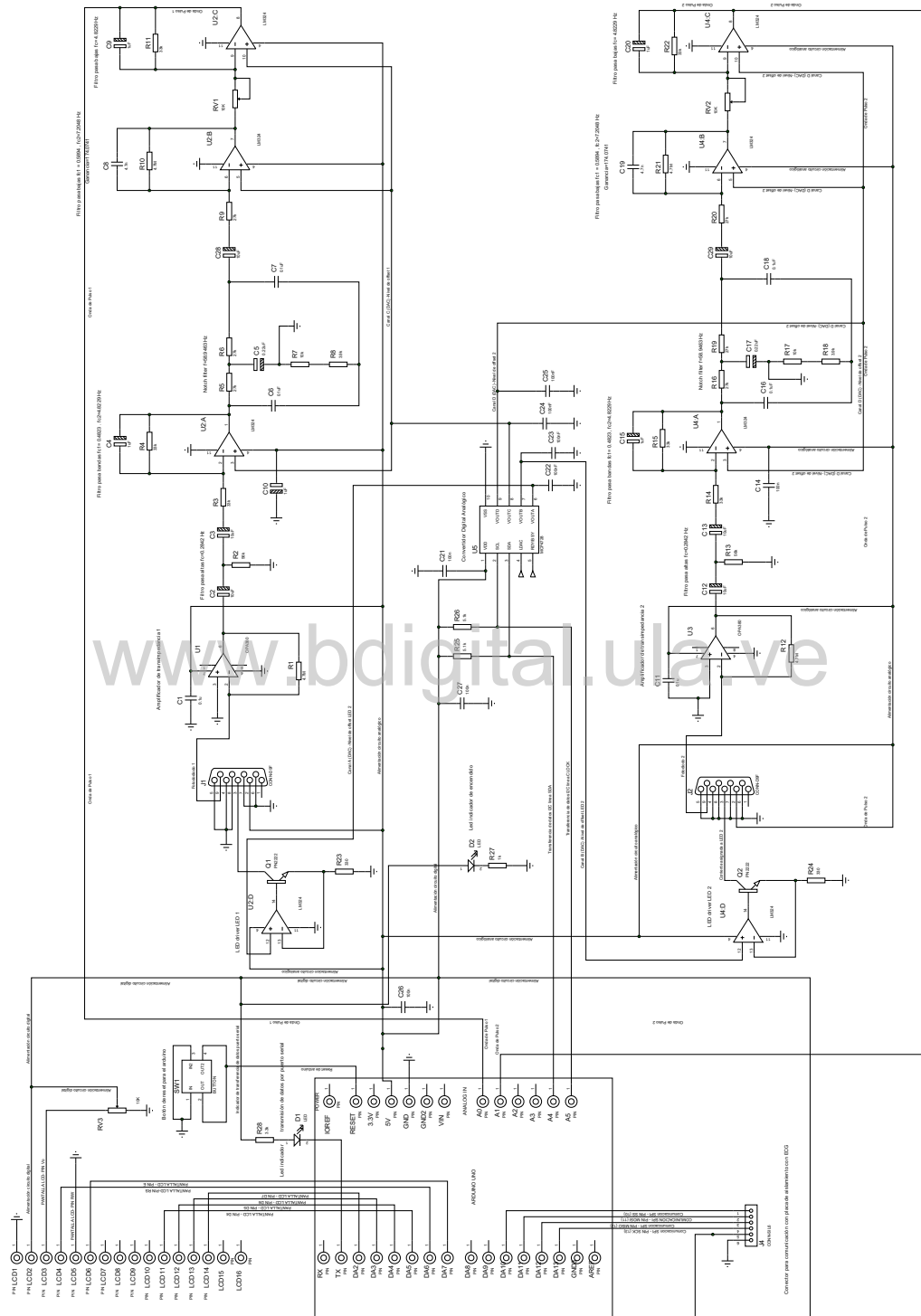


Figura 3.10. Diagrama general del circuito MVOP

electrocución.

La solución a dicho problema fue resuelto con la implementación de un circuito de aislamiento eléctrico, el cual separa el circuito de amplificación de ECG (alimentado por baterías para seguridad del paciente) con respecto al microcontrolador y otras etapas del sistema alimentadas por la fuente de poder del computador personal.

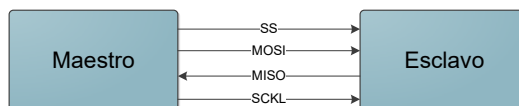
En el circuito de aislamiento se realiza la conversión digital de la señal ECG con un MCP3002, que es un convertidor analógico digital de 10 bits que funciona con el protocolo de comunicación SPI.

La implementación SPI se realizó atendiendo a lo siguiente:

- La interfaz de comunicación SPI (Serial Protocol Interface) es un protocolo de comunicación serial entre dos o varios dispositivos. Se exige que uno de los dispositivos funja como maestro y el restante (o los restantes) como esclavo. La comunicación se realiza con cuatro vías, ver figura 3.11.
- La primera vía (denominada  $\bar{SS}$ ) es la de habilitación de comunicación. Cuando  $\bar{SS}$  está en un nivel lógico bajo, el esclavo está habilitado para recibir o transmitir información al maestro. Esta vía es especialmente útil cuando existe más de un esclavo, pues sólo se activaría el esclavo que ve un cero lógico en su entrada  $\bar{SS}$  y el resto de ellos permanecerían inactivos.
- En la siguiente vía la información viaja desde el maestro al esclavo (MOSI o *Master Output Slave Input*) y de forma contraria, del esclavo al maestro en la línea MISO (*Master Input Slave Output*).
- Se requiere una señal de reloj SCLK proveniente del maestro hacia el esclavo para la sincronización de los dispositivos.

Como se ha podido constatar en los puntos recién mencionados, el protocolo de comunicación SPI consta de vías unidireccionales, lo cual resulta totalmente favorable para la implementación de un circuito de aislamiento.

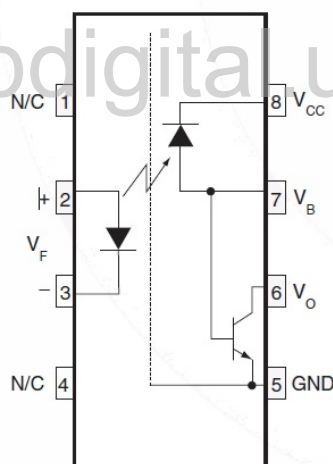
En las líneas de comunicación entre el microcontrolador y el MCP3002 se colocaron optoacopladores para efectuar un aislamiento eléctrico entre los dos dispositivos. Se seleccionaron optoacopladores 6N136 para uso en aplicaciones digitales (compatibles con



**Figura 3.11. Protocolo SPI**

tecnología TTL) de alta velocidad (1 Mbit/s). La velocidad de respuesta por parte de los optoacopladores es de alta importancia porque estos marcan el límite en la velocidad de transferencia de datos en el puerto SPI.

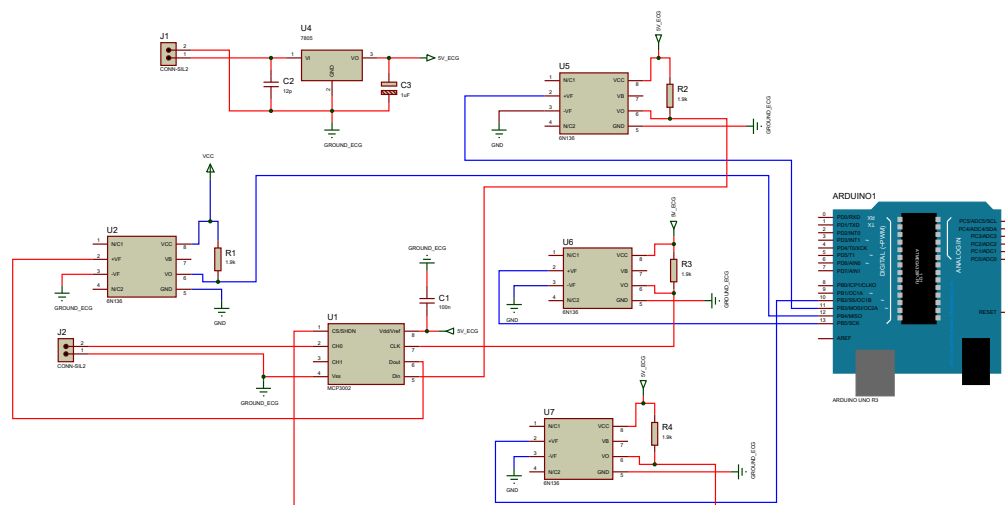
Los optoacopladores comúnmente están constituidos por un LED y un fototransistor. La señal emitida por el LED es recibida por el fototransistor en forma de luz y no de forma eléctrica, por lo que se asegura el aislamiento eléctrico entre ambas partes (ver figura 3.12). Cabe destacar que el aislamiento efectivo entre un dispositivo y otro se obtiene sólo si las conexiones de ambos provienen de alimentaciones distintas y estos no están conectados entre sí.



**Figura 3.12. Optoacoplador 6N136**

Una rápida observación a la configuración interna del optoacoplador hace evidente que la transmisión sólo puede ser posible de forma unidireccional (esta característica es la que realmente restringe el uso de otros protocolos de comunicación para la presente aplicación). La disposición de los optoacopladores, en consecuencia, debe realizarse atendiendo a la dirección de transferencia de los datos entre los dos dispositivos en comunicación.

Por otra parte, una limitante en el uso de los optoacopladores digitales es que estos generan un retardo en la señal, es decir, la respuesta del optoacoplador no es inmediata



**Figura 3.13. Diagrama del circuito de aislamiento**

porque el foto transistor interno necesita un tiempo para su excitación ante cambios en la señal transmitida. Esto es un punto de especial importancia si se requiere aislamiento en aplicaciones donde se necesite una muy rápida transferencia de datos.

Por último, es necesario recalcar que es posible el aislamiento eléctrico por medio de optoacopladores analógicos. Ello simplificaría el diseño e implementación del circuito significativamente. Sin embargo, un aspecto altamente desfavorable es que los optoacopladores analógicos de bajo costo tienden a presentar comportamientos aléales que pueden distorsionar gravemente las señales a transmitir. De igual manera, aunque existan optoacopladores analógicos de alta calidad, el costo de dichos componentes y su nula disponibilidad en el mercado nacional hacen descartable su empleo en la aplicación.

En la figura 3.13 se puede observar el esquemático del circuito de aislamiento. Las líneas en rojo están referenciadas por la alimentación de baterías del ECG. Las azules por su parte, por la alimentación del Arduino.

### 3.11 Diseño del PCB

El diseño de las placas del circuito general y de aislamiento se realizó de forma independiente. Las conexiones entre ellas se efectúan por medio de cables y conectores anexos



al diseño.

El diseño de todos los circuitos impresos se realizó con ISIS-Proteus (software especializado para el diseño de PCB y simulaciones de circuitos electrónicos) y la fabricación de las placas por el método de fresado.

Para la minimización de interferencias se hizo especial énfasis en los siguientes criterios:

- ✓ Se buscó en lo posible la reducción del área de bucles formados por el recorrido de las pistas en el circuito impreso.
- ✓ Se buscó la generación de planos de tierra en lugares no ocupados por los componentes. La conexión de los planos de referencia y la tierra de los circuitos se realizó desde un solo punto.
- ✓ Se definieron áreas separadas dentro del PCB para la circuitería analógica y la digital. Aún cuando la alimentación de las áreas sea la misma, se procuró en lo posible la separación de una rama de alimentación para el circuito analógico y una para la digital. Se implementaron condensadores en la bifurcación de la alimentación de dichas etapas para establecer una separación entre ellas.

En las figuras 3.14 y 3.15 se puede observar el circuito impreso del circuito general (exceptuando la etapa de aislamiento) en su capa superior e inferior.

En la figura 3.16 se ilustra el circuito impreso desarrollado para la etapa de aislamiento.

### 3.12 Programación del Arduino

La plataforma de Arduino dispone de un lenguaje de programación denominado Arduino (basado en *Wiring*) y un entorno de desarrollo Arduino (basado en *Processing*).

La estructura básica del lenguaje de programación Arduino está constituida por dos funciones indispensables: *setup* y *loop*. En la primera se realiza la configuración del arduino: se configuran los puertos como entradas o salidas, se declaran variables, se inicializan protocolos de comunicación, entre otros. Esta función se ejecuta una sola vez y se invoca cuando el programa empieza. Por otra parte, la segunda corresponde a un bucle donde se encuentra el código que se ejecuta continuamente, es básicamente el núcleo de todos los programas de Arduino.

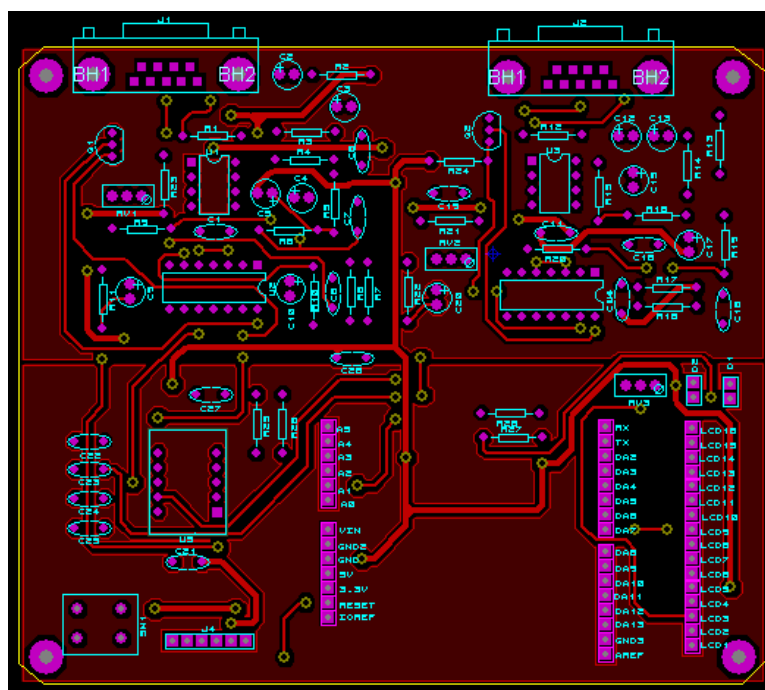


Figura 3.14. Capa superior del circuito impreso principal

[www.bdigital.ula.ve](http://www.bdigital.ula.ve)

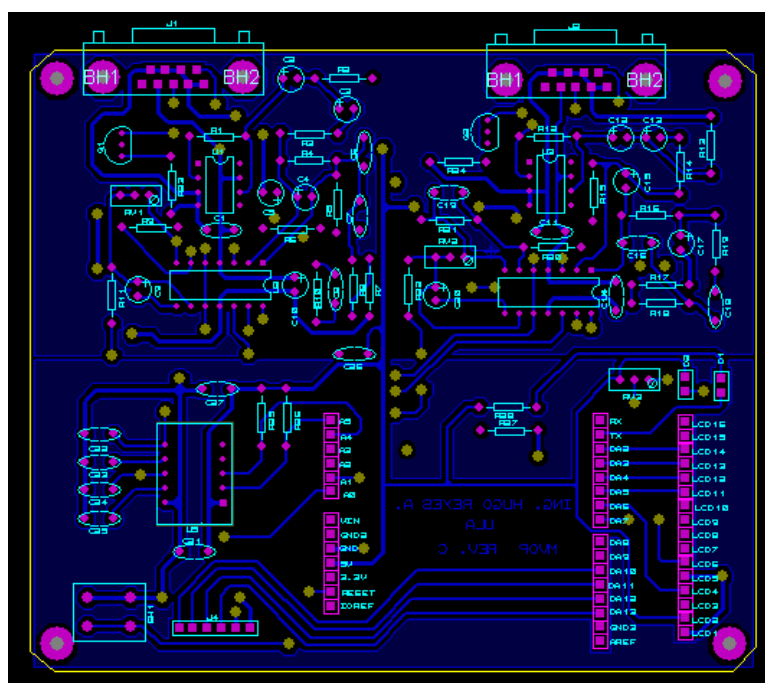


Figura 3.15. Capa inferior del circuito impreso principal

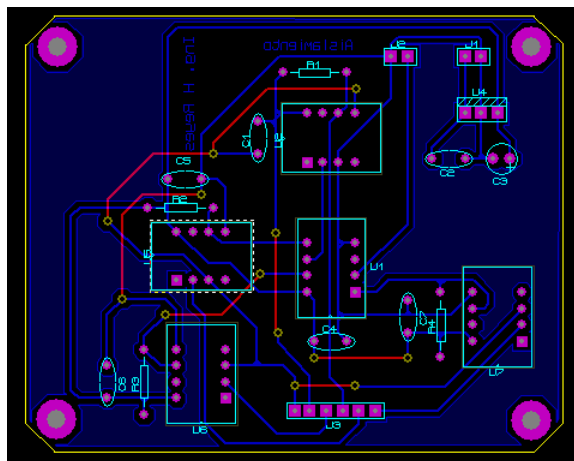


Figura 3.16. Circuito impreso de la etapa de aislamiento

En base a las consideraciones anteriores se procedió a la programación del Arduino cuyo diagrama de flujo puede observarse en la figura 3.17. Puede inferirse mediante la observación del diagrama que los primeros bloques representan a la función *setup* de la estructura básica del Arduino. En ella se declaran variables, se inicializan puertos y se envía un mensaje al computador personal para indicar que el dispositivo Arduino está preparado para recibir instrucciones.

Posterior a las instrucciones mencionadas se ejecuta la función *loop*. En ella el arduino primeramente ejecuta instrucciones de espera hasta que exista un dato disponible en el puerto serial para realizar una lectura. La lectura debe corresponder a un número entero que represente a uno de los dos posibles “comandos” presentes en el diagrama de flujo (ver figura 3.17). En base al número recibido, se ejecuta la subrutina de calibración o la de muestreo.

### 3.12.1 Subrutina de calibración

La subrutina de configuración tiene como objetivo recibir y transmitir los diferentes niveles de tensión a establecer en los canales del DAC para el ajuste de los sensores y filtros de la tarjeta de adquisición de datos. El diagrama de flujo correspondiente se ilustra en la figura 3.18.

Los parámetros a recibir y transmitir en la presente rutina son los siguientes:

- Nivel de tensión para el LED del sensor 1 de fotopleitismografía.

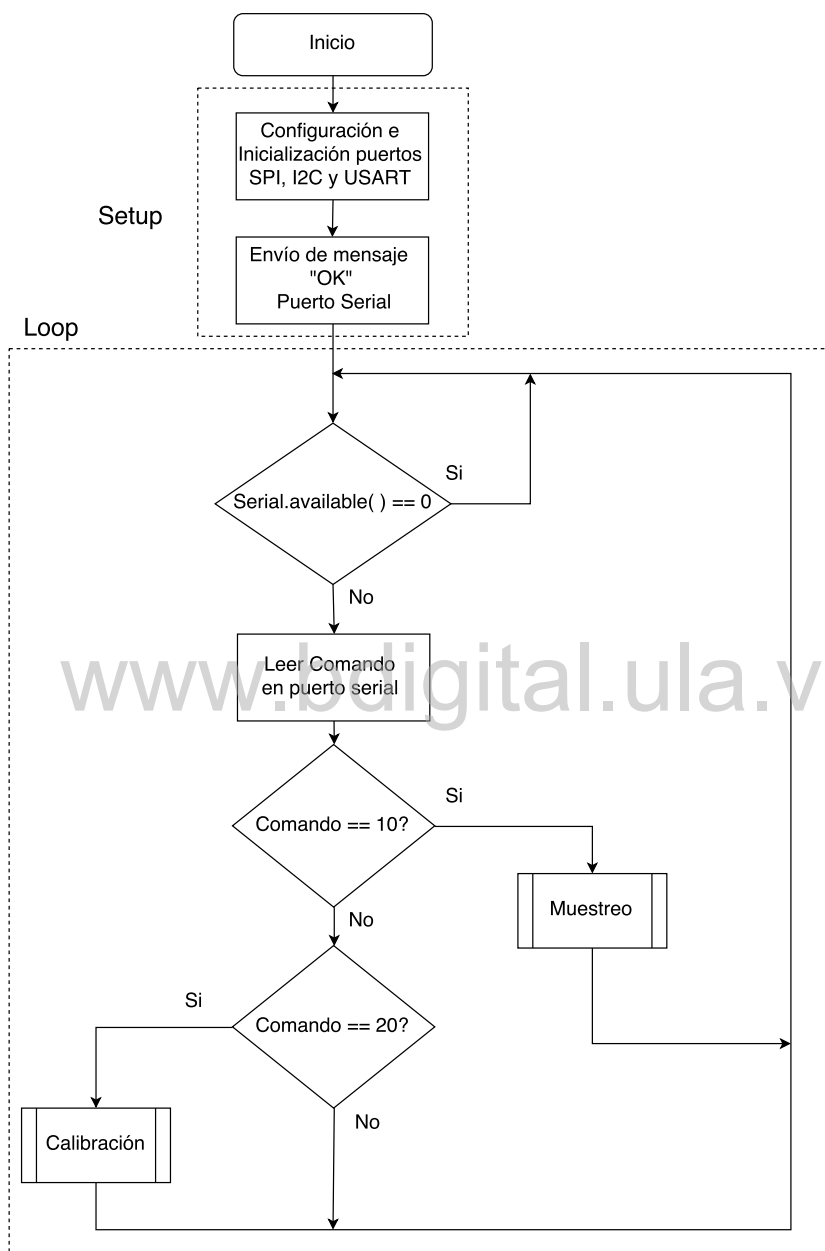
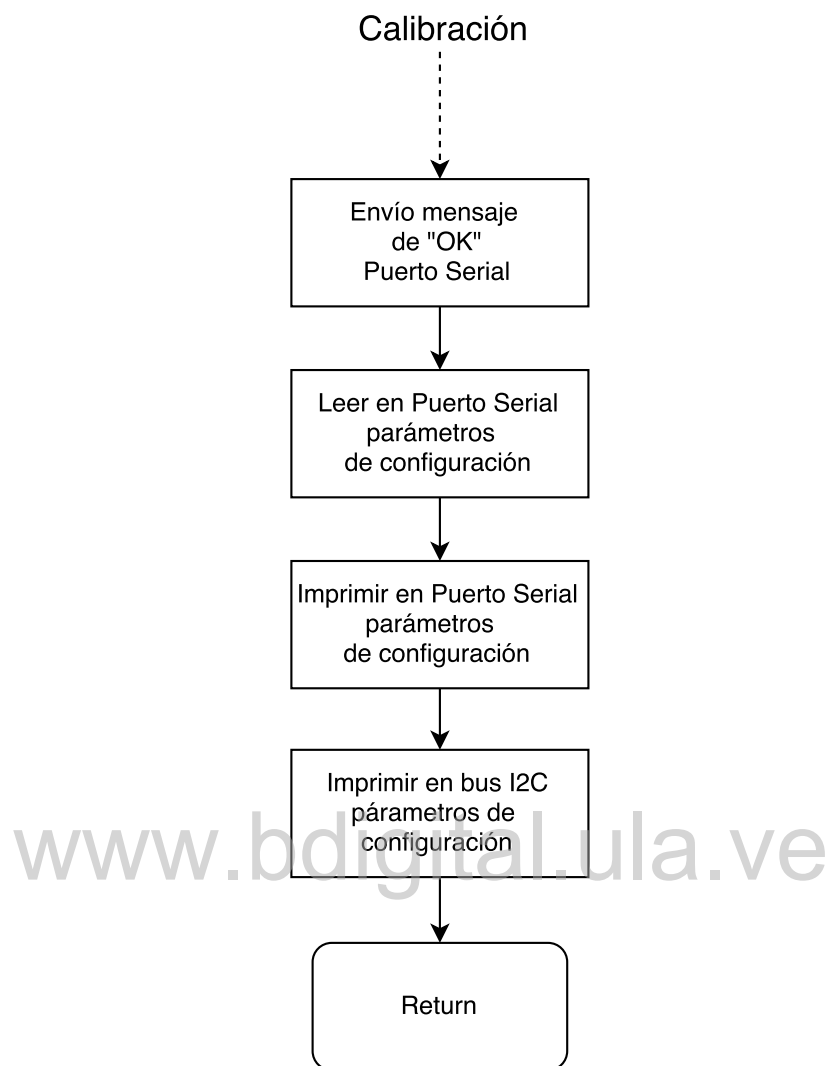


Figura 3.17. Diagrama de flujo general en programación del arduino



**Figura 3.18. Diagrama de flujo de la subrutina de calibración**

- Nivel de tensión para el LED del sensor 2 de fotopleletismografía.
- Nivel de tensión para el nivel cero de la etapa de filtros de la señal de pulso 1.
- Nivel de tensión para el nivel cero de la etapa de filtros de la señal de pulso 2.
- Número de muestras a tomar por señal.

### 3.12.2 Subrutina de Muestreo

En la subrutina de muestreo se realiza la adquisición y transmisión de los datos al computador personal. En la figura 3.19 se detalla el diagrama de flujo correspondiente.

En el inicio de la subrutina se ejecutan dos acciones claves: la declaración de variables locales de temporización y la escritura del puerto serial para indicar al computador personal que el Arduino está preparado para iniciar el muestreo.

Cabe destacar que la función *muestreo* depende de la función *calibración* por medio de la variable *NumMuestras* (ilustrada en la figura 3.19). Esta variable define el número de muestras a tomar por señal y en consecuencia el número de repeticiones del lazo *For* detallado en el diagrama de flujo.

A continuación se ejecuta el lazo *For* mencionado, donde en cada ciclo se realizan en orden las siguientes acciones:

- Se efectúa la conversión analógica digital de la señal de pulso 1 y se envía el dato al puerto serial.
- Se recibe una muestra de la señal ECG por medio del bus de datos SPI y se envía al puerto serial.
- Se efectúa la conversión analógica digital de la señal de pulso 2 y se envía el dato al puerto serial.
- Se recibe una muestra de la señal ECG por medio del bus de datos SPI y se envía al puerto serial.

Adicionalmente y de forma simultánea se realiza la medición de los tiempos de cómputo de cada una de las acciones detalladas.

Al final de cada ciclo de la sentencia *For*, los tiempos de cada una de las acciones detalladas son acumulados en variables de temporización, todo ello, en conjunto con la medición del tiempo de cómputo global de la sentencia *For*.

La temporización se efectuó con el empleo de la función *micros()* del lenguaje Arduino, que retorna el número de microsegundos transcurridos desde el inicio del programa hasta la función *micros()* en ejecución. Dicho número cabe acotar, vuelve a cero (se desborda) luego de 70 minutos y tiene una resolución de 4  $\mu s$  para el Arduino Uno.

Para la temporización de un bloque de instrucciones con la función *micros()* sólo es necesario restar los microsegundos transcurridos antes y después de la ejecución de éste. El resultado de la resta es el tiempo de cómputo en cuestión.

Con respecto a las temporizaciones en el lazo *For*, cuando se han ejecutado todas las repeticiones, se obtienen los siguientes valores:

- El tiempo de cómputo acumulado del muestreo de la señal de pulso 1.
- El tiempo de cómputo acumulado del muestreo de la señal de pulso 2.
- El tiempo de cómputo acumulado de la adquisición de la señal ECG.
- El tiempo global de la ejecución del lazo *For*.

Dichas variables son entonces transmitidas al puerto serial con la finalidad de ser procesadas para la determinación de la frecuencia de muestreo efectiva, así como para la determinación de los tiempos de adquisición de cada una de las muestras de las señales obtenidas.

[www.bdigital.ula.ve](http://www.bdigital.ula.ve)

### 3.12.3 Comunicación serial (protocolo RS-232)

Un punto destacable en la programación del Arduino en conjunto con la del computador personal detallada en la sección 3.13.1.1 es la comunicación serial por medio del protocolo RS-232.

El microcontrolador Atmega328 de la placa Arduino Uno posee un módulo USART que es compatible con el protocolo RS-232 utilizado en las computadoras personales. Sin embargo, es bien sabido que los ordenadores actuales generalmente no disponen de un puerto serial con conector R2-232. Por ello la placa Arduino Uno dispone de un microcontrolador Atmel ATmega8U2 que establece la interfaz entre el módulo USART y el puerto USB del computador. Esto habilita la comunicación serial por protocolo USB (Com Virtual).

Ha de detallarse además que la transferencia vía USB puede realizarse con mucha más rapidez que por el protocolo R2-232 y en consecuencia, la velocidad máxima de transmisión por el puerto serial con el Arduino está limitada a la velocidad máxima normalizada de 115200 *bits/seg* del bus R2-232.

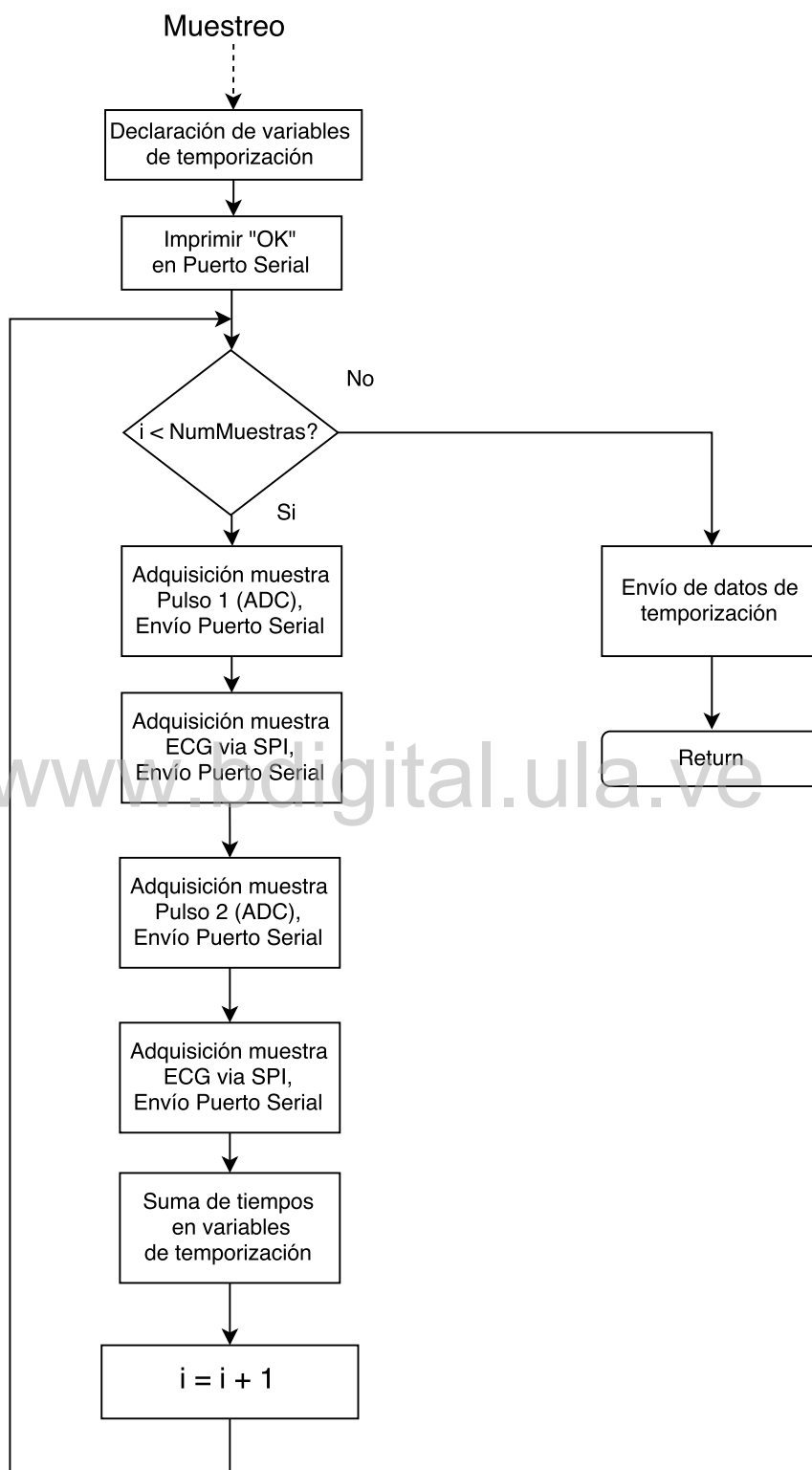


Figura 3.19. Diagrama de flujo de la subrutina de muestreo



En cuanto al conjunto de instrucciones de comunicación del Arduino, en el paquete de software de Arduino existen varias funciones para el envío y recepción de datos por el puerto serial. La más utilizadas son *Serial.println()* y *Serial.read()*. El uso de estas funciones resulta de utilidad para la transferencia o lectura de mensajes cortos entre dispositivos pues cumple con las necesidades de múltiples aplicaciones específicas. No obstante, dichas funciones tienen un comportamiento increíblemente ineficiente en aplicaciones donde se necesita una transferencia de datos de forma continua y con la mayor velocidad posible.

La función *Serial.println()* por ejemplo, transforma el valor o el *string* contenido dentro de los paréntesis en código *ASCII* y lo transmite bajo esa codificación. En consecuencia, para escribir un número entero de 8 bits en el puerto serial como por ejemplo (200), es necesario transferir de 33 bits por el puerto en cuestión ( 8 bits por cada carácter contando un salto de línea y un bit de parada).

La solución a dicha ineficiencia fue resuelta con el envío de datos de forma binaria utilizando la función *Serial.write()*, función de Arduino mucho menos utilizada por implicar mayor cuidado en el envío de datos y por el aumento en la dificultad de su lectura en el dispositivo receptor.

Por último, se detalla la configuración de los parámetros del puerto serial utilizada:

- Velocidad de transmisión de 115200 *bits/seg*.
- Tamaño del *Byte* de transferencia: 8 bits (en la comunicación por el puerto serial el *Byte* de transferencia puede ajustarse a otros tamaños).
- Un bit de parada.

### 3.13 Programación en el computador personal

La programación en el computador personal se realizó en Python. Python es un lenguaje de alto nivel de propósito general, interpretado, multiplataforma y con licencia de código abierto (Software libre).

La escogencia de dicho lenguaje se efectuó en base a su facilidad de uso y versatilidad en aplicaciones de ciencias e ingeniería. Python presenta múltiples librerías que guardan

cierta similitud con entornos comerciales como Matlab, tales como: SciPy, NumPy, Matplotlib, entre otros.

El desarrollo de la aplicación en Python puede dividirse en dos partes: *front-end* y *back-end*. La primera está relacionada con la interfaz gráfica de usuario y la segunda al motor o al código (que no está al alcance del usuario) donde se realiza la transmisión, recepción y procesamiento de la datos.

Cabe destacar, que la aplicación desarrollada es multiplataforma, por lo que puede ser utilizada en Linux (probada en distribuciones basadas en Ubuntu) y en Windows (probada en XP y 7).

En las secciones siguientes se hará una descripción detallada acerca de la programación en las partes mencionadas.

### 3.13.1 *Back-end*

De manera más específica dentro del código de *back-end*, el computador tiene las siguientes funciones o responsabilidades:

- Enviar por el puerto serial los parámetros de configuración al Arduino: nivel de tensión en los sensores y filtros, y número de muestras a tomar por señal para la medición de la VOP.
- Recibir por el puerto serial las dos señales de pulso y de ECG, así como los tiempos de adquisición de éstas.
- Almacenar la información del paciente y las señales adquiridas en archivos ubicados en memorias no volátiles (disco duro o memoria flash).
- Ejecutar los algoritmos para la detección de puntos de medición en las señales y realizar el cálculo de la VOP.

La programación de las acciones mencionadas se realizó con orientación a objetos y en algunos casos a base de funciones o métodos. En las subsecciones siguientes se detallan los distintos módulos desarrollados para el *back-end* de la aplicación:

### 3.13.1.1 Comunicación serial (protocolo RS-232)

En el lenguaje Python no existen módulos nativos para establecer una comunicación vía puerto serial, sin embargo, existen librerías de licencia libre altamente documentadas y muy utilizadas por la comunidad. Tal es el caso de *pySerial*, módulo que habilita la comunicación por puerto serial virtual (COM) con otros dispositivos (compatibles con protocolo RS-232), para lo cual fue utilizada en el presente trabajo.

La ventaja de este módulo es que además de las características mencionadas, posee una cantidad considerable de funciones sumamente útiles para establecer comunicaciones de forma efectiva por medio del puerto serial. Y por otra parte, es compatible con diferentes plataformas o sistemas operativos como Linux, Windows y MacOS.

De manera similar a la mencionada en la sección 3.12.3 la comunicación serial fue establecida con la siguiente configuración:

- Velocidad de transmisión de 115200 *bits/seg*.
- Tamaño del *Byte* de transferencia de 8 bits (en la comunicación por el puerto serial éste ajustarse a otros tamaños).
- Un bit de parada.
- Tiempo de espera por dato o *timeout* de 0.2 seg.

La programación de la comunicación serial además se realizó con orientación a objetos. Es decir que se programó una clase llamada *ComSerial* conteniendo funcionalidades para la comunicación serial específicas de la aplicación.

La clase *ComSerial* dispone de varios métodos. Los más relevantes se detallan en la lista siguiente:

- *Handshake*, donde se efectúa la sincronización de comunicación o el *hanshaking* entre el dispositivo Arduino y el computador.
- *Calibración*, donde se realiza la transmisión y recepción de parámetros de calibración para el circuito principal.
- *EstadoPuerto*, provee información del estado del puerto (abierto o cerrado).

A continuación se presentan los primeros dos métodos mencionados de la clase *Com-Serial*:

**3.13.1.1.1 Handshake** La sincronización en la comunicación entre el Arduino y el computador o *handshaking* consiste en realizar lecturas sucesivas en el puerto serial hasta obtener un dato válido. Por medio de la programación en Python, se realizan lecturas esperando un dato tipo *string* que contenga un mensaje de (“OK”) enviado por el Arduino. En caso de no obtener dicho mensaje, se reinicia el proceso de espera.

Adicional a ello, se utiliza una variable de conteo para enumerar el número de fallas de lectura, si éste llega a un número determinado (20 por ejemplo) entonces se imprime un mensaje de “procedimiento fallido”. El diagrama de flujo del método *handshake* puede observarse en la figura 3.20.

El método *handshake* es especialmente útil cuando se inicia un programa en Python que implique el uso del puerto serial con el Arduino, pues en el momento de apertura del puerto, el Arduino se reinicia y entra en un periodo de configuración que tiende a tener una duración considerablemente variable para los efectos de esta aplicación. En consecuencia y sin una rutina de sincronización, el ordenador puede enviar instrucciones vía puerto serial cuando el Arduino aún no ha terminado de configurarse, originando errores en la comunicación entre los dispositivos.

**3.13.1.1.2 Calibración** En el método de calibración se transmiten y reciben de vuelta los parámetros de configuración requeridos por el DAC y por el Arduino en el circuito general. En la figura 3.21 se observa el diagrama de flujo correspondiente.

La comunicación se realiza también vía puerto serial virtual (protocolo RS-232) con el dispositivo Arduino. De manera similar al detallado en la sección 3.12.1, los parámetros transmitidos corresponden a los niveles de tensión de: los LEDs en los sensores por fotople-tismografía, los niveles de cero en los amplificadores de los filtros analógicos que trabajan con alimentación (0 – 5)V y un número entero que refiere la cantidad de muestras a tomar por cada señal en los circuitos.

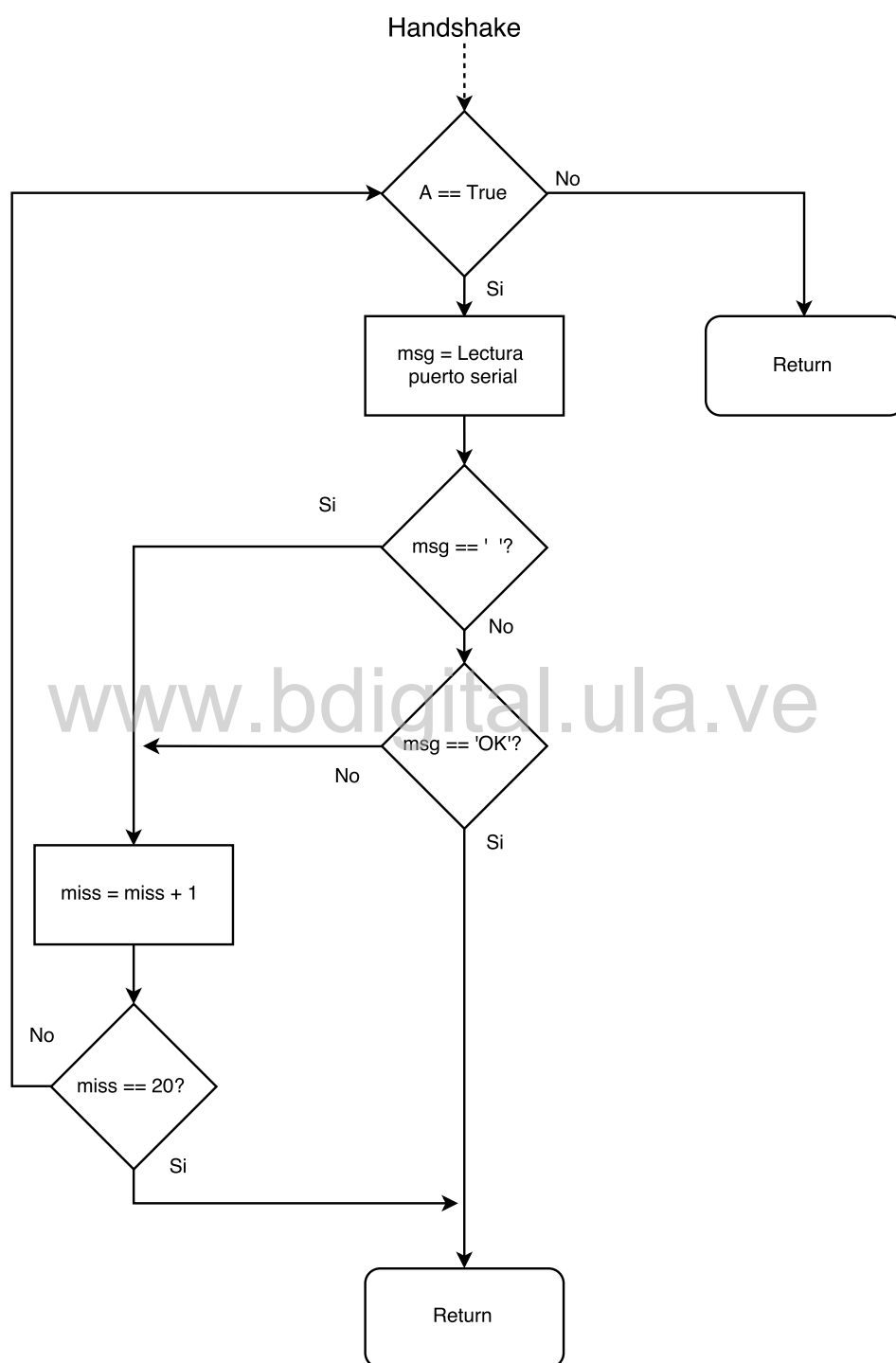
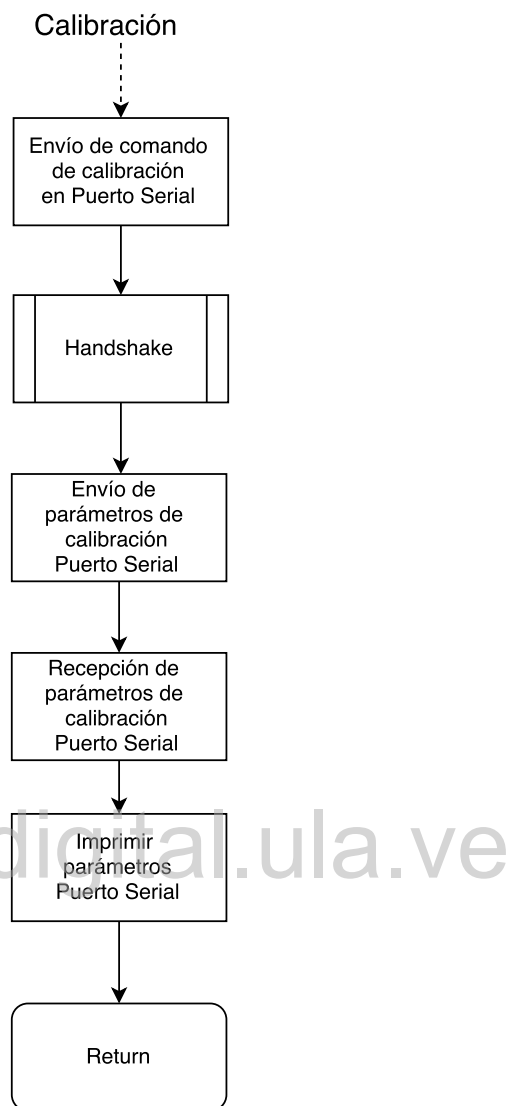


Figura 3.20. Diagrama de flujo del método *handshake* en Python



**Figura 3.21. Diagrama de flujo del método calibración en Python**

### 3.13.1.2 Subrutina de Muestreo

En la subrutina de muestreo se realiza la adquisición y ploteo en tiempo real de las bioseñales del paciente. Además de ello, también se reciben las variables de temporización mencionadas en la sección 3.12.2. En la figura 3.22 se observa el diagrama de flujo correspondiente.

La subrutina de muestreo contiene un proceso denominado *TiemposMuestras* en el que se reciben variables de temporización. Estas variables son las siguientes:

- *TiempoPulso1*, representa el tiempo de cómputo total requerido para las conversiones

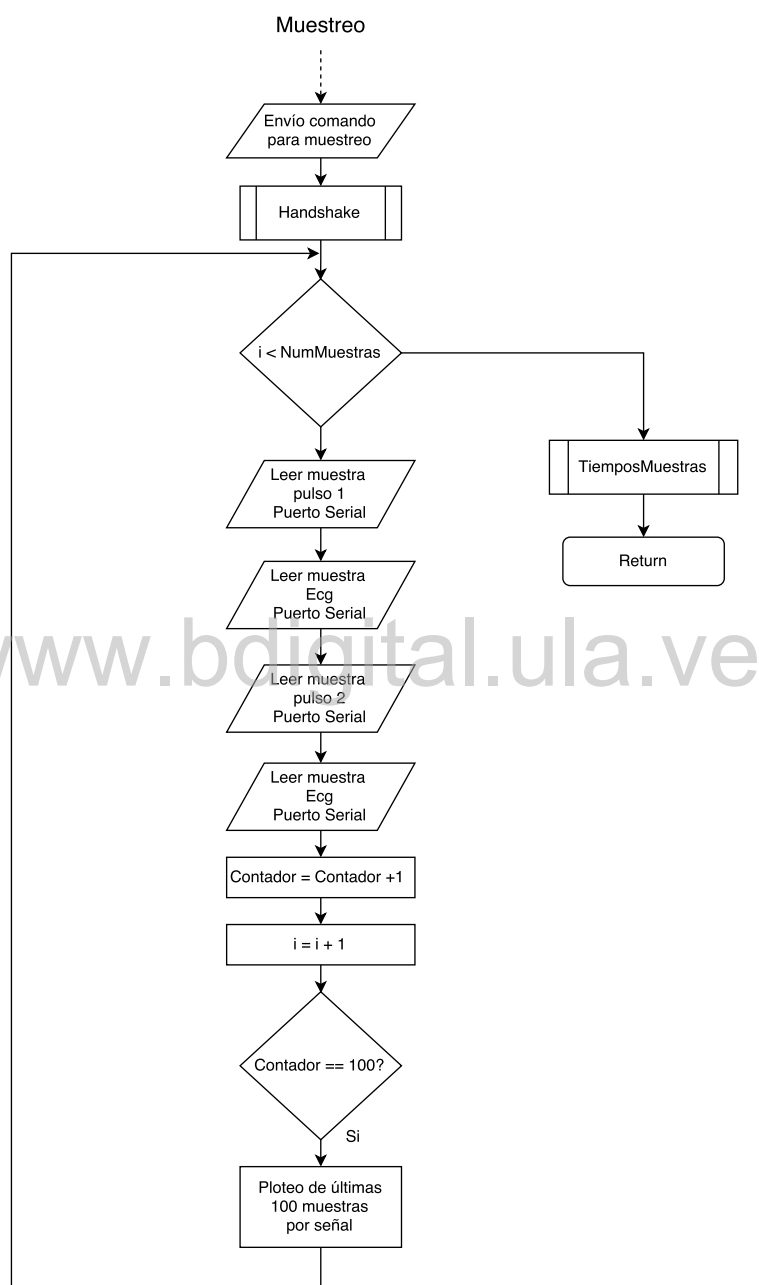


Figura 3.22. Diagrama de flujo de la subrutina de muestreo en Python

analógico digital de todas las muestras de la señal de pulso 1 y su envío al puerto serial.

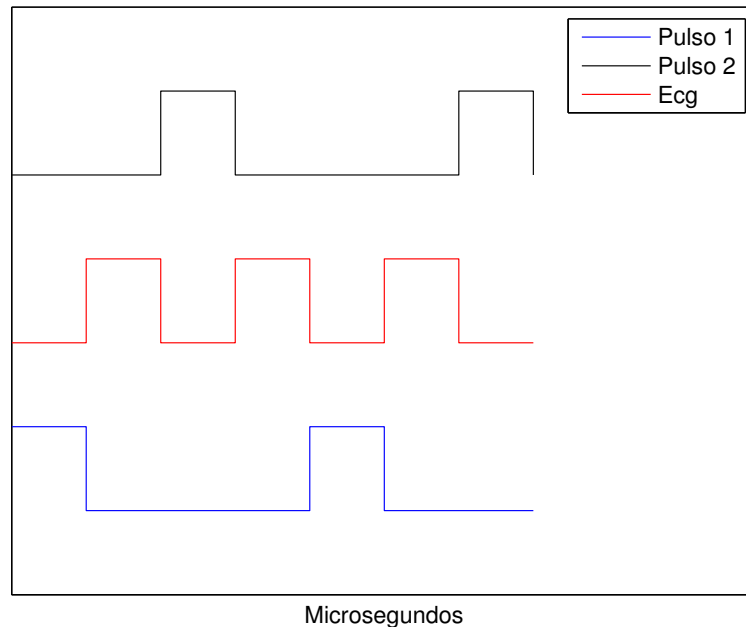
- *TiempoPulso2*, representa el tiempo de cómputo total requerido para las conversiones analógico digital de todas las muestras de la señal de pulso 2 y su envío al puerto serial.
- *TiempoECG*, representa el tiempo de cómputo total requerido para la adquisición de las muestras de la señal electrocardiográfica vía SPI y su envío al puerto serial.
- *TiempoTotalMuestreo*, representa el tiempo de cómputo de todo el proceso de adquisición y envío de datos realizado por el Arduino. Proceso que es realizado dentro de un lazo *For* en dicho dispositivo detallado en la sección 3.12.2.

Las variables de temporización mencionadas representan la sumatoria de los tiempos en todos los ciclos del *For*. Si por otra parte se detalla un solo ciclo, los tiempos individuales de cada uno los bloques de muestreo se distribuyen de acuerdo a lo ilustrado en la figura 3.23. Se puede observar que en cada ciclo, se obtiene una muestra de la señal de pulso 1, una muestra del pulso 2 y dos de la señal electrocardiográfica. Adicionalmente, se tiene un retardo (el cual no está presente en la figura 3.23) que está relacionado con el salto dado hacia el inicio de la siguiente repetición (la sumatoria de este tiempo está incluida en la variable *TiempoTotalMuestreo*).

Las variables de temporización resultan de utilidad en la determinación de los tiempos de adquisición de las muestras, así como para proveer información sobre la frecuencia de muestreo a la que han sido muestreadas. En base a ello, la ecuación 3.4 estima el tiempo requerido entre muestra y muestra para la señal de pulso 1. La ecuación considera todas las variables mencionadas considerando además el número de muestras obtenidas en las distintas señales, de manera que *tMuestP1* es el promedio del tiempo requerido para obtener una muestra de pulso1.

$$tMuestP1 = \frac{tiempoPulso1}{NumMuestrasPulso1} + \frac{tiempoPulso2}{NumMuestrasPulso2} + \frac{tiempoEcg}{NumMuestrasEcg} + \frac{tiempoTotalMuestreo}{NumRepFor} \quad (3.4)$$





**Figura 3.23. Tiempos de muestreo**

Como el proceso de muestreo de la señal de pulso 1 y la señal de pulso 2 es el mismo, entonces puede decirse que los tiempos entre muestra y muestra de ambas señales son iguales:

$$tMuestP2 = tMuestP1 \quad (3.5)$$

Por otra parte, como se detalla en el diagrama de flujo de la figura 3.22, en cada ciclo se obtienen dos muestras de ECG a diferencia de las de Pulso, en las que se obtiene una muestra por ciclo respectivamente. Este argumento ejerce influencia en el tiempo entre muestras en la señal ECG, ver ecuación 3.6.

$$tMuestEcg = \frac{tMuestP1}{2} \quad (3.6)$$

En base a las ecuaciones detalladas y a experiencias con el funcionamiento del equipo, puede detallarse que las frecuencias de muestreo efectiva para las señales de Pulso es de aproximadamente 800 Hz. La señal electrocardiográfica es muestreada a una frecuencia de 1700 Hz.

Finalmente se construyen los vectores de tiempos de adquisición de las muestras. El

tamaño de estos vectores depende del número de muestras de las señales. Cada vector se construye considerando un incremento (que puede ser  $tMuestP1$ ,  $tMuestP2$  o  $tMuestEcg$  dependiendo de cual corresponda) y el tiempo inicial. El tiempo inicial es diferente porque las muestras de las señales no son tomadas de forma concurrente (ver nuevamente figura 3.23), de esta manera el vector de tiempos de la señal de pulso 1 inicia en 0.0 segundos, por ser la muestra inicial. En cambio, el vector de tiempos de la señal ECG inicia en  $(0.0 \text{ seg} + tMuesP1)$  segundos.

Finalmente, se construyen vectores lineales cuyas longitudes corresponden al número de muestras de las señales y con incrementos dados por las combinaciones de las magnitudes  $tMuestP1$ ,  $tMuestP2$  y  $tMuestEcg$  para cada señal respectivamente. Se ha considerado además el orden de adquisición de los datos de manera que la muestra obtenida en el tiempo 0.0 segundos pertenece a la señal de pulso 1 (ver nuevamente figura 3.23), la primera de ECG es obtenida en  $0.0 + tMuestP1$  segundos y el primer dato del pulso 2 en  $tMuestP1 + tMuestEcg$  segundos.

### 3.13.1.3 Manejo de archivos

El entorno desarrollado para la medición de la VOP requiere un sistema de almacenamiento de datos en una memoria no volátil para la lectura de los datos, posterior a la medición hecha al paciente.

El almacenamiento de los datos se implementó en archivos de formato *.csv* (*Comma-separated values*) y *.txt* (texto). Para cada medición de la VOP por paciente se crean como mínimo dos archivos de almacenamiento, uno de cada formato mencionado. El archivo CSV guarda la información de las bioseñales adquiridas y el archivo de texto los datos personales del paciente y el reporte clínico. Cabe destacar además, que el formato mencionado *.csv* es compatible con entornos como Matlab, Excel y por supuesto, Python.

Los archivos son almacenados en una carpeta con el número de consulta realizada al paciente. Dicha carpeta a su vez, es ubicada en otra cuyo nombre contiene el número de cédula y nombre del paciente. Esta estructura permite organizar las mediciones por paciente y número de consulta, ver figura 3.24.

Cabe destacar que dentro de cada consulta, el profesional de la salud o investigador,

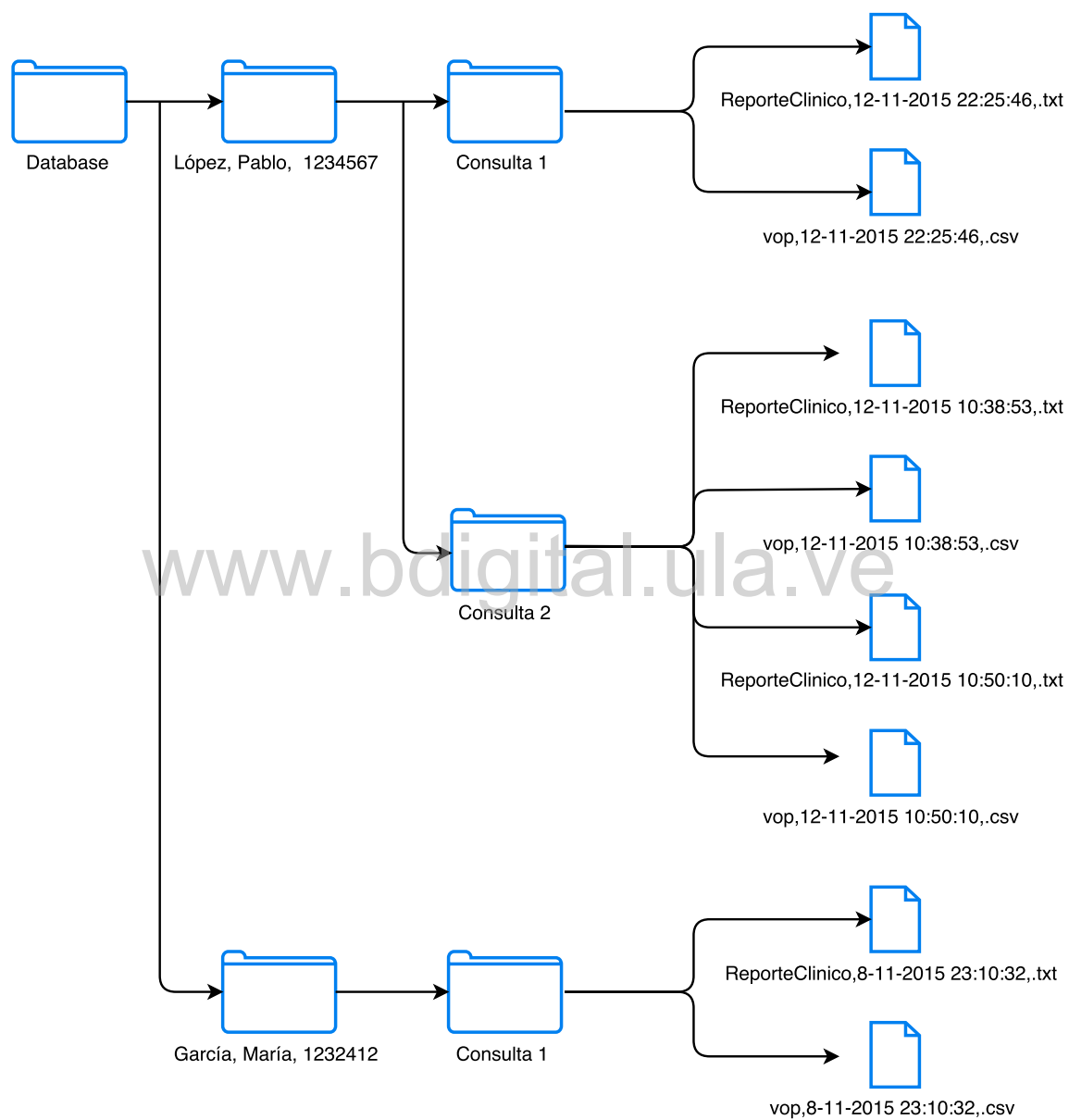


Figura 3.24. Estructura organizacional del almacenamiento de los datos

podrá realizar varias mediciones de VOP y éstas se almacenaran en sus respectivos archivos CSV y .txt.

En cuanto al manejo de los archivos con los formatos mencionados. Se utilizaron las siguientes librerías de Python:

- *Glob*, para la lectura y búsqueda de archivos.
- *OS*, para funcionalidades del sistema operativo, como la creación y el manejo de carpetas.
- *CSV*, para la escritura y lectura de archivos .csv.

#### 3.13.1.4 Detección de puntos máximos en la señal electrocardiográfica

El procesamiento de la onda ECG se requiere para el cálculo de la velocidad de la onda de pulso porque las ubicaciones de los picos R de la señal marcan los intervalos de cada ciclo cardíaco. Dichos intervalos demarcan las regiones de búsqueda para los algoritmos de identificación de puntos de las ondas de pulso.

La señal ECG originalmente es adquirida por medio de la disposición de electrodos en el paciente (dispuestos para obtener la primera o la segunda derivación de acuerdo al triángulo de Einthoven) y un circuito amplificador. La señal posteriormente es digitalizada y enviada al Arduino donde luego es transmitida al computador personal.

En todo el proceso mencionado la onda ha sido sometida solo a un filtrado analógico, por lo que requiere de un filtrado digital para disminuir los componentes de alta frecuencia, con el fin de facilitar la identificación de los puntos máximos o picos R de la señal.

La onda ECG es sometida a diferentes procesos para la detección de sus picos R. En la figura 3.25 puede observarse de manera general el procedimiento realizado. De esta manera el procesamiento digital de la onda ECG puede resumirse en los siguientes pasos:

- Lectura de la señal original.
- Eliminación de ruido de alta frecuencia por medio de un filtro FIR pasa bajas de 50 Hz y 31 coeficientes.
- Reducción de la línea base de la señal.

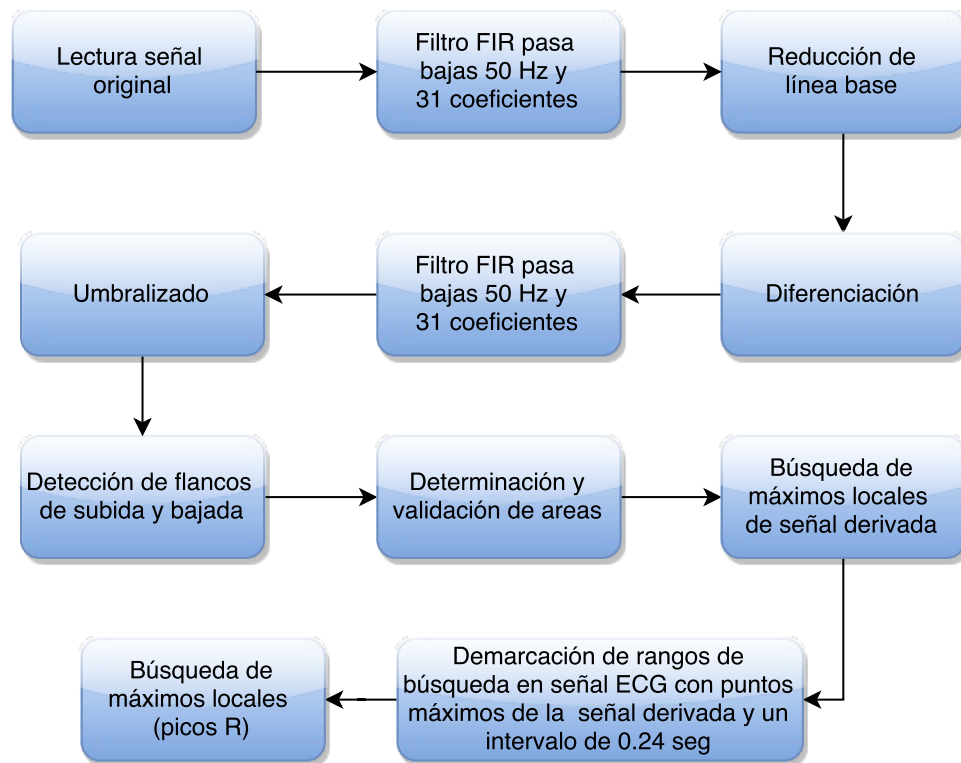
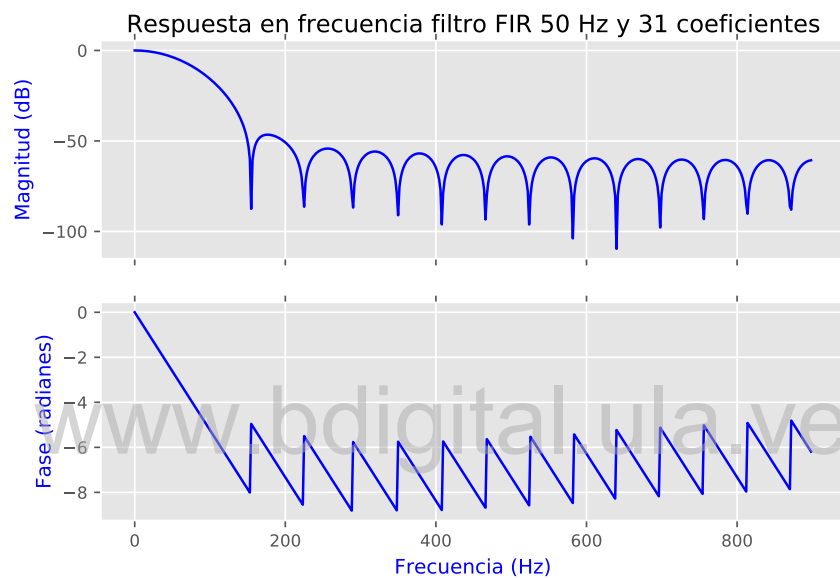


Figura 3.25. Diagrama de bloques de algoritmo de detección de picos R en señal ECG

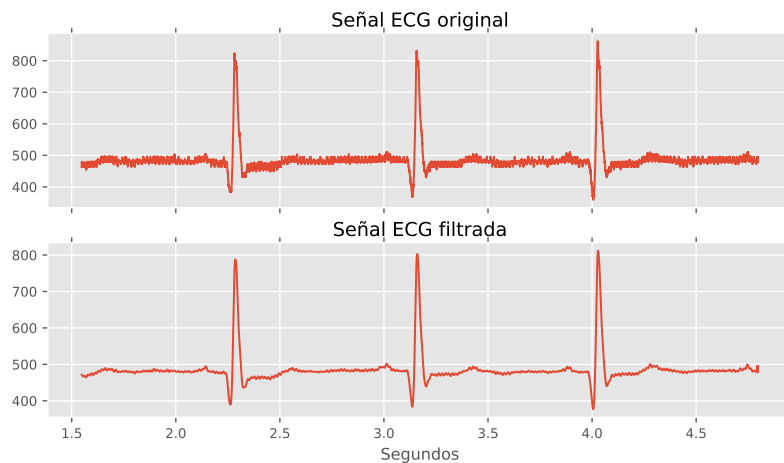
- Diferenciación de la señal filtrada:  $y[n] = x[n] - x[n - 1]$ .
- Eliminación de interferencias residuales con un filtro FIR pasa bajas de 50 Hz y 31 coeficientes.
- Umbralizado de la señal diferenciada y filtrada.
- Utilización de la señal resultante (compuesta por flancos de subida y bajada) para determinar el tamaño y ubicación de las zonas donde pueden existir los máximos locales.
- Validación de las zonas detectadas en base a su tamaño (número de muestras que abarcan).
- Localización de los puntos máximos (de forma numérica) de cada una de las zonas validadas de la onda diferenciada y filtrada.
- Búsqueda de los puntos máximos de la señal ECG a partir de los puntos máximos de su derivada y un intervalo de 0.24 segundos.

A continuación se detalla más exhaustivamente el algoritmo desarrollado:

Como la señal ECG recibida por el computador presenta altos niveles de ruido, se hace necesario el uso de filtros digitales. En consecuencia, se utilizó un filtro FIR pasa bajas de 31 coeficientes (de ventana *Hamming*) y frecuencia de corte de 50 Hz para reducir las componentes ruidosas de la señal. En la figura 3.26 se observa la respuesta en frecuencia del filtro y sus coeficientes en la tabla 3.1. Por otra parte, en la figura 3.27 se presenta la señal original ECG ruidosa y filtrada.



**Figura 3.26.** Respuesta en frecuencia de filtro FIR 50 Hz (31 coeficientes)



**Figura 3.27.** Filtrado de la señal ECG

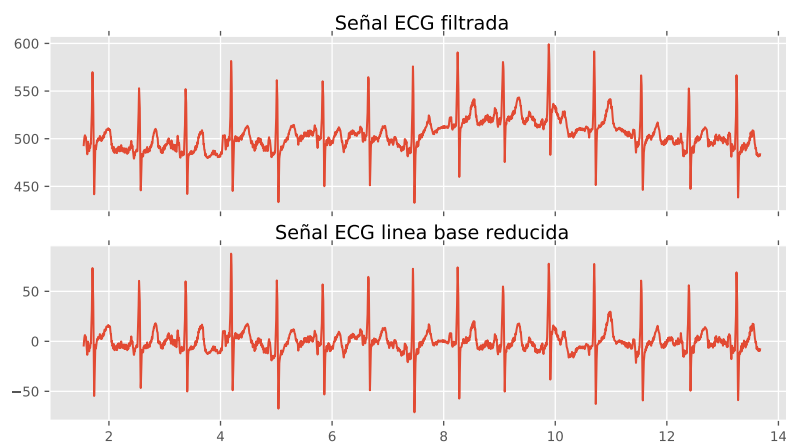
En algunas circunstancias durante el tiempo de muestreo de las señales, la onda ECG

**Cuadro 3.1. Valores de coeficientes del filtro FIR pasa bajas de 50 Hz**

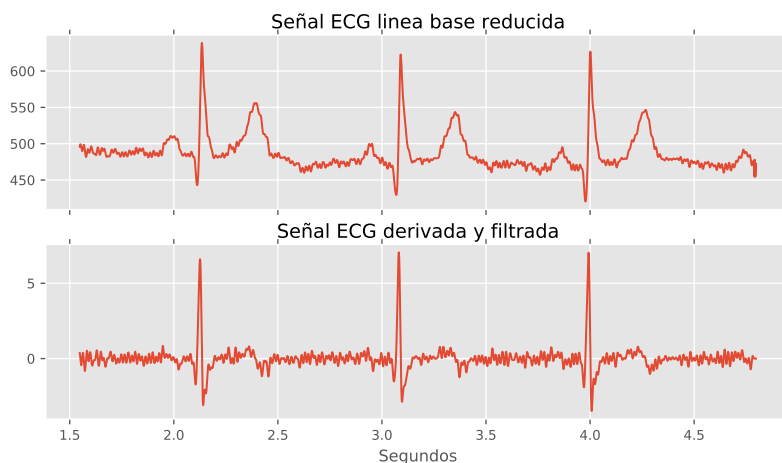
#	FIR	#	FIR	#	FIR
1	0.00112222	12	0.05733397	23	0.0332229
2	0.00173998	13	0.06397717	24	0.0254838
3	0.00297007	14	0.06910483	25	0.01860331
4	0.00509785	15	0.07234251	26	0.01284766
5	0.00834769	16	0.07344943	27	0.00834769
6	0.01284766	17	0.07234251	28	0.00509785
7	0.01860331	18	0.06910483	29	0.00297007
8	0.0254838	19	0.06397717	30	0.00173998
9	0.0332229	20	0.05733397	31	0.00112222
10	0.04143515	21	0.04964617		
11	0.04964617	22	0.04143515		

puede presentar un cambio en su línea base, es decir, un desplazamiento con respecto a su eje promedio. El efecto mencionado (provocado comúnmente por el movimiento del paciente) reduce considerablemente la efectividad del algoritmo.

Para disminuir la presencia de dichos desplazamientos se realizaron regresiones lineales de grado 15, en manera de hacer un ligero *overfitting* de la señal. El polinomio generado es evaluado para cada valor de tiempo (eje de las abscisas) y el resultado es restado a la original. En la figura 3.28 puede observarse la eficacia del método mencionado.

**Figura 3.28. Reducción de línea de base (*baseline*) de la señal ECG**

De forma subsiguiente la onda es sometida a un proceso de diferenciación ( $y[n] = x[n] - x[n - 1]$ ) con el objetivo de enfatizar la zona perteneciente al complejo QRS de



**Figura 3.29. Diferenciación de la señal ECG**

la señal electrocardiográfica. Este paso en particular fue necesario debido a las variaciones morfológicas que presenta la onda entre diferentes pacientes. De esta manera, algunos pacientes presentan registros electrocardiográficos donde la proporción en cuanto a la amplitud del complejo QRS y las ondas P y T es bastante acentuada, pero en otros pacientes por el contrario, las ondas P y T pueden tener una altura comparable al pico R de la señal, lo que interfiere con el correcto funcionamiento del algoritmo.

Cabe destacar que la señal diferenciada presenta altos niveles de interferencia porque la operación de sustracción amplifica las pequeñas distorsiones. Por esta razón la señal es nuevamente procesada con un filtro FIR de 31 coeficientes y frecuencia de corte de 50 Hz. En la figura 3.29 se detalla la señal diferenciada de la onda ECG que además ha sido procesada con el filtro FIR mencionado (ver figura 3.26 para más detalles del filtro).

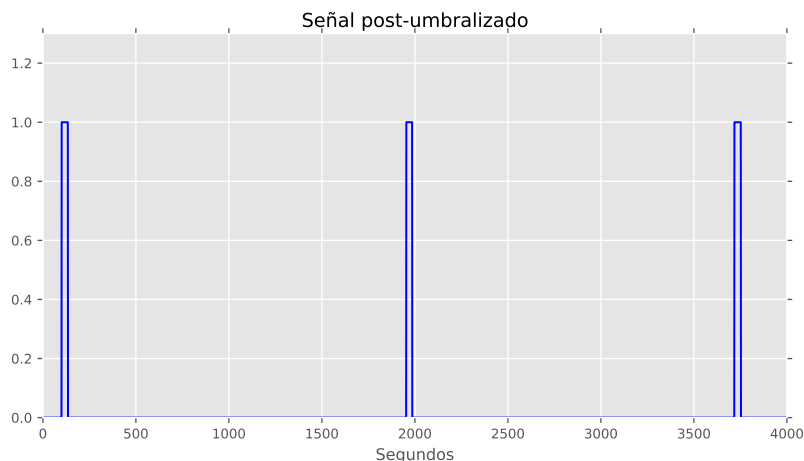
Posterior al filtrado de la señal diferenciada, es necesario un proceso de umbralización con la finalidad de descartar zonas donde no hay presencia de máximos locales. Se seleccionó como umbral, el valor máximo de la señal multiplicado por un factor de 0.5.

La señal resultante puede observarse en la figura 3.30. Los puntos con valor de 1 representan aquellos cuyas amplitudes son iguales o superiores al umbral asignado. Por otra parte, los puntos con valor de 0 corresponden a aquellos con amplitudes inferiores al umbral.

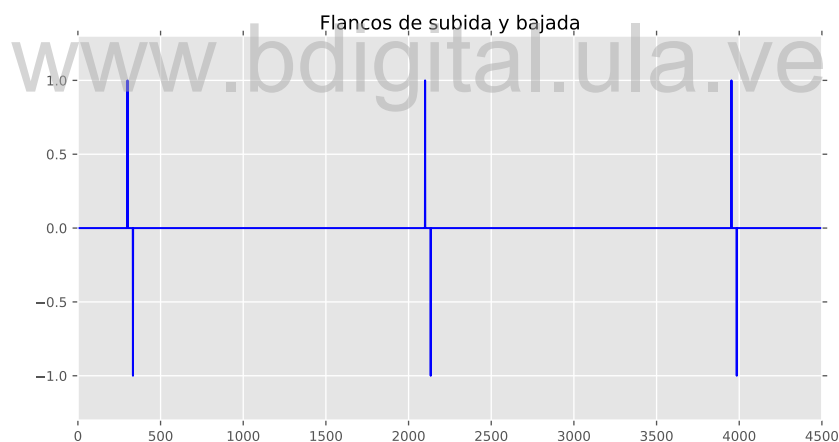
Seguidamente se realizó un proceso de diferenciación a la onda umbralizada ( $y[n] = x[n] - x[n - 1]$ ), obteniéndose la figura 3.31. La señal obtenida detalla flancos de subida o



bajada que representan los inicios o finales de áreas de máximos locales en la señal original. Las posiciones de dichos flancos permiten la localización de las áreas donde están presentes cada uno de los máximos locales.



**Figura 3.30. Señal post-umbralizado**



**Figura 3.31. Flancos de subida y bajada provenientes de la señal umbralizada**

Adicionalmente, las zonas identificadas fueron sometidas a un proceso de validación. Un ejemplo para enfatizar la necesidad de esta verificación es la presencia de ruido justo en el límite de amplitud del umbral, debido a que en esta situación el algoritmo puede detectar erróneamente áreas que sólo ocupan una muestra. Esto se produce porque una muestra ruidosa puede pasar las condiciones dadas por el umbral, pero no las muestras adyacentes. El resultado es un área de una muestra o bien pueden ser varias muestras pero en una cantidad muy inferior a la correspondiente de un área válida.

El proceso de validación mencionado consiste en la contabilización de muestras contenidas en cada una de las zonas identificadas. Si se considera que las señales adquiridas son cuasi-periódicas, entonces las zonas (o áreas en este contexto) tienen tamaños similares. De manera que si existen algunas que presentan amplias disimilitudes con el resto mayoritario, son consideradas erróneas.

Para la evaluación de las áreas se utilizó el operador de mediana con un rango o valor de tolerancia del 30 por ciento. Más detalladamente, se generó un vector de los tamaños de las diferentes zonas y se determinó la mediana. Con dicho valor y en conjunto con un factor de tolerancia, se determinó si cada uno de los tamaños satisfizo el rango establecido. En caso contrario, la zona correspondiente es eliminada.

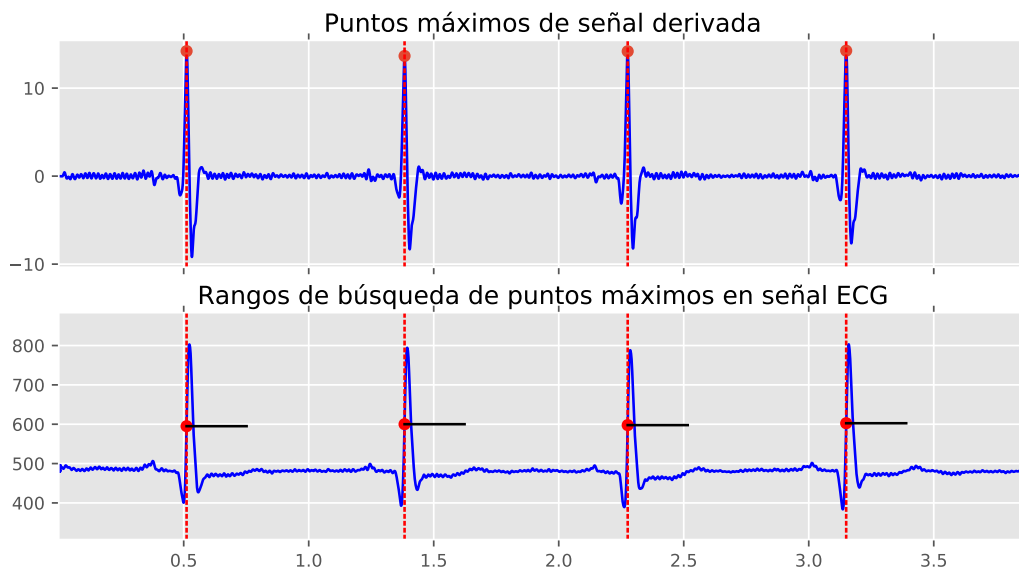
Posterior a la verificación de las zonas, se procedió a la búsqueda numérica de los puntos máximos en la señal derivada, dentro de las zonas verificadas. Los puntos identificados se ubicaron en los inicios del complejo QRS, de manera que a partir de la ubicación de dichos puntos se demarcaron límites de búsqueda hasta 0.24 segundos después del punto correspondiente en la señal ECG original filtrada. La limitación temporal de 0.24 segundos es llevada a cabo con el fin de evitar la detección de falsos puntos máximos en la onda T de la señal. En la figura 3.32 se observa un ejemplo de los puntos detectados para la señal ECG diferenciada y la demarcación de los rangos de búsqueda en la señal ECG no diferenciada (los rangos de búsqueda en cada periodo están marcados por las líneas negras en la figura 3.32).

Finalmente en la figura 3.33 se muestra la detección de puntos máximos en la señal ECG a partir de los rangos delimitados.

#### **3.13.1.5 Determinación de puntos de medición en las señales de pulso**

Los puntos de medición son puntos de referencia que deben ser identificados en cada periodo de la señal para el cálculo del tiempo de tránsito entre las señales de pulso y en consecuencia para la medición de la VOP.

En base a las investigaciones antecedentes puede decirse que las metodologías para la identificación de los puntos en las señales de pulso, están enfocadas en las zonas del pie de la señal (puntos mínimos) [15], picos máximos locales (puntos máximos) [28] y puntos



**Figura 3.32.** Delimitación de rangos de búsqueda de puntos máximos en señal ECG

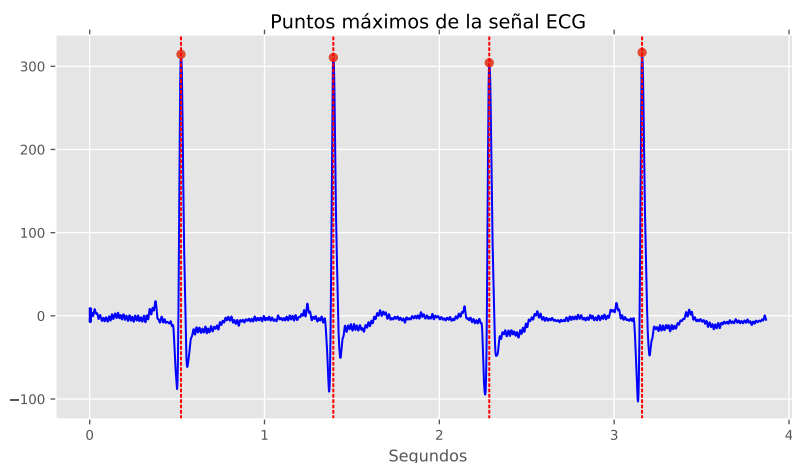
con más rápido crecimiento (tangente máxima) [12].

Considerando los métodos mencionados, se implementaron algoritmos basados en investigaciones antecedentes que buscan la tangente máxima o la segunda derivada máxima de los periodos de la señal [12, 15]. Por otra parte, se ha propuesto un método para la determinación de puntos máximos.

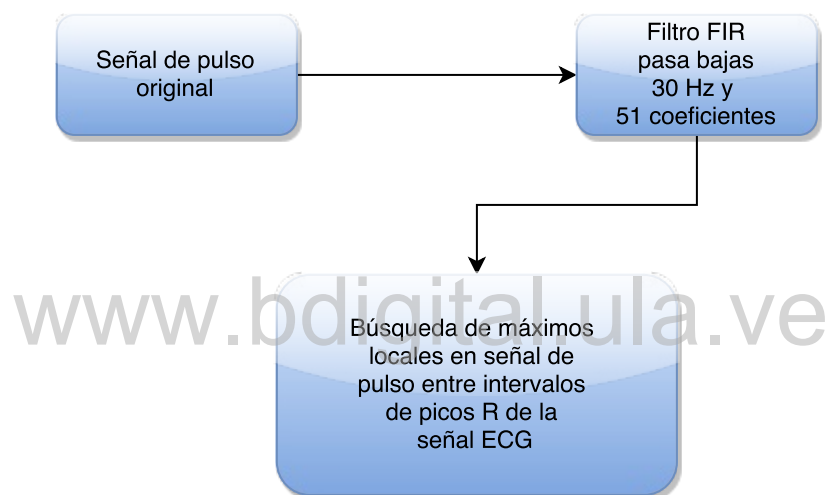
Por último es necesario enfatizar que en todos los métodos mencionados se ha incluido la señal electrocardiográfica como señal de referencia.

**3.13.1.5.1 Algoritmo de puntos máximos** El algoritmo de puntos máximos es un método propuesto en la presente investigación con la finalidad de localizar puntos máximos locales en las señales objetivo. En la figura 3.34 se detalla el diagrama de bloques del algoritmo.

A diferencia de la señal electrocardiográfica, la señal de pulso es recibida por el computador con un nivel de ruido más leve. No obstante, la señal es sometida a un proceso de filtrado con el propósito de incrementar la estabilidad y eficacia del método. Se utilizó entonces un filtro FIR de 51 coeficientes pasa bajas (ventana *Hamming*) con una frecuencia de corte de 30 Hz. Ver figura 3.35 para detalles de respuesta en frecuencia del filtro utilizado y la tabla 3.2 para observar los coeficientes correspondientes.



**Figura 3.33. Identificación de puntos máximos en la señal ECG**



**Figura 3.34. Diagrama de bloques de algoritmo de detección de puntos máximos**

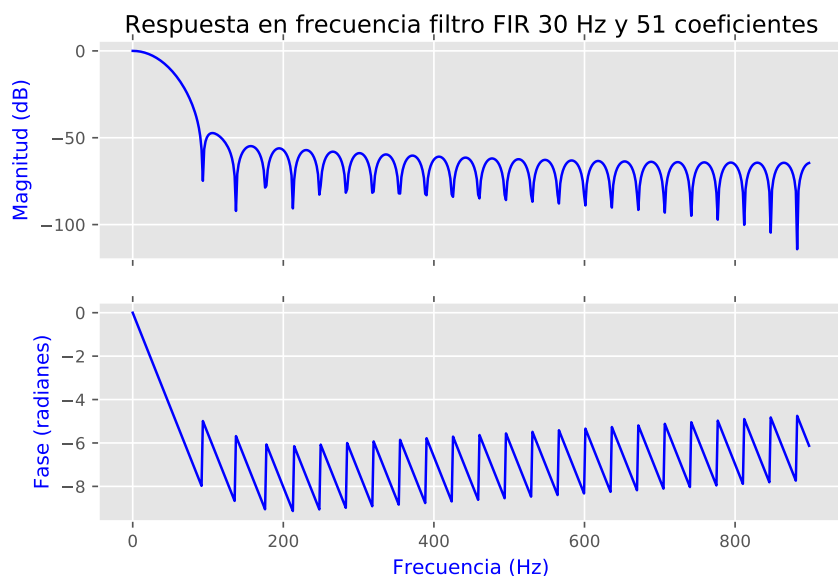
Además del filtrado, es importante destacar que el método presente es dependiente del algoritmo de detección de picos R de la señal ECG de la sección 3.13.1.4. Los picos de la onda R determinan los intervalos de búsqueda de máximos locales en la señal de pulso. La búsqueda se hace de forma numérica, es decir, se busca simplemente el valor máximo en cada intervalo.

En la figura 3.36 se muestran los puntos máximos identificados para dos señales de pulso y una onda ECG adquiridas de forma simultánea por el equipo desarrollado.

**3.13.1.5.2 Algoritmo de tangente máxima** Este algoritmo es implementado en [12] y se basa fundamentalmente en la determinación del punto de máxima pendiente en cada periodo de la señal, es decir, el punto con la tasa de cambio más alta.

**Cuadro 3.2. Valores de coeficientes del filtro FIR pasa bajas de 30 Hz**

#	FIR	#	FIR	#	FIR
1	0.00067361	18	0.03075803	35	0.02784817
2	0.00086227	19	0.03353182	36	0.02487117
3	0.00115686	20	0.03610141	37	0.02189447
4	0.00159709	21	0.03840187	38	0.01898173
5	0.00222037	22	0.04037371	39	0.01619079
6	0.00305995	23	0.04196505	40	0.01357195
7	0.00414312	24	0.04313343	41	0.01116651
8	0.00548959	25	0.04384737	42	0.00900572
9	0.00711017	26	0.04408753	43	0.00711017
10	0.00900572	27	0.04384737	44	0.00548959
11	0.01116651	28	0.04313343	45	0.00414312
12	0.01357195	29	0.04196505	46	0.00305995
13	0.01619079	30	0.04037371	47	0.00222037
14	0.01898173	31	0.03840187	48	0.00159709
15	0.02189447	32	0.03610141	49	0.00115686
16	0.02487117	33	0.03353182	50	0.00086227
17	0.02784817	34	0.03075803	51	0.00067361



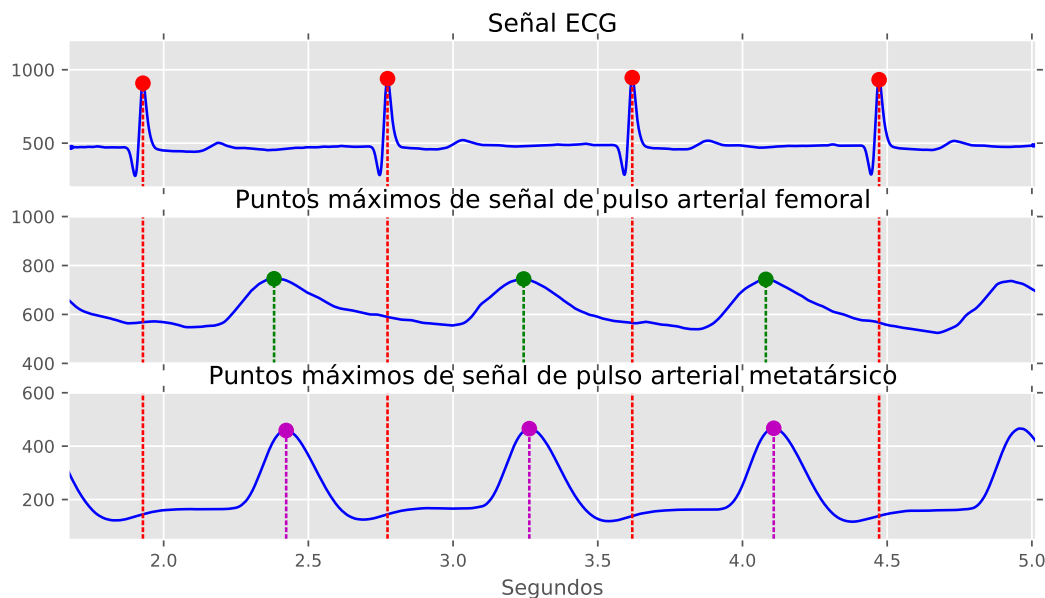
**Figura 3.35. Respuesta en frecuencia de filtro FIR 30 Hz (51 coeficientes)**

Este método aprovecha la morfología de la señal de pulso al considerar que en el periodo sistólico la onda de pulso tiene el comportamiento más lineal de todo el ciclo cardíaco. Aunado a ello, la forma de dicho periodo tiene una morfología bastante similar entre pacientes en relación con otras partes de la señal como el pie o el punto máximo. En la figura 3.37 se observa el diagrama de bloques del algoritmo en cuestión.

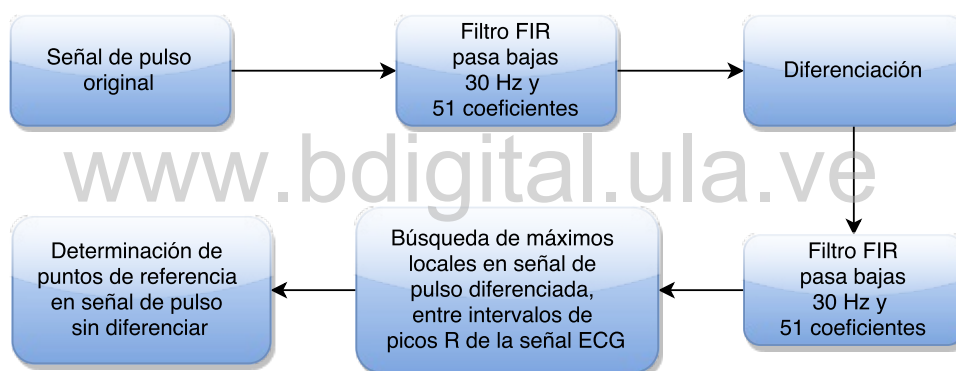
Para la determinación de los puntos de medición, se utilizaron los picos R de la onda ECG para la demarcación de los periodos de la señal. La determinación de puntos máximos para la ECG se realizó con el algoritmo detallado en la sección 3.13.1.4.

En cuanto al algoritmo de tangente máxima, primeramente la señal de pulso es procesada con un filtro FIR de 51 coeficientes y frecuencia de corte de 30 Hz (ver figura 3.35 para detalles de respuesta en frecuencia del filtro utilizado). Este filtrado es particularmente necesario para disminuir la presencia de interferencias provenientes del convertidor analógico digital.

Luego la señal filtrada es sometida a un proceso de diferenciación ( $y[n] = x[n+1] + x[n]$ ). Dicho proceso resulta de utilidad porque provee información sobre la tasa de cambio que presenta la señal a lo largo del tiempo. En consecuencia, los puntos máximos de la señal derivada detallan los puntos donde la señal aumenta su amplitud más bruscamente, por lo que son utilizados como referencia.



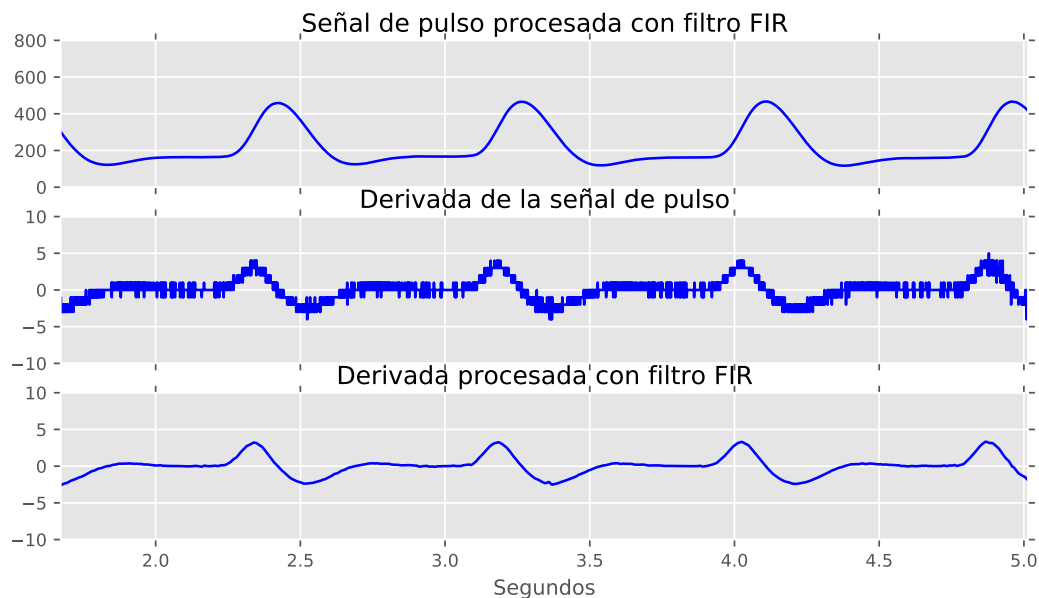
**Figura 3.36. Identificación de puntos máximos**



**Figura 3.37. Identificación de puntos con algoritmo de la primera derivada o tangente.**

Un punto importante en la implementación de este método es su alta sensibilidad a la presencia de ruido. Aun cuando el ruido o interferencia es miles de veces menor en amplitud que la señal de pulso, su influencia aumenta luego del cálculo de la primera derivada. Esto es debido a la operación de derivación, pues cada derivada es el resultado de subtracciones entre muestras adyacentes. Así, cada punto representa la diferencia entre una muestra actual y la siguiente, de manera que en caso de existir una muestra ruidosa, ésta incrementaría significativamente su influencia luego de la diferenciación de la señal.

En base a lo anterior, puede observarse la figura 3.38 donde se detalla una cantidad considerable de ruido producto de las interferencias mencionadas, por lo que se procesa con un filtro FIR (de las mismas características que el anterior) la señal en cuestión.



**Figura 3.38. Derivación de la señal de pulso**

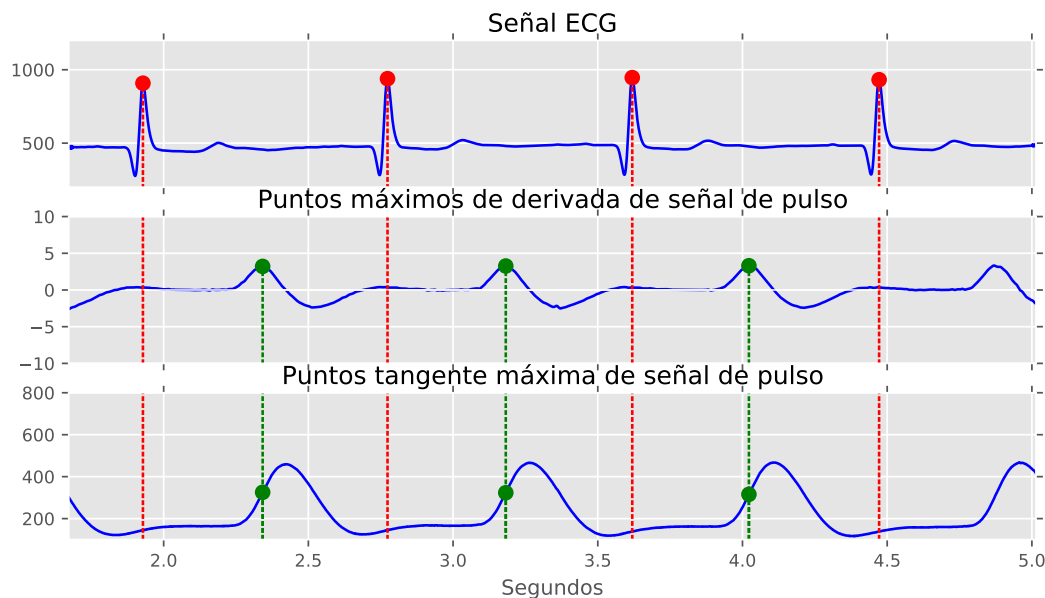
La identificación de los puntos máximos para la señal de pulso derivada se realizó con el mismo algoritmo mencionado en la sección 3.13.1.5.1. De esta manera se obtienen las posiciones correspondientes a los puntos con la pendiente máxima de cada periodo de la señal. En la figura 3.39 pueden detallarse los puntos mencionados.

Finalmente, en la figura 3.40 pueden observarse las tres señales adquiridas durante el proceso de adquisición de datos y sus respectivos puntos de medición identificados con el presente método.

**3.13.1.5.3 Algoritmo de segunda derivada máxima** El método de la segunda derivada máxima está fundamentado en la consideración que el punto mínimo o pie de la señal es el punto máximo de su segunda derivada. En la figura 3.41 pueden detallarse las diferentes etapas de procesamiento necesarias para la identificación de dichos puntos.

El procedimiento inicial es muy similar al detallado en el método de la tangente, sólo que la señal de pulso derivada es procesada nuevamente con la operación de diferenciación. La segunda derivada de la señal de pulso es sometida posteriormente a un filtro FIR pasa bajas de 51 coeficientes y frecuencia de corte 30 Hz (ver figura 3.35 para detalles de respuesta en frecuencia del filtro utilizado y la tabla 3.2 para detalles de los coeficientes), para reducir la alta presencia de ruido. Evidentemente, este método es aún más susceptible





**Figura 3.39. Identificación de puntos con tangente máxima**

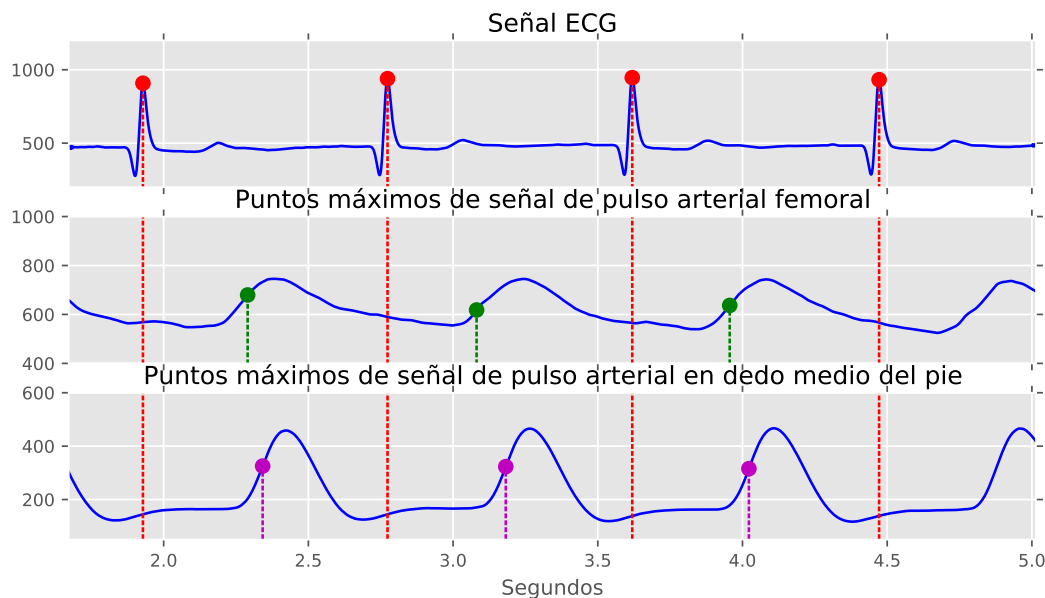
a las interferencias que el de la tangente máxima porque involucra un doble proceso de diferenciación.

En la figura 3.42 se observa el procedimiento en cuestión.

Posterior al cálculo de la segunda derivada de la señal de pulso, se procede a la determinación de sus puntos máximos de manera similar a los métodos anteriormente detallados; mediante la demarcación de los periodos de la señal a partir de los picos R de la onda ECG y la búsqueda de puntos máximos en cada uno de los tramos. No obstante, en el caso particular del presente algoritmo fue necesario un procedimiento adicional en la demarcación de tramos de búsqueda.

En algunas circunstancias la señal de pulso presenta ondulaciones justo al final del ciclo cardíaco. Estas oscilaciones afectan gravemente el rendimiento del algoritmo porque además de tener una morfología similar al pie de la señal <sup>3</sup>, usualmente se localizan antes del pico R del siguiente ciclo, de manera que dentro de determinados intervalos y luego haberse realizado dos procesos de diferenciación, pueden haber dos picos que representan posibles ubicaciones del pie de la señal, pero uno de ellos se ubica en el punto más lejano desde donde se espera el pie de la señal verdadero (se espera que el punto mínimo se ubique

<sup>3</sup>El pie de la señal es el punto ubicado antes del inicio de la pendiente abrupta de la primera onda de cada periodo en la señal de pulso. Cabe destacar que en realidad cuando se menciona la búsqueda del punto mínimo de la señal, lo que realmente se tiene como objetivo es la búsqueda del pie de ésta.



**Figura 3.40. Puntos identificados con tangente máxima en señales de pulso**

en los inicios del periodo).

La solución a este problema consistió en limitar los tramos provistos por los picos R a la mitad. De esta manera, si existe una oscilación en el final del periodo actual de búsqueda, no se tomará en cuenta porque dicho periodo se ubica al final del ciclo cardíaco y no al principio. Ver figura 3.43.

Finalmente, siguiendo el procedimiento detallado en la señal de pulso restante, se obtiene la totalidad de los puntos de medición, ver figura 3.44.

### 3.13.2 *Front-end* (Interfaz gráfica de usuario)

La interfaz gráfica de usuario fue desarrollada con Qt. Qt es una librería multiplataforma usada para la creación de aplicaciones que facilita y provee amplias funcionalidades para la construcción de interfaces gráficas. Adicionalmente, Qt es código de licencia libre y presenta buena documentación para su uso. Es una librería codificada en C++ en su forma nativa pero que puede ser utilizada en otros lenguajes de programación como Python.

Dentro de las adaptaciones a Python o *bindings* existen varias versiones, la más reciente corresponde al módulo PyQt5, pero en el presente trabajo se ha utilizado PyQt4 por ser una versión con mayor documentación.

El desarrollo de la interfaz gráfica será detallado en las siguientes secciones en base a

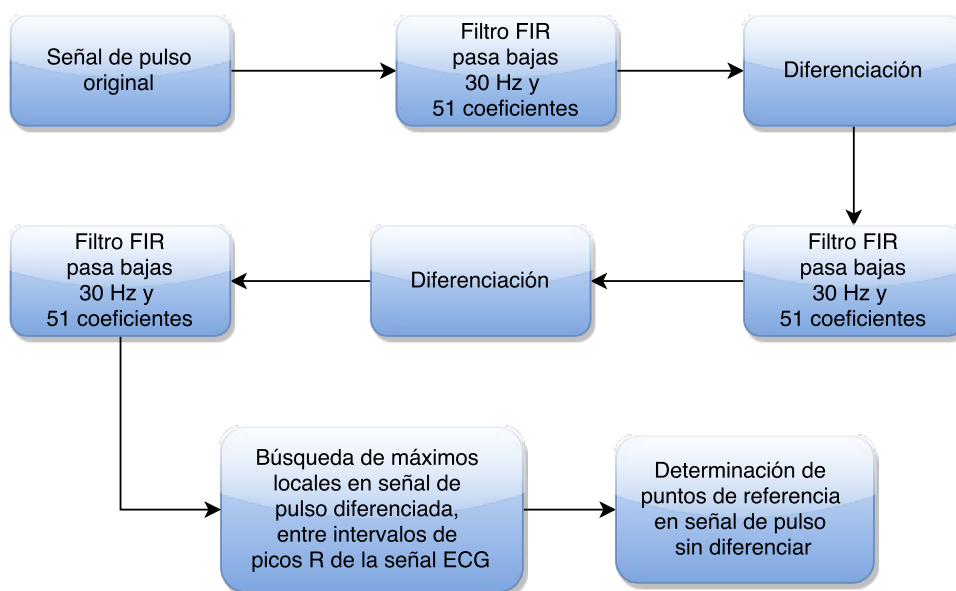


Figura 3.41. Identificación de puntos con algoritmo de segunda derivada

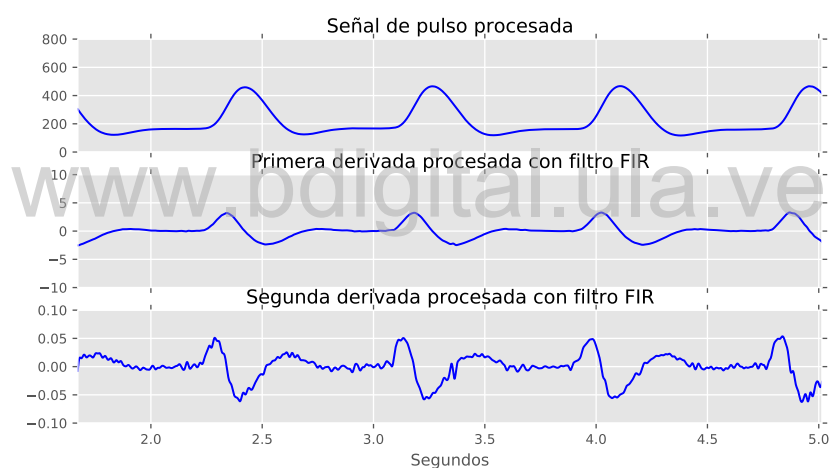


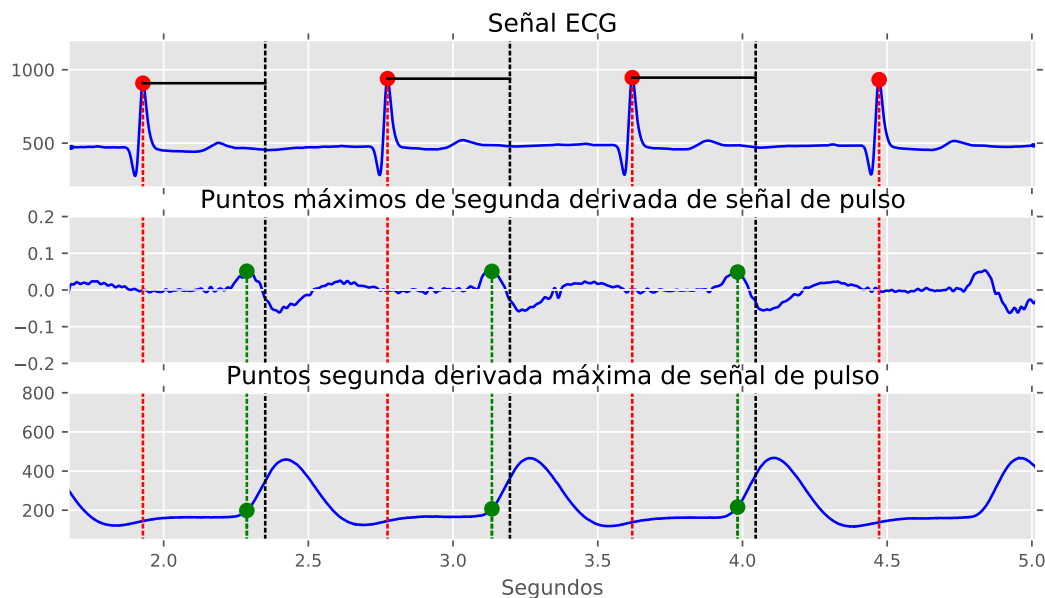
Figura 3.42. Segunda derivada de señal de pulso

las ventanas diseñadas con sus respectivas funcionalidades.

### 3.13.2.1 Ventana de solicitud de dirección de base de datos

La interfaz gráfica está constituida principalmente por una ventana principal que contiene varios *widgets* (con distintas funcionalidades) y por ventanas emergentes o diálogos para eventos puntuales. Una de las ventanas mencionadas es la detallada en la presente sección.

La ventana de solicitud de dirección de base de datos hereda de la clase de QT *Qt-Gui.QDialog()* y fue configurada para requerir el ingreso de la dirección de memoria o *path*



**Figura 3.43. Identificación de puntos mínimos**

de la base de datos, donde se leerán y almacenarán los archivos de medición del paciente. El *path* es requerido para construir o manejar la estructura de archivos detallada en la sección 3.13.1.3.

El diálogo de selección de base de datos es el inicial del programa y se presenta como una ventana emergente. El usuario, posterior a la especificación del *path*, puede cancelar el proceso (y cerrar el programa) o continuar presionando el botón aceptar.

En la figura 3.45 puede observarse la apariencia de la ventana a colación.

### 3.13.2.2 Ventana principal

La ventana principal del programa es ejecutada luego de que el usuario determina la dirección de memoria de la base de datos en la ventana inicial y presiona el botón de aceptar. En el entorno Qt para Python las ventanas principales heredan de la clase *QtGui.QMainWindow()*. Las ventanas de la clase *QtGui.QMainWindow()* constan de varios *widgets*, entre ellos:

- Barra de menú: localizada en la parte superior de la ventana y donde se instancian las pestañas de la ventana principal.
- Barra de herramientas: ubicada en la parte superior (o bien en las regiones laterales)

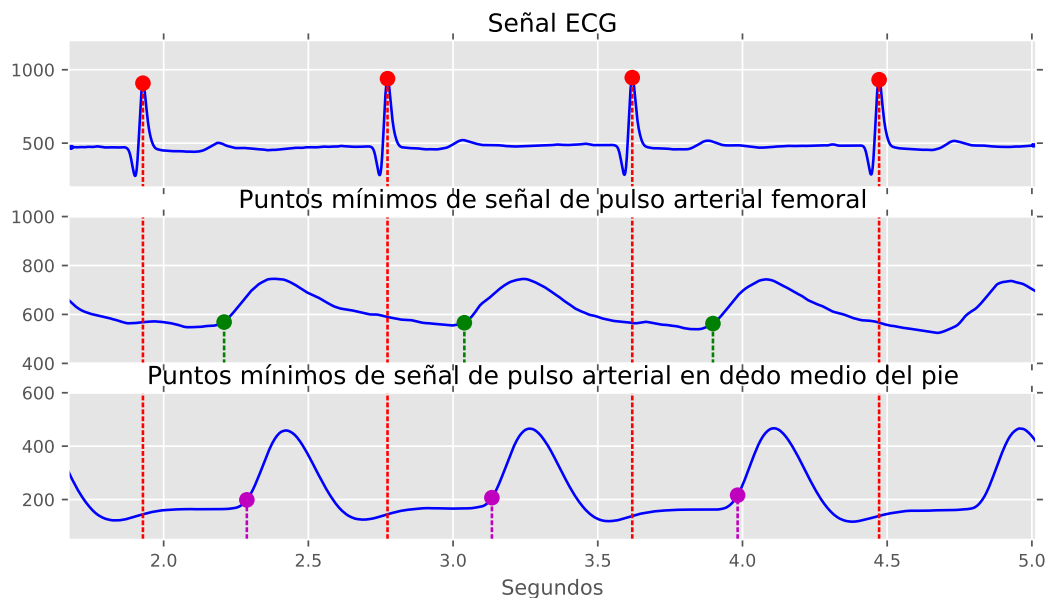


Figura 3.44. Puntos identificados con segunda derivada máxima en señales de pulso

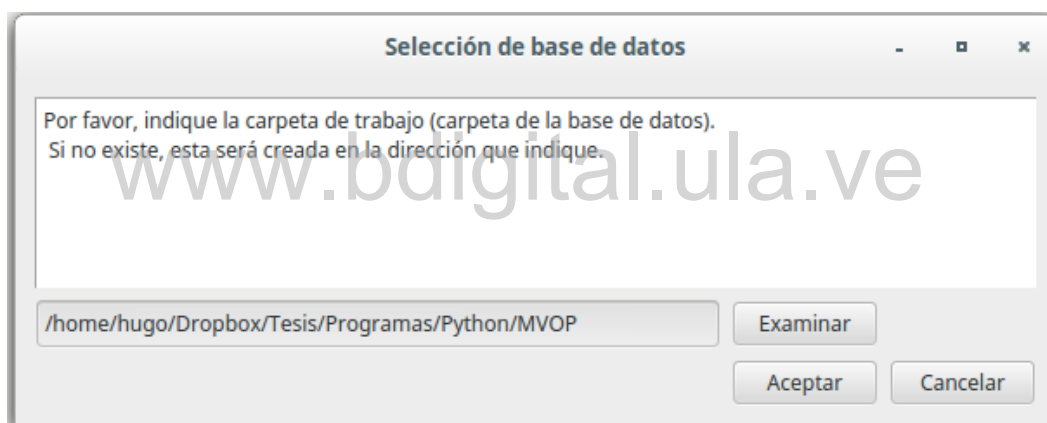














Figura 3.45. Ventana de base de datos

de la ventana donde se muestran diferentes botones con funcionalidades específicas para la aplicación.

- Barra de estado: localizada en la parte inferior de la ventana donde se detalla información sobre los estados del programa.
- *Widget* central: ubicado en la zona central de la ventana donde se muestran los elementos que interactúan con las acciones de la barra de menú y de herramientas.

En la figura 3.46 pueden observarse los *widgets* mencionados en la ventana principal.

En la aplicación desarrollada, la lista de botones correspondientes a la barra de herramientas, observados en la figura 3.46 de izquierda a derecha, son los siguientes:

- Nuevo : inicia el *widget* de formulario, para la introducción de los datos del paciente.
- Registro : inicia el *widget* de registro de pacientes. En él se visualiza una tabla con los diferentes pacientes a los que se le han realizado mediciones de VOP.
- Guardar : ejecuta rutinas para el almacenamiento de la información luego del muestreo y procesamiento de las señales.
- Anterior : regresa al *widget* anterior.
- Siguiente : inicia el *widget* siguiente de acuerdo al procedimiento ejecutado.
- Muestrear : inicia el proceso de muestreo de las señales de pulso y ECG.
- Parar : detiene el proceso de adquisición de señales en ejecución.
- Calibrar : crea *widgets* para el ajuste de potencia de los sensores, así como habilita los botones de Ajustar y Listo.
- Ajustar : modifica el parámetro de potencia para la luminosidad de los LEDs en los sensores, de acuerdo al *widget* de ajuste de potencia de LEDs correspondiente.
- Listo : indica la finalización del proceso de calibración.
- Calcular : realiza el proceso de identificación de puntos de medición en las señales y el cálculo de la VOP.
- Ayuda : provee información acerca de la calibración de los sensores.

### 3.13.2.3 Ventana de formulario

En base al procedimiento para realizar la medición de VOP, el *widget* central inicial corresponde a un formulario, donde el usuario deberá rellenar los campos referentes a la información del paciente, ver figura 3.47.

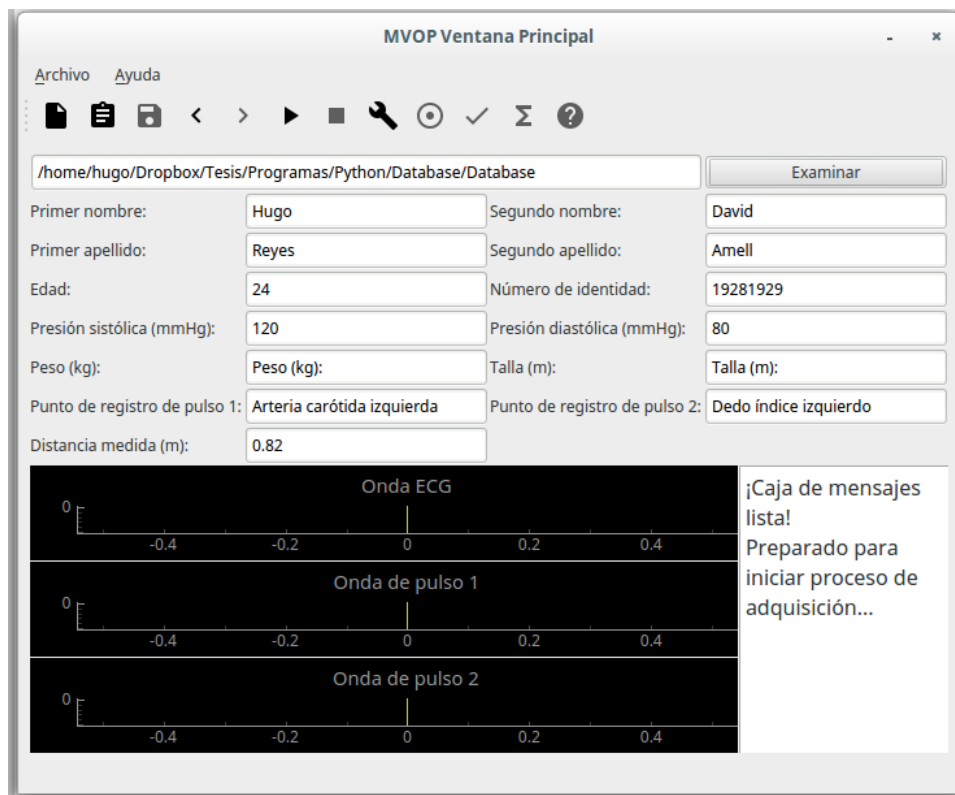


Figura 3.46. Ventana principal

El *widget* formulario hereda de la clase `QtGui.QWidget()` del entorno Qt. Adicionalmente, incluye procesos de verificación de contenido en caso que el usuario ingrese caracteres especiales o números en campos de letras o viceversa. La verificación se realizó con el módulo de expresiones regulares (*re*) de Python.

En la figura 3.48 se muestran los siguientes ejemplos de verificación: en caso de que el usuario deje uno de los campos necesarios en blanco, se resalta el borde en amarillo. Si por otra parte, el usuario ingresa incorrectamente la información requerida en un campo (caracteres no permitidos) el borde se tornará rojo y se emitirá un mensaje de error en el cuadro de texto. Por último, si el usuario ingresa de forma correcta los datos demandados aparecerá el color verde indicando una validación positiva de la información provista.

Posterior al ingreso de la información demandada, el usuario ha de presionar el botón de Siguiente  $\blacktriangleright$ , el cual en el presente contexto tiene como función el cierre del *widget* actual y la apertura del panel de adquisición de datos.

Figura 3.47. Ventana principal, formulario

#### 3.13.2.4 Panel de adquisición de datos

El panel de adquisición de datos es el *widget* principal de la aplicación (hereda de la clase de Qt `QtGui.QWidget()`). En él se ejecutan las acciones de muestreo, calibración y cálculo de la VOP. El panel de control está constituido por campos de texto que detallan la información introducida del paciente, una caja de texto que funje de terminal (donde se imprimen mensajes de progreso para las diferentes acciones en ejecución) y finalmente tres gráficas donde se realiza el ploteo de los datos adquiridos en *cuasi* tiempo real, ver 3.49.

La graficación de datos se realizó por medio de la librería *PyQtGraph*, la cual es considerada una extensión de PyQt (Qt para Python). Este módulo resulta óptimo para sistemas embebidos por su rapidez para el ploteo de resultados en comparación con otros módulos de graficación mucho más populares como Matplotlib.





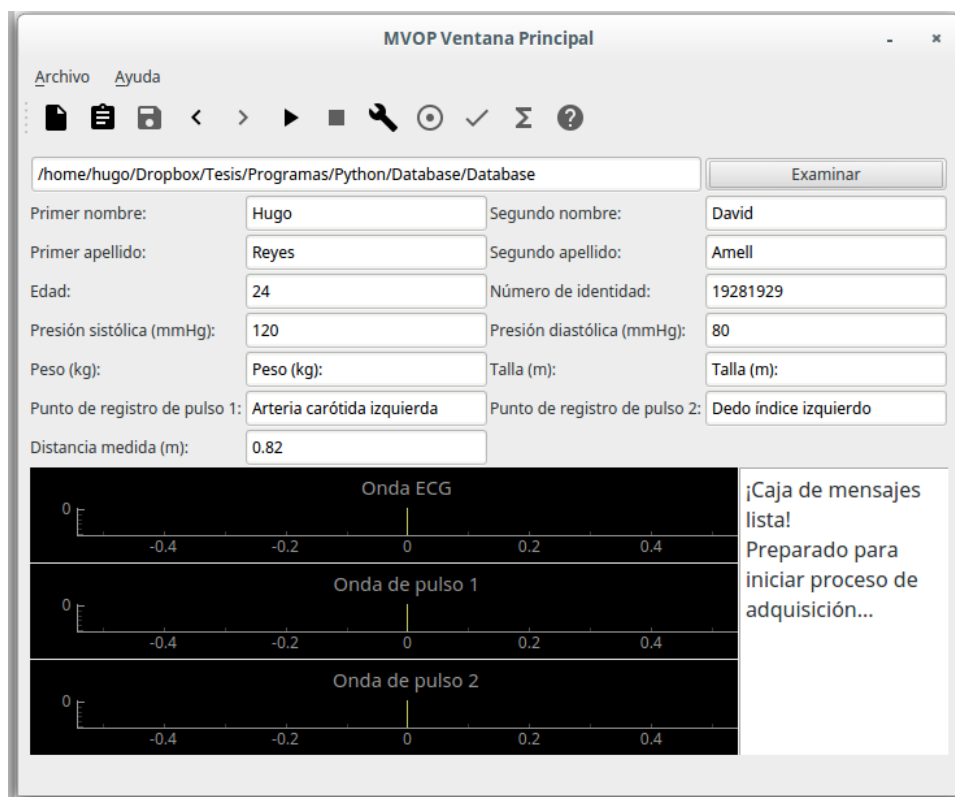
Figura 3.48. Ventana principal, validación en formulario

**3.13.2.4.1 Calibración** La calibración en el contexto de la interfaz gráfica consiste en adaptar los sensores de fotopletimografía a las condiciones particulares de cada paciente y de cada región anatómica.

Como se ha hecho referencia en el presente trabajo, la cantidad de tejido en la zona de registro influye directamente en la calidad y potencia de la señal a obtener. Esta característica hace necesario el ajuste de los sensores en manera de incrementar o disminuir la cantidad de luz que incide sobre el tejido para mejorar la calidad de la señal de pulso correspondiente.


Cuando el usuario presiona el botón *Calibrar* , se realizan las siguientes acciones:

- Se realiza el muestreo de las señales hasta un total de 5000 muestras por cada señal de pulso. Esto con la finalidad de que el usuario pueda apreciar si es requerido un aumento o disminución en la intensidad de luz emitida por los LEDs en los sensores.
- Se habilitan los botones de Ajustar  y Listo .

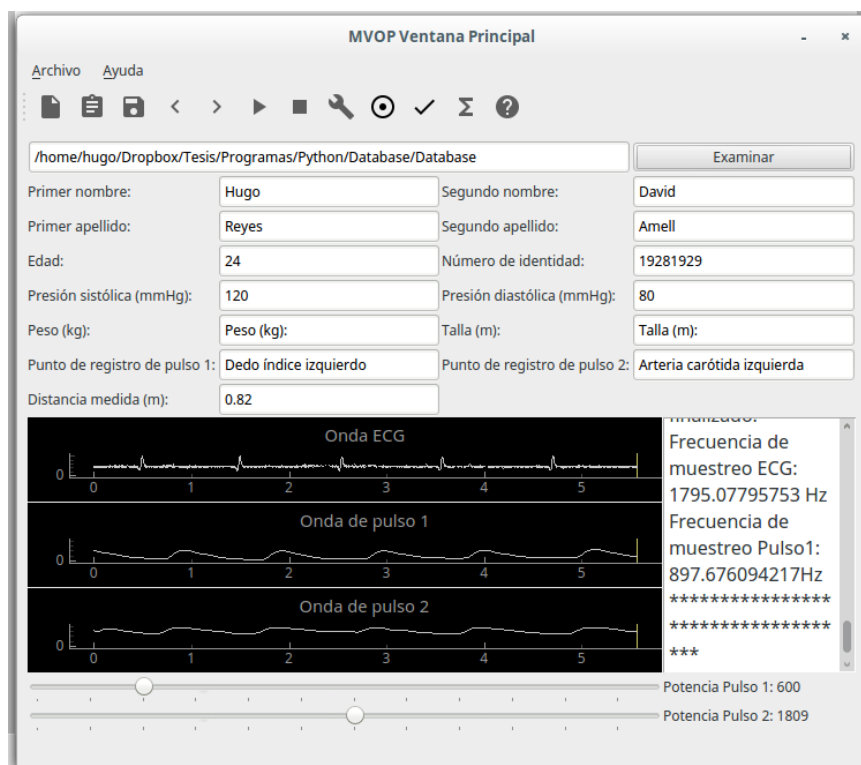


**Figura 3.49. Ventana principal, panel de adquisición de datos**

- Se crean *widjets* con botones deslizantes para habilitar los cambios de luminosidad mencionados, ver figura 3.50.

Cuando se efectúa un deslizamiento en uno de los dos botones deslizantes y se presiona el botón Ajustar , los nuevos valores de configuración generados son enviados vía puerto serial al circuito de adquisición de datos. Posterior a ello, se realiza nuevamente el proceso de muestreo de 5000 muestras con el objetivo de evidenciar el efecto de dichos cambios. Cabe destacar que en algunos casos, debido a la configuración y a los tipos de filtros analógicos implementados, las señales presentan tramos importantes de saturación hasta luego de aproximadamente 15 segundos. Es luego de un tiempo determinado cuando los filtros ofrecen los nuevos resultados de forma claramente apreciable.

Es importante detallar que la regla general en cuanto al ajuste de amplitudes con la luminosidad de los LEDs es que entre mayor es la potencia en éstos, mayor es la amplitud de la señal. En otras palabras: un movimiento hacia la derecha de los botones deslizantes deberá producir un incremento en la amplitud de la señal y uno hacia la izquierda su disminución. No obstante, ha de recalarse que en algunos casos los cambios mencionados



**Figura 3.50. Ventana principal, calibración**

no son suficientes para producir incrementos apreciables en la señales, ante ello es necesario efectuar una redistribución de los sensores en el paciente con el fin de optimizar el registro del pulso.

Posterior al proceso de calibración mencionado y cuando el usuario pueda percibir de forma clara las señales de pulso, se deberá presionar el botón *Listo* ✓ para eliminar los *widjets* de botón deslizante y continuar el proceso para la medición de VOP.

**3.13.2.4.2 Muestreo** El muestreo principal es el ejecutado cuando se presiona el botón *Muestrear* ►. Si bien en la calibración se efectúa la adquisición de señales, ésta no se realiza con una gran cantidad de muestras (5000 por cada señal de pulso). Es por tanto, que en el muestreo principal se realiza la adquisición de 20000 muestras por señal de pulso y 40000 para la señal electrocardiográfica.

Durante la ejecución de los procesos de la subrutina de muestreo (detallados en la sección 3.13.1.2), las señales mencionadas son ploteadas en paquetes de 100 muestras cada una emulando un ploteo en tiempo real. Al término del proceso de adquisición, se reciben variables de temporización que son utilizadas para la estimación de los tiempos en los que



Figura 3.51. Ventana principal, muestreo

han sido obtenidas las muestras.

Al finalizar las acciones mencionadas, se realiza nuevamente la graficación de las señales pero incluyendo los tiempos correspondientes. El eje horizontal detalla entonces la información temporal (segundos) de las señales adquiridas, ver figura 3.51.


Por último, debido a que ya se tiene toda la información requerida, se procede a habilitar el botón *Calcular*  $\Sigma$ , para el procesamiento de las señales y cálculo de la VOP.

### 3.13.2.5 Cálculo de la VOP

Esta sección de la aplicación será detallada en el capítulo 4 debido a su dependencia con la selección del algoritmo de identificación. La cual es realizada en base a los resultados dados en dicho capítulo. Ir a la sección 4.5 para mayor información.

### 3.13.2.6 Almacenamiento de archivos

Posterior al procesamiento y cálculo de la VOP el usuario deberá almacenar los datos obtenidos en la dirección de la base de datos. Esto por supuesto, habilitará posibles consultas y creará el registro del paciente en dicha base de datos (en caso de que no exista).

Para iniciar el proceso referido es necesario presionar el botón *Guardar* , con lo que se generará una ventana emergente (que hereda de la clase de Qt `QtGui.QDialog()`) que requerirá la información referente al diagnóstico por parte del médico. Dentro de dicha ventana, el usuario podrá especificar si es requerida la creación de una nueva carpeta de consulta o utilizar una existente (si ésta efectivamente existe, en caso de no existir, se crea una nueva).

El proceso de guardado no inicia hasta que el usuario presione aceptar. Ver figura 3.52.

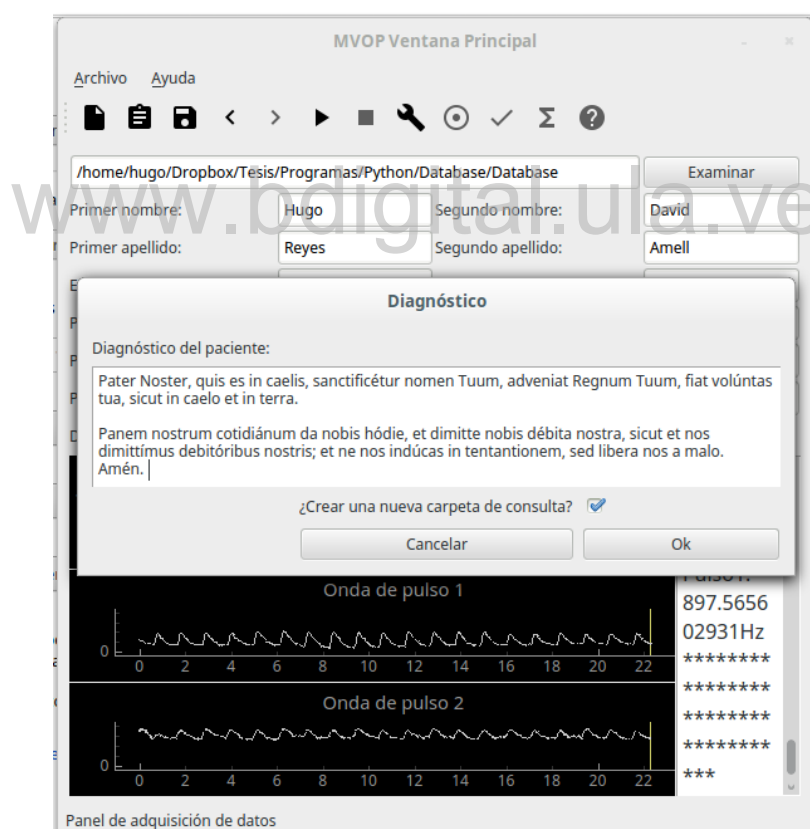
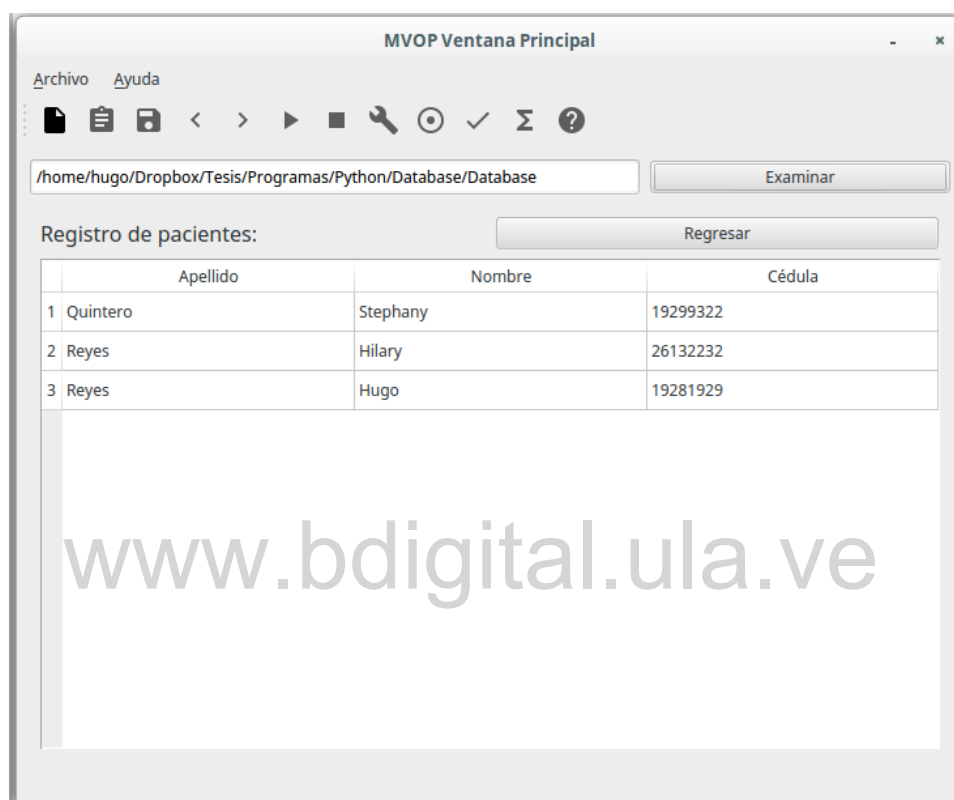


Figura 3.52. Ventana principal, almacenamiento de datos

### 3.13.2.7 Registro de pacientes

Este *widget* realiza la lectura de los archivos de pacientes contenidos en la base de datos. Para su implementación se utilizó la clase de Qt *QtGui.QTableWidget()* con el fin de organizar la lista de pacientes en cuanto a su apellido, nombre y número de cédula. Ver figura 3.53.

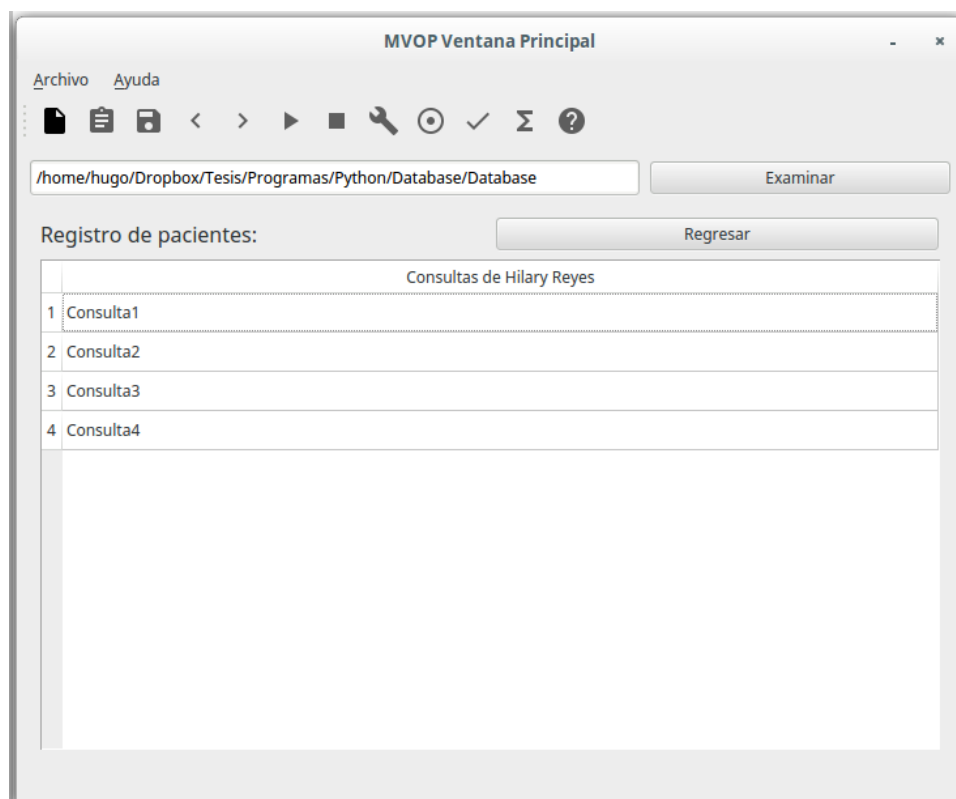


**Figura 3.53. Ventana principal, registro de pacientes**

Adicionalmente, si el usuario realiza un doble *click* sobre un determinado paciente, se reescribirá la tabla de acuerdo al número de consultas (o sesiones de medición) que han sido realizadas a dicho paciente, ver figura 3.54.

De manera sucesiva, al usuario realizar la misma acción sobre una de las consultas, aparecerán los reportes de cada una de las mediciones registradas en la consulta correspondiente, ver figura 3.55.

El usuario podrá visualizar los reportes de la lista realizando un doble *click* sobre cada uno. En cada reporte se detalla la VOP, las dos señales de pulso y la onda ECG, ver figura 3.56. Así como también se detalla el diagnóstico clínico en el campo de texto, hacia la derecha de los *widgets* de graficación.



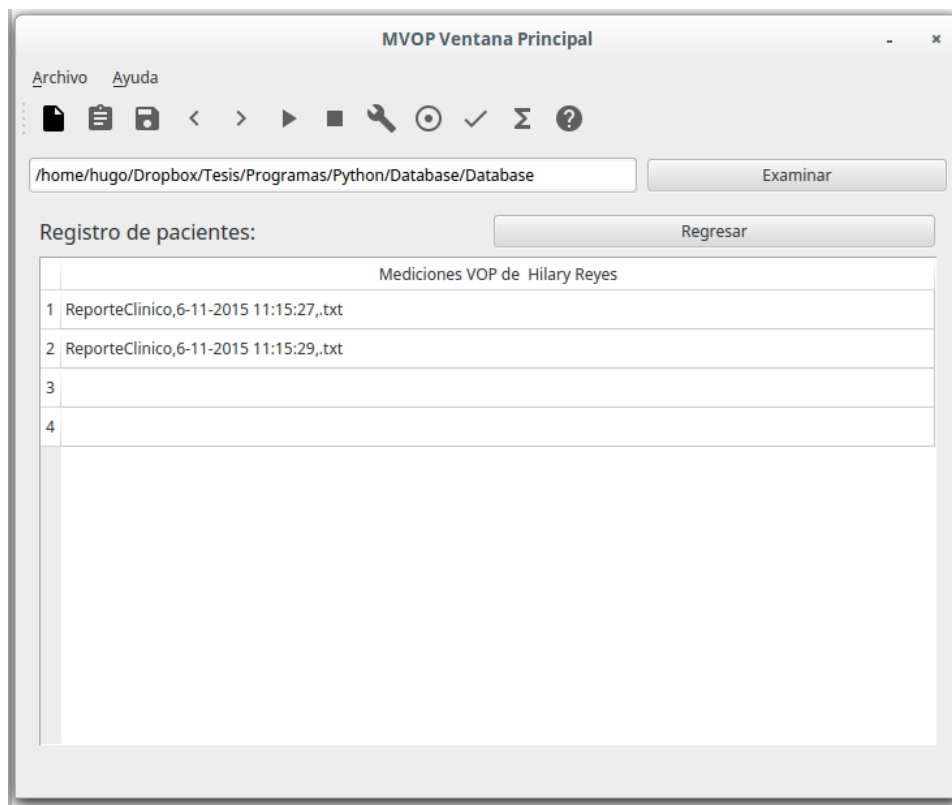
**Figura 3.54. Ventana principal, registro de consultas**

En la parte inferior de la ventana de reporte, pueden observarse también los botones *Salir* y *PDF*. Intuitivamente, el botón *Salir* cierra la ventana presente conservando la ventana principal. Por otra parte, el botón *PDF* construye una nueva ventana emergente para la introducción del nombre y dirección de memoria del archivo PDF a generar.

El archivo PDF incluye toda la información detallada en la ventana de reporte con la diferencia que ésta es organizada en dos páginas de tamaño tipo carta. En las figuras 3.57 y 3.58 se puede observar un documento generado en manera de ejemplo.

Para la construcción del archivo PDF se utilizaron tres herramientas separadas para el manejo de estos archivos. El archivo generado por el código desarrollado consta de dos páginas. La primera página es construida con ayuda del módulo *reportlab*, el cual no es interno de Python pero es totalmente descargable e utilizable debido a que es software libre. Con este módulo es procesada la información del paciente, el diagnóstico por parte del médico y una imagen de logo.

La segunda página contiene las gráficas de las señales adquiridas que son generadas por medio del *backend* de Matplotlib *PDFPages*. Esta herramienta permite producir archivos



**Figura 3.55. Ventana principal, registro de reportes**

PDF con las gráficas creadas por la librería.

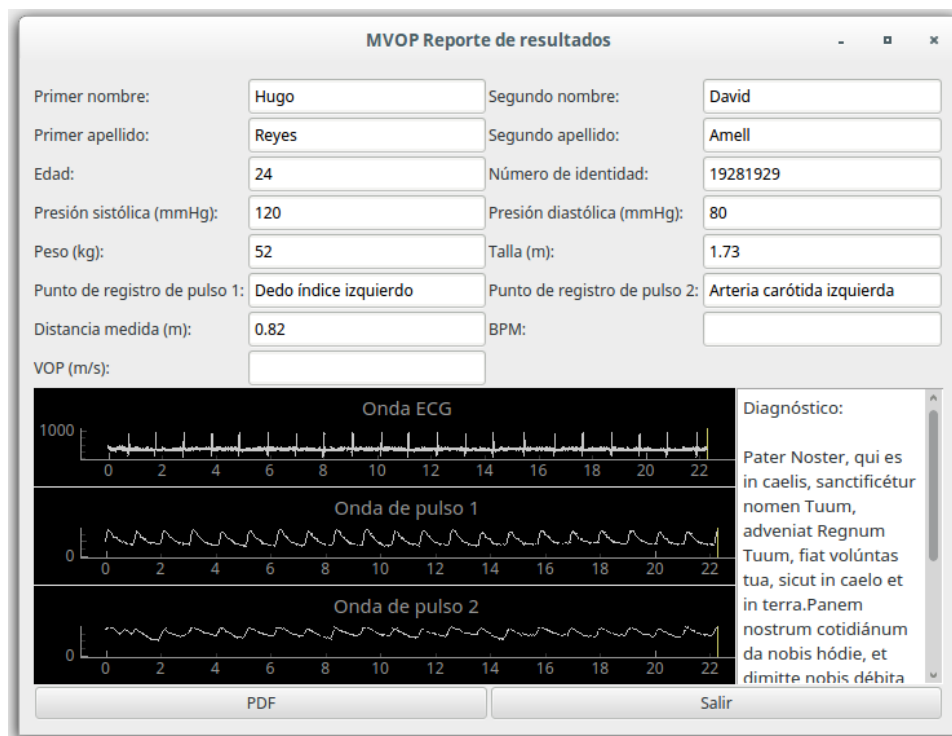
Finalmente con el uso de las librerías mencionadas se generan dos archivos PDF de una página cada uno. Para unir las dos páginas en un solo documento es necesario un tercer módulo: PyPDF2. PyPDF2 es una librería software libre de terceros que ofrece la opción de *merge* o unión de archivos para formato PDF.

### **3.14 Evaluación de los algoritmos de identificación de puntos de medición**

Los algoritmos detallados en la sección 3.13.1.5 fueron evaluados desde el punto de vista de su precisión. Es conocido que la precisión y el nivel de dispersión de los datos resultantes conlleva una relación directa. Por esta razón la evaluación se realizó considerando la desviación estándar de los resultados obtenidos.

Las mediciones efectuadas se realizaron en 3 pacientes bajo condiciones reproducibles. A los pacientes se les ordenó disponerse en posición supina y los puntos de adquisición de





**Figura 3.56. Ventana de reporte**

pulso fueron el dedo medio del pie derecho y la arteria femoral (en la ingles en la zona del triángulo de Scarpa). Cabe destacar que a cada paciente se le realizaron 10 mediciones en tiempos muy cortos entre sí, lo que conlleva a esperar que los resultados guarden un grado de similitud considerable. El tamaño total del *dataset* bajo estudio involucró un total de 30 mediciones.

En cada medición se obtuvieron dos señales de pulso y la señal ECG durante un tiempo aproximado de 22 segundos. El número de períodos de las señales varía de acuerdo a la frecuencia cardíaca del paciente.

A cada una de las mediciones obtenidas se les procesó con los distintos algoritmos detallados en la sección 3.13.1.5. Los algoritmos mencionados realizan la identificación de los puntos de medición e incluyen una rutina para la determinación del tiempo de tránsito o el desfase entre las señales. La velocidad de pulso se determina con el tiempo de tránsito entre señales y la distancia medida entre los puntos de medición.

El resultado final de cada uno de los algoritmos corresponde a un vector con la velocidad de pulso en cada periodo de la señales procesadas. Consecuentemente se obtuvieron 30 vectores con las VOP en cada periodo de las señales.

-----Archivo de reporte de VOP-----



**UNIVERSIDAD  
DE LOS ANDES**  
MERIDA-VENEZUELA

---

Fecha y hora: 11-11-2015 0:7:34

-----Datos del paciente-----

Nombre: Hugo David Reyes Amell  
 Cedula: 19281929  
 Edad: 24  
 Peso: 52 kg  
 Talla: 1.73 m  
 Presión sistólica: 120 mmHg  
 Presión diastólica: 80 mmHg  
 Primer punto de registro de pulso: Dedo índice izquierdo  
 Segundo punto de registro de pulso: Arteria carótida izquierda  
 Distancia: 0.82 m

-----Resultados-----

Velocidad de onda de pulso: 0 m/s  
 Ritmo cardíaco: 0 bpm

-----Diagnóstico clínico-----

Esto es un simulacro de diagnóstico. Es evidente que pueden ser utilizados signos de puntuación, acentos y la letra ñ perteneciente al castellano. Un ejemplo en manera de demostración es la frase siguiente: ÁEÍOU áeíou ñÑ ñ ñ ñ.

---

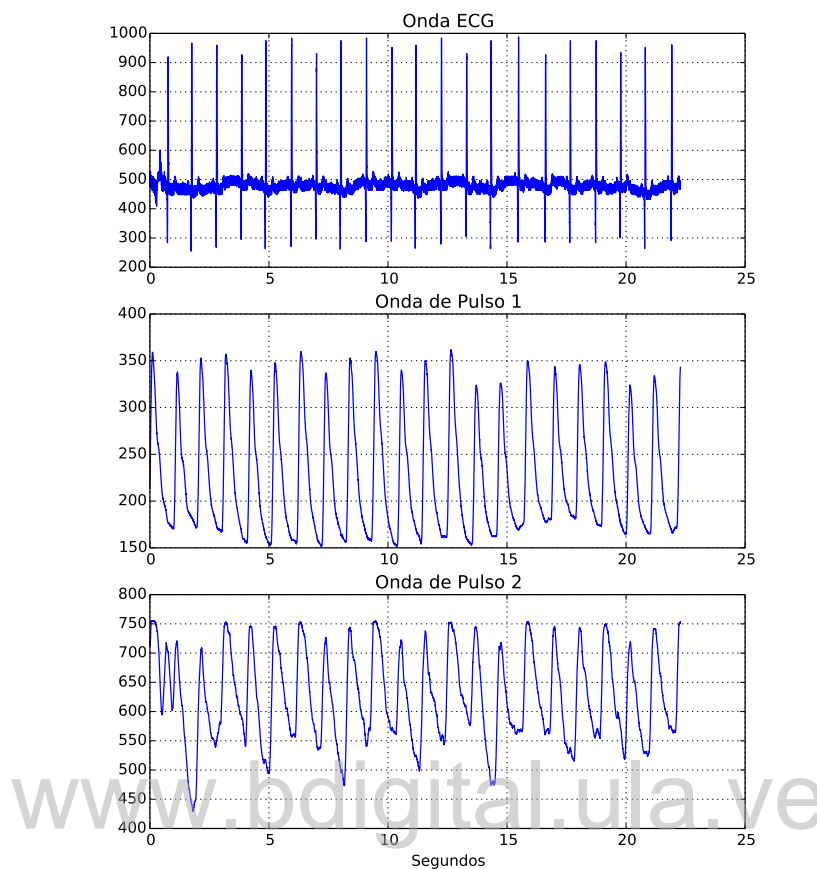
www.bdigital.ula.ve

**Figura 3.57. Documento PDF, primera página**

### 3.14.1 Comparación con resultados manuales

Del total de vectores obtenidos (para cada una de las mediciones de los 3 pacientes) se seleccionó un *set* de un paciente de manera aleatoria para el cálculo manual de la VOP, con la finalidad de establecer una comparación entre los resultados manuales y los provistos por los algoritmos.

Para ello, a las señales de pulso cardíaco se les graficó en un computador personal y se les realizó la identificación de puntos máximos, mínimos y tangente máxima de forma manual (visual) para finalmente calcular las velocidades. Los puntos fueron identificados por el investigador principal del presente trabajo, aunque no obstante, los resultados fueron posteriormente verificados por profesionales de la salud. La comparación entre los resultados manuales y los obtenidos por los algoritmos se llevó a cabo de forma gráfica y también



**Figura 3.58. Documento PDF, segunda página**

estadísticamente.

El análisis gráfico consistió en la comparación visual de los resultados de velocidad para las distintas mediciones y algoritmos. Por otra parte, el análisis estadístico consistió en realizar una substracción entre los resultados de cada algoritmo con sus correspondientes resultados manuales. El vector resultante fue elevado al cuadrado y se le calculó el promedio.

El valor obtenido con el procedimiento mencionado revela un estimado de la similitud que tienen los distintos resultados de los algoritmos con sus pares manuales. Entre menor es promedio del vector de diferencia cuadrático, mayor es la cercanía de los resultados.

### 3.14.2 Análisis de similitud de resultados de VOP por periodos

Adicionalmente a la comparación entre los resultados manuales y los computados, se evaluó también el nivel de similitud entre las distintas mediciones para cada uno de los algoritmos.

La cuantificación de la similitud se realizó efectuando restas entre dos resultados de determinado método y calculando la dispersión. En la ecuación 3.7 se detalla el procedimiento de substracción, donde  $x_1$  y  $x_2$  corresponden a dos mediciones respectivamente y  $d$  al vector de diferencias resultante.

$$d(i) = x_2(i) - x_1(i) \quad (3.7)$$

Obteniendo  $d$ , puede determinarse el grado de dispersión de dicho vector por medio de la desviación estándar  $\sigma$ , ver ecuación 3.8.

$$\sigma = \sqrt{\frac{1}{N} \sum_{i=0}^{N-1} (d(i) - \bar{d})^2} \quad (3.8)$$

El procedimiento detallado es válido para la evaluación de similitud de datos entre sólo dos mediciones. Sin embargo, cuando se dispone de un mayor número de mediciones, una opción es realizar el mismo procedimiento descrito con todas las combinaciones posibles de pares distintos. El número de pares distintos  $M$  para el número de vectores de medición disponibles  $N$  esta dado por la ecuación 3.9 [15].

$$M = C_N^2 = \frac{N!}{2!(N-2)!} \quad (3.9)$$

Del procedimiento mencionado se obtienen todas las desviaciones para cada uno de los  $M$  pares. Finalmente se calcula el promedio dado por la ecuación 3.10. La desviación resultante provee una medida de similitud en los resultados de determinado método bajo estudio.

$$\bar{\sigma} = \frac{1}{N} \sum_{i=0}^{N-1} \sigma(i) \quad (3.10)$$

### 3.14.3 Análisis de dispersión de VOP promedio

En la sección anterior, el procedimiento del cálculo de la desviación de los datos se realizó considerando cada uno de los valores de velocidad calculados en los periodos de las señales. Bajo esa metodología se obtienen valores de dispersión más ajustados a un comportamiento puntual (por cada pulso) del dispositivo de medición.

Sin embargo, en el proceso de medición de VOP con el equipo en funcionamiento normal (una medición típica del paciente por parte del médico) lo que se obtiene es un promedio de las velocidades de pulso de cada periodo. La operación de promedio oculta la variabilidad presente entre las velocidades de los distintos periodos y resulta obviamente un resultado mucho menos variable y una estimación más representativa del valor esperado durante el uso normal del equipo.

En base a lo expuesto se hace necesaria la determinación del grado de dispersión de los resultados ante dichas estimaciones. El resultado de este experimento adicionalmente puede proveer al usuario el nivel de desviación posible en los resultados de VOP con el equipo desarrollado.

La determinación del grado de dispersión simplemente se efectuó calculando la desviación estándar de los promedios de VOP de cada una de las mediciones. Los resultados de este procedimiento se detallan en la sección 4.3.

### 3.14.4 Eliminación de valores atípicos

En conjunto con los procesos de evaluación detallados, se ha implementado una rutina de validación con la finalidad de descartar valores atípicos de velocidad como consecuencia de errores de medición y/o interferencias en las señales. Estos valores atípicos o también llamados *outliers* afectan gravemente los resultados promedios de velocidad de pulso, además de incrementar enormemente los niveles de varianza y desviación estándar utilizados como referencia para la evaluación de los algoritmos.

Primeramente para la eliminación de los valores atípicos es necesario un proceso de identificación. Existen diferentes formas de detectar *outliers* dentro de un conjunto de datos pero en gran parte dependen de dos aspectos: las características del proceso o el tipo de datos bajo evaluación (no es lo mismo evaluar una lista de precios que datos de variables

fisiológicas) y la distribución de los datos. Si los datos, aunque con ciertos componentes aleatorios se asemejan a una distribución *gaussiana*, puede resultar conveniente establecer un umbral de 3 veces la desviación estándar del conjunto de datos ( $3\sigma$ ) para descartar datos muy alejados de la media.

No obstante, para conjuntos de datos cuya distribución de probabilidad presenta colas pesadas, el parámetro de localización basado en el promedio puede verse altamente afectado y distorsionar en consecuencia los límites de detección de valores atípicos. Esto hace que se conserven más datos que pueden ser componentes ruidosos del conjunto de datos, y que perjudican la variabilidad de los datos (la cual debe ser lo más baja posible).

Tal es el caso de los datos obtenidos con los diferentes algoritmos, en los cuales se presentan valores exageradamente alejados de la media muestral y que afectan fuertemente sobre los cálculos de la velocidad promedio, así como la varianza de los resultados. Por esta razón se utilizó el criterio de la mediana de la desviación mediana absoluta  $\lambda$  dado por la ecuación 3.11 para la asignación del umbral en la detección de outliers. Adicionalmente, se utilizó un factor de 23 multiplicado a  $\lambda$  para la determinación del umbral mencionado, es decir:  $\mu = 23\lambda$ .

$$\lambda = \text{median}_i(|x_i - \text{median}_i(x)|) \quad (3.11)$$

Adicionalmente, se implementó también una restricción de valores de velocidad de acuerdo al rango posible de valores provistos en [1], donde se establece que las velocidades de pulso comprenden un rango de 2 a 35 m/s. De esta manera se ha establecido un intervalo de tolerancia de 2 a 50 m/s, lo que implica el descarte de velocidades de pulso con valores por fuera de éste.

### 3.15 Comparación de mediciones de VOP promedio con valores de referencia

El enfoque de los procedimientos detallados en la sección 3.14 fue la selección del algoritmo con el menor grado de dispersión en sus resultados. Por otra parte, surge la necesidad de evaluar si los valores de velocidad obtenidos en diferentes mediciones corresponden a

magnitudes razonables para el cuerpo humano.

La validación requiere de valores de referencia. En la bibliografía se detallan rangos para diferentes tamaños de arterias [1], pero ello resulta insuficiente para un estudio más exhaustivo.

Afortunadamente en investigaciones antecedentes se han enlistan valores referenciales de VOP para tres poblaciones pertenecientes geográficamente a: Europa, Argentina y Uruguay respectivamente [2]. Cabe destacar que las mediciones de referencia realizadas en [2] fueron obtenidas mediante la adquisición de la señal del pulso carotídeo y el femoral. En consecuencia, para la comparación de los resultados del equipo con los valores de referencia se hizo necesaria la medición de pulso en dichos puntos anatómicos.

Resulta importante detallar que las mediciones en [2] han sido tomadas con un equipo comercial que trabaja con sensores de diferente tecnología al de los utilizados en el equipo del presente trabajo, por lo que pueden existir diferencias entre las velocidades de pulso obtenidas y las detalladas en las investigaciones antecedentes.

Por otra parte, la metodología para la comparación de resultados con valores de referencia, tiene no obstante una importante restricción para la presente aplicación: la indisponibilidad de los sensores requeridos para las mediciones simultáneas del pulso carotídeo y femoral, ésto debido a que sólo se dispone de un sensor por fotopletoisografía tipo pinza y uno de reflexión (son necesarios dos de estos últimos).

La solución a este problema consistió en realizar adquisiciones de las señales de forma separada (en tiempos lo más cercamente posible entre mediciones) aprovechando la señal ECG como valor de referencia. En consecuencia, la mediciones en los puntos Carótida-Femoral se realizaron siguiendo el siguiente procedimiento:

- Medición de la distancia entre los puntos de medición.
- Medición del pulso carotídeo y de la señal ECG.
- Medición del pulso femoral y de la señal ECG.
- Identificación de picos R en la señal ECG anexa al pulso carotídeo.
- Identificación de puntos de referencia en la señal de pulso carotídeo.

- Cálculo del tiempo de tránsito entre los puntos de la señal de pulso carotídeo y los picos R de la señal ECG.
- Identificación de picos R en la señal ECG anexa al pulso femoral.
- Identificación de puntos de referencia en la señal de pulso femoral.
- Cálculo del tiempo de tránsito entre los puntos de la señal de pulso femoral y los picos R de la señal ECG.
- Sustracción del tiempo de tránsito ECG-Pulso femoral y ECG-Pulso carotídeo.
- Cálculo de la velocidad vía  $v = d/t$ . Donde  $t$  es el tiempo de transito en diferencias ECG-Pulso femoral y ECG-Pulso carotídeo (calculado en el proceso anterior).

Con el procedimiento detallado se realizó la medición de VOP a 10 pacientes (3 mediciones por paciente) y se recopiló información referente a edad, peso, talla, género y presión arterial.

Por último, los resultados fueron analizados considerando los valores de referencia de [2] y las características de cada paciente.



## Capítulo 4

# RESULTADOS Y ANÁLISIS DE RESULTADOS

En el presente capítulo se presentan los resultados referentes al procesamiento de las mediciones de VOP realizadas bajo los procedimientos detallados en la secciones 3.14 y 3.15.

Para la evaluación de los algoritmos en cuanto al nivel de dispersión de los datos, cabe reiterar que se realizaron 10 mediciones a cada paciente (3 pacientes en total) bajo las mismas condiciones ambientales y temporales. A los pacientes se les dispuso en posición supina y se adquirieron los pulsos femorales y del dedo medio del pie. En primera instancia se efectuó la comparación de los resultados manuales y los obtenidos por medio de los algoritmos propuestos. Esta comparación se realizó de forma gráfica, y estadísticamente evaluando el promedio de vectores en diferencia entre los resultados manuales y los obtenidos por los algoritmos.

Posteriormente se buscó determinar el grado de similitud entre las mediciones realizadas a los pacientes en base al procedimiento detallado en 3.14. El procedimiento consistió en agrupar las mediciones de cada paciente y unir las en pares. En cada grupo se calculó la diferencia y al vector resultante se le calculó la desviación estándar. Dicha desviación fue determinada para cada una de las combinaciones de pares posibles dado el número de mediciones. Finalmente se determinó un promedio de las desviaciones resultantes.

Por otra parte también, se realizó la evaluación del nivel de dispersión en los resultados de VOP promediados (cada valor de VOP es el promedio de las mediciones de VOP incluidas en el tiempo de muestreo normalizado de 22 segundos). Estos datos proporcionan una medida de la desviación que pueden presentar los valores de medición dados por el equipo desarrollado en su funcionamiento normal.

Finalmente, luego de la selección del algoritmo más óptimo para la aplicación (aquel

más estable y por tanto, con el menor grado de dispersión en sus resultados), se analizaron los resultados de 30 mediciones hechas a un total de 10 pacientes (3 mediciones por paciente) considerando valores de referencia dados en investigaciones antecedentes. Se llevó a cabo un procedimiento exhaustivo en manera de observar una relación entre los datos recopilados y los valores de referencia.

A continuación se desarrollan más detalladamente los procedimientos mencionados:

#### **4.1 Resultados de comparación de procedimiento manual con los algoritmos propuestos**

Se eligió aleatoriamente un *set* de 10 mediciones realizadas a un paciente sano bajo las mismas condiciones ambientales. Las señales de dicho *set* fueron graficadas en un computador personal y se realizó la identificación manual de los puntos de las señales. El procedimiento manual consistió en visualizar las señales en el computador e identificar visualmente los puntos de referencia, de esta manera se registraron los tiempos correspondientes de cada uno de los puntos. Con los tiempos obtenidos se calculó la diferencia de tiempo entre las señales y luego la velocidad del pulso. El procedimiento mencionado fue realizado por el autor principal del presente trabajo, verificado por profesionales de la salud.

Posterior a la identificación y registro de los puntos en un computador personal, se procedió al cálculo de velocidad.

Los resultados manuales fueron comparados con los obtenidos por los algoritmos propuestos. Cabe destacar que en ambos procesos se realizó el método de eliminación de valores atípicos detallado en la sección 3.14.4.

La comparación se realizó de forma gráfica y de forma estadística, en la figura 4.1 se puede observar claramente la relación que conllevan los resultados manuales y los obtenidos por los distintos algoritmos.

Es importante destacar, que los algoritmos mencionados en la leyenda de la figura 4.1 corresponden a los realizados en las señales de pulso y en el caso de los manuales, a la identificación de puntos en las diferentes zonas de la señal. Por otra parte, para la señal ECG se utilizó para todos los casos, el algoritmo detallado en 3.13.1.4.

En relación al comportamiento de los algoritmos se percibe que el de segunda derivada

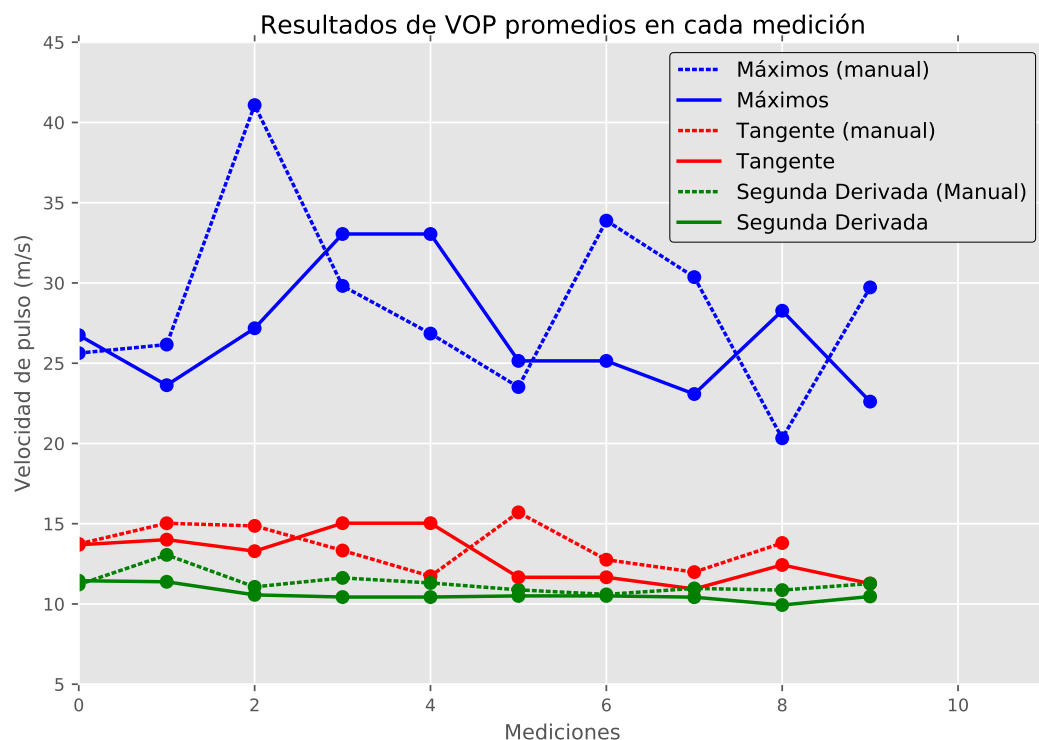


Figura 4.1. Resultados manuales y provistos por algoritmos

máxima describe muy adecuadamente los valores dados por la medición manual. En contraste con el algoritmo de puntos máximos, que ofrece los peores resultados desde el punto de vista gráfico.

Por otra parte, los datos presentados fueron analizados desde un enfoque estadístico. Se realizó una substracción entre los resultados manuales y los obtenidos por el algoritmo correspondiente, el resultado fue elevado al cuadrado y se le calculó el promedio. Es evidente que entre menor es el resultado (en magnitud) de dicho procedimiento, menor es el sesgo con el valor esperado (considerando que el valor esperado es la medición manual). En la tabla 4.1 se observan los resultados obtenidos, los cuales reiteran una menor diferencia entre lo manual y automático con el método de la segunda derivada máxima, tal como se expresó en el análisis gráfico.

Cuadro 4.1. Resultados de comparación entre valores manuales y los provistos por algoritmos

	Máximos	Tangente	Segunda derivada
Promedio de diferencias	51.0436	4.2064	<b>0.7267</b>

Otro punto importante que resalta en el análisis presente es la variabilidad de resultados entre los distintos métodos, en otras palabras, el promedio de los resultados entre un método y otro varía de forma apreciable. Este punto resulta de especial importancia debido a que no se tiene el valor verdadero o exacto de la velocidad (ésto se resolvería mediante la comparación de resultados originados por un dispositivo científico certificado de medición de VOP) sino un conjunto de resultados que conservan un grado de relación entre medición y medición pero cuyos valores difieren entre un método y otro.

Por otra parte, resulta natural que se quiera establecer una preferencia con los algoritmos de tangente y segunda derivada porque sus resultados conservan más cercanía entre ellos que los ofrecidos por el algoritmo de puntos máximos, lo cual hace asumible que este último presenta una estimación con mayor sesgo en relación al valor esperado.

Finalmente considerando los puntos expuestos, puede agregarse que los algoritmos de tangente máxima y segunda derivada máxima presentan resultados con magnitudes mucho mas bajas que el de puntos máximos, argumento que ha de considerarse también secciones posteriores.

www.bdigital.ula.ve

## **4.2 Resultados del análisis de similitud de resultados de VOP por periodos**

### **4.2.1 Resultados de paciente 1**

En la figura 4.2 se muestran todos los resultados de velocidad de pulso calculados con los diferentes algoritmos para cada periodo de la señales adquiridas. Podrá observarse en dicha figura que el algoritmo visualmente con mayor dispersión de datos es el de puntos máximos, siguiendo en el mismo orden el algoritmo de la tangente y finalmente el método de segunda derivada.

Los resultados de desviación estándar obtenidos con el procedimiento de la sección 3.14.2 con cada uno de los algoritmos para el paciente 1 se presentan en la tabla 4.2. Dichos resultados detallan el mismo comportamiento observado en la figura 4.2, en el que el algoritmo de segunda derivada presenta los resultados con el mayor grado de precisión.

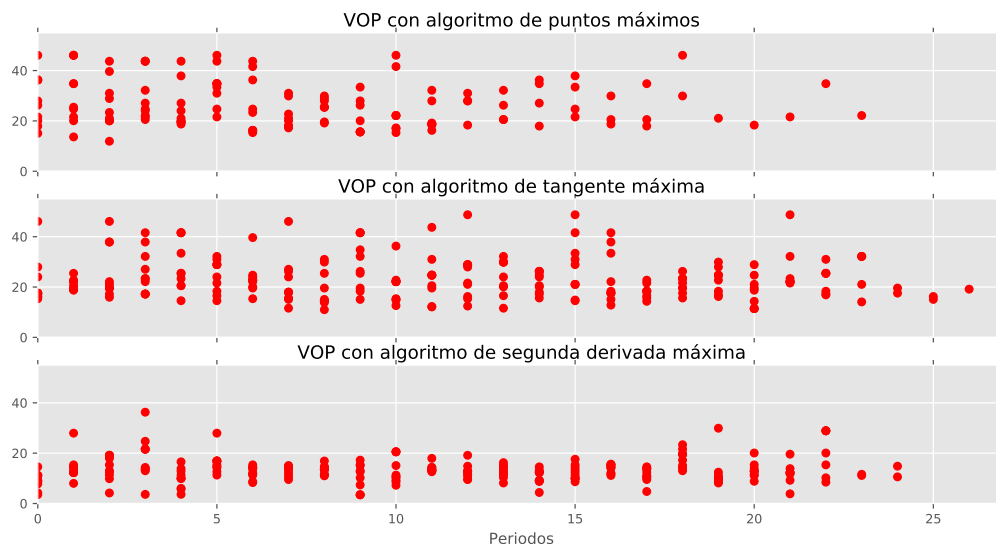


Figura 4.2. VOP por periodos de paciente 1

Cuadro 4.2. Resultados de experimentos de sección 3.14.2 con paciente 1

	Puntos máximos	Tangente máxima	Segunda derivada máxima
$\sigma$	12.4307	11.1710	<b>5.9612</b>

#### 4.2.2 Resultados de paciente 2

De igual manera en la figura 4.3 se observan los resultados de velocidad para cada periodo de las señales del paciente 2 con todos los algoritmos bajo estudio. Nuevamente el algoritmo de segunda derivada es el que ofrece resultados más estables (menos dispersos). Por otra parte, el algoritmo de menor rendimiento en cuanto a precisión fue el de puntos máximos. En la tabla 4.3 se observan los resultados obtenidos.

Cuadro 4.3. Resultados de experimentos de sección 3.14.2 con paciente 2

	Puntos máximos	Tangente máxima	Segunda derivada máxima
$\sigma$	13.2413	6.5081	<b>4.2144</b>

#### 4.2.3 Resultados de paciente 3

En cuanto a los resultados del paciente 3, en la figura 4.4 se detallan los resultados obtenidos. Estos siguen el mismo comportamiento de los del paciente 1 y 2.

De forma similar, el mejor resultado hace referencia al algoritmo de segunda derivada

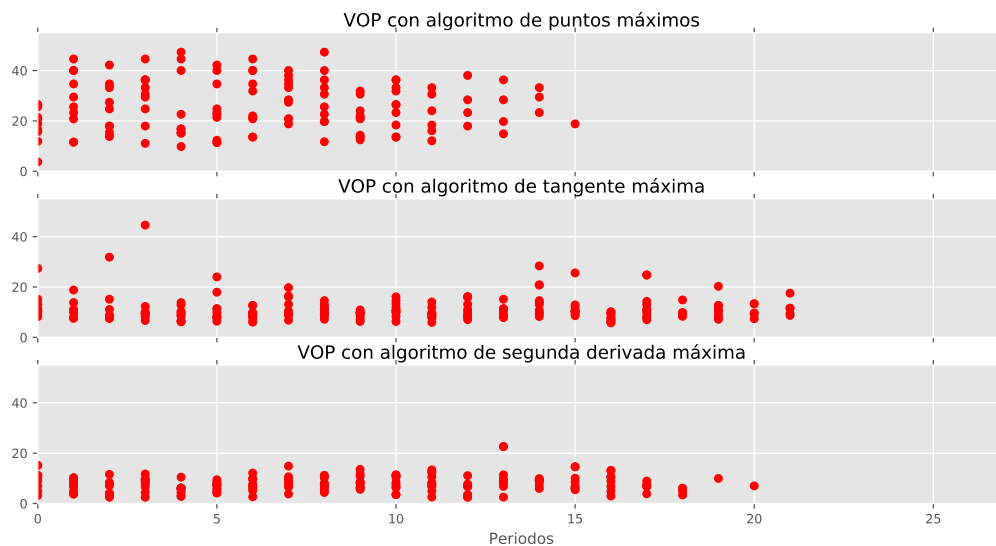


Figura 4.3. VOP por periodos de paciente 2

y el peor al algoritmo de puntos máximos. En la tabla 4.4 se observan los resultados obtenidos.

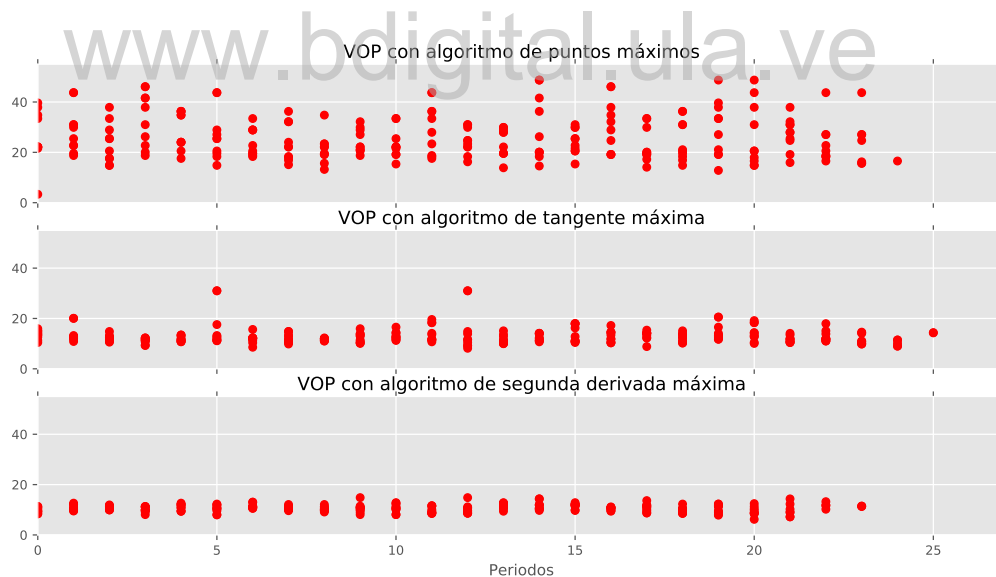


Figura 4.4. VOP por periodos de paciente 3

#### 4.2.4 Resultados de dispersión con todos los pacientes

La tabla 4.5 detalla los resultados generales de todos los pacientes.

Promediando los resultados de cada paciente se observa nuevamente que el algoritmo

**Cuadro 4.4. Resultados de experimentos de sección 3.14.2 con paciente 3**

	<b>Puntos máximos</b>	<b>Tangente máxima</b>	<b>Segunda derivada máxima</b>
$\sigma$	11.1084	3.8556	<b>1.8133</b>

**Cuadro 4.5. Resultados de dispersión con todos los pacientes**

<b>Paciente</b>	<b>Puntos máximos</b>	<b>Tangente máxima</b>	<b>Segunda derivada máxima</b>
<b>1</b>	12.4307	11.1710	5.9612
<b>2</b>	13.2413	6.5081	4.2144
<b>3</b>	11.1084	3.8556	1.8133
<b>Promedio</b>	<b>12.2601</b>	<b>7.1782</b>	<b>3.9963</b>

mo de segunda derivada es el que ofrece el comportamiento más óptimo en cuanto a la reproducibilidad de los resultados.

### 4.3 Resultados del análisis por nivel de dispersión en mediciones promedio de VOP

Se ha mencionado previamente que la evaluación de los algoritmos se ha realizado con un total de 30 mediciones a un total de 3 pacientes (10 mediciones por paciente). Los resultados de la sección anterior son los referentes al procedimiento detallado en 3.14.2, el cual considera cada una de las velocidades dadas a lo largo de los periodos de las señales. En esta sección en cambio (utilizando el mismo *dataset*), se estudió el nivel de variación de las velocidades promedio en cada medición.

En base a lo anterior, en la figura 4.5 se detallan los resultados de VOP al paciente 1 para diferentes mediciones. Cabe destacar que cada una de las mediciones (eje horizontal) fue realizada bajo las mismas condiciones ambientales y en tiempos reducidos (segundos) entre medida y medida.

De igual manera, en las figuras 4.6 y 4.7 se observan los resultados de las velocidades para los pacientes 2 y 3 respectivamente.

El procedimiento de evaluación en esta sección consistió en determinar la dispersión de los resultados de velocidad promedio dados por cada uno de los algoritmos. Dicha dispersión puede ser observada en primera instancia en las curvas de las figuras mencionadas. Los resultados observados gráficamente siguen de manera general el mismo comportamiento

**Cuadro 4.6. Tabla de resultados de desviación en VOP promedios por paciente**

Paciente	Puntos máximos	Tangente máxima	Segunda derivada máxima
1	8.9572	7.7706	4.6097
2	9.3725	4.2894	2.9461
3	7.9760	2.3596	1.3395
<b>Promedio</b>	<b>8.7685</b>	<b>4.8065</b>	<b>2.9651</b>

detallado en la sección 4.2. El algoritmo con mayor dispersión es el de puntos máximos seguido de la tangente y por último (y más favorable) el de segunda derivada.

En la tabla 4.6 se detallan las desviaciones estándar resultantes para cada uno de los métodos en todos los pacientes, los cuales expresan las mismas conclusiones. El método de segunda derivada provee resultados más idóneos y el algoritmo de puntos máximos los menos óptimos.

**Cuadro 4.7. VOP (m/s) promedios por paciente**

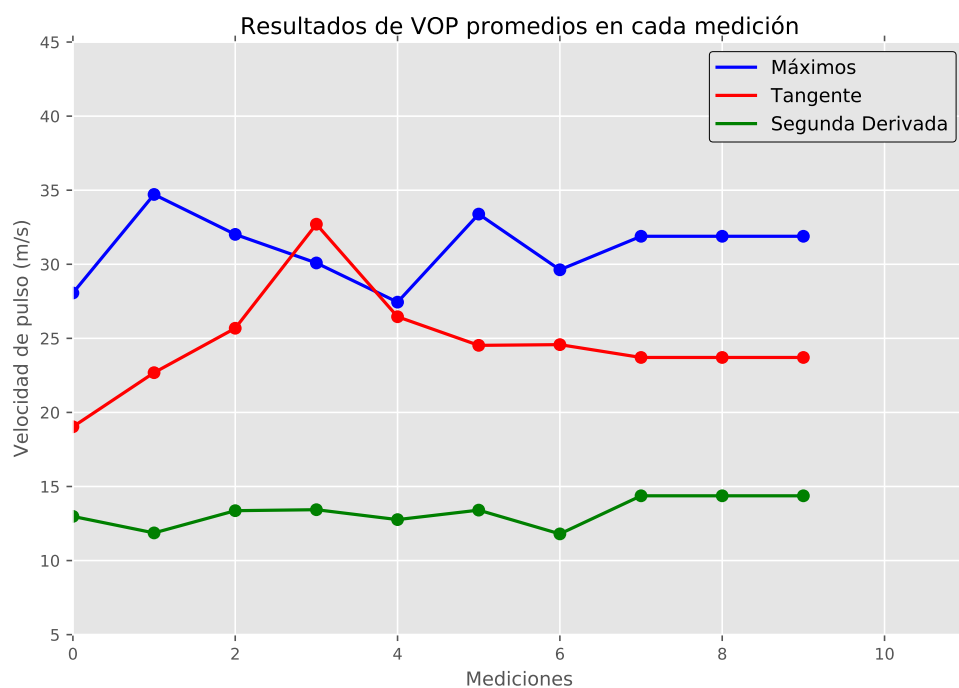
Paciente	Puntos máximos	Tangente máxima	Segunda derivada máxima
1	26.5462	23.5347	13.2733
2	25.7070	11.0353	7.8140
3	25.9867	12.9004	10.6082

Por último en la tabla 4.7 se observan los promedios de velocidad de las mediciones ilustradas en las figuras 4.5, 4.6, 4.7. Los resultados de velocidad detallados están contenidos dentro del rango válido de velocidades pulso posibles de acuerdo a la bibliografía. De forma más específica en [1] se detallan los siguientes intervalos:

- Arteria aorta: 3 a 5 m/s.
- Arterias grandes: 7 a 10 m/s.
- Arterias medianas: 15 a 35 m/s.

Conociendo que los puntos de medición en los experimentos comprenden el trayecto de una arteria grande como lo es la arteria femoral, una arteria mediana como la arteria tibial y pequeñas como las metatársicas. Todos los resultados dados por los algoritmos pudieran ser resultados válidos pues las amplitudes de las velocidades resultantes están





**Figura 4.5. Resultado de VOP promedios en mediciones paciente 1**

comprendidas entre el extremo superior de las arterias grandes y el extremo inferior de las arterias medianas.

Finalmente, de acuerdo a los resultados de todas las secciones del presente capítulo, se ha seleccionado como el algoritmo más apto en base al nivel de similitud en resultados y estabilidad, al método de segunda derivada máxima.

#### 4.4 Comparación de las mediciones realizadas con valores de referencia

De acuerdo al procedimiento mencionado en 3.15, se tomaron 3 mediciones de VOP a 10 pacientes (3 mediciones por paciente), donde además fueron recopilados datos como la talla, peso, presión arterial y edad de cada persona. Las señales de pulso fueron procesadas con el algoritmo de puntos mínimos o de segunda derivada máxima y la señal ECG con el procedimiento detallado en la sección 3.13.1.4.

Por otra parte, los valores de VOP considerados como referencia fueron extraídos de [2], donde se establecieron tres grupos poblacionales: Europa, Argentina y Uruguay. Dichos

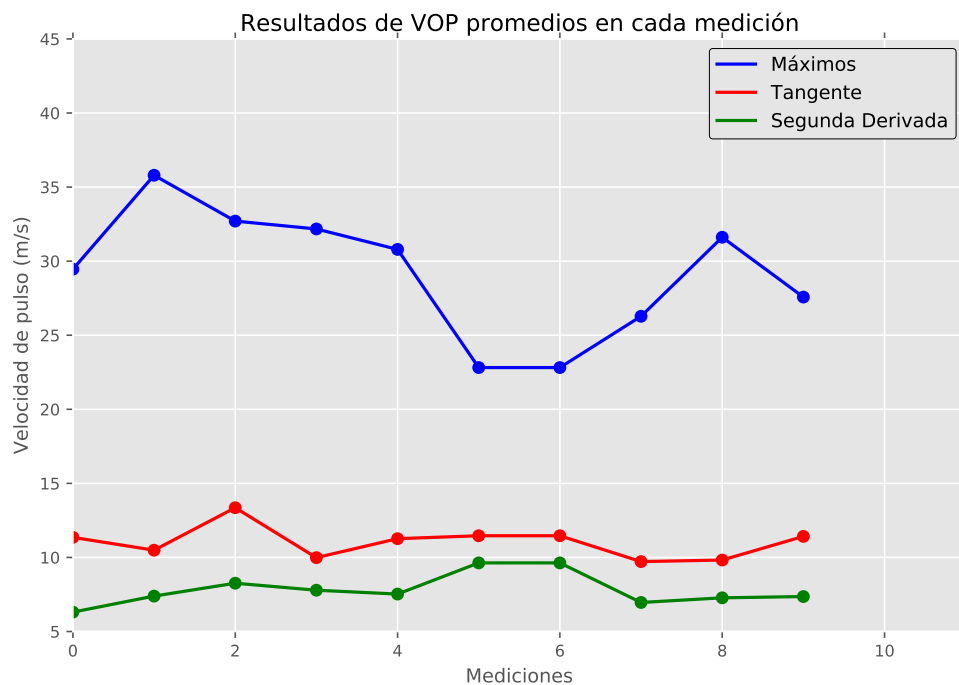


Figura 4.6. Resultado de VOP promedios en mediciones paciente 2

www.bdigital.ula.ve

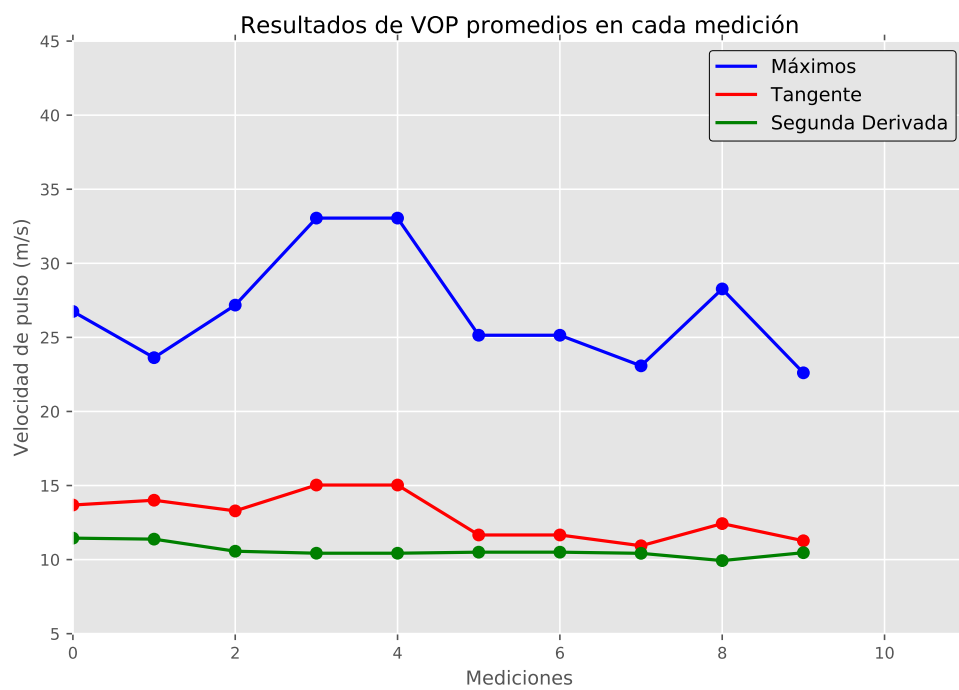


Figura 4.7. Resultado de VOP promedios en mediciones paciente 3

C.C.Reconocimiento

Cuadro 4.8. Tabla de valores de referencia detallados en [2]

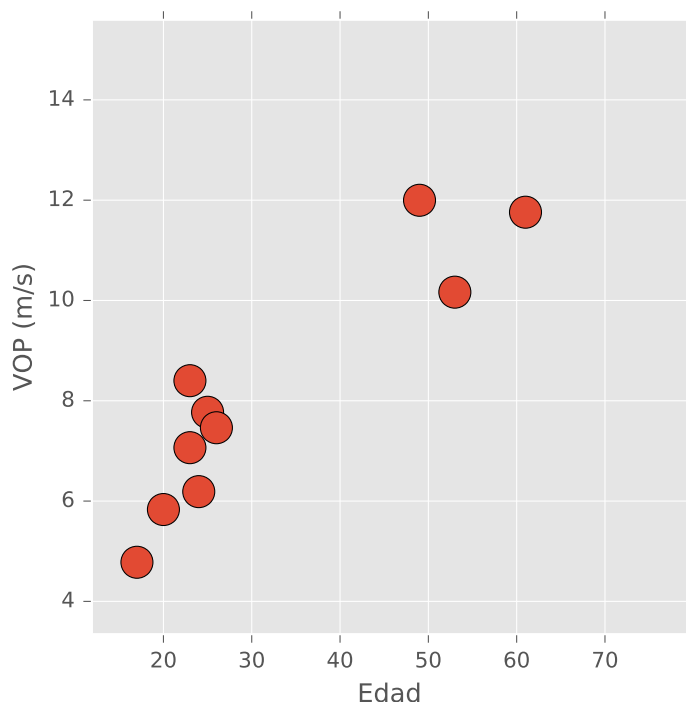
Grupo de edades (years)	Europe Mean +/- 2SD	Uruguay Mean +/- 2SD	Argentina Mean +/- 1SD
10-19	NDA	5.8 (4.7 - 6.8)	5.0 +/- 0.7 (4.9-5.1)
<30	6.2 (4.7-7.6)	6.9 (4.4-9.4)	5.8 +/- 0.9 (5.6-6.0)
30-39	6.5 (3.8-9.2)	7.7 (6.1-9.2)	6.3 +/- 0.8 (6.1 - 6.4)
40-49	7.2 (4.6 - 9.8)	7.8 (6 - 9.7)	6.8 +/- 0.9 (6.6 -7.0)
50-59	8.3 (4.5 -12.1)	8.9 (6.4 - 11.5)	8.1 +/- 1.1 (7.9 - 8.3)
60-69	10.3 (5.5 - 15)	NDA	8.4 +/- 1.0 (8.2 - 8.6)
>70	10.9 (5.5 -16.3)	NDA	9.0 +/- 2 (8.2 - 9.7)

datos se detallan en la tabla 4.8, donde las VOP han sido separadas en secciones por edades (en períodos de 10 años exceptuando todos los menores de 30 años y los mayores de 70 años) y además se detallan valores máximos y mínimos de acuerdo a la desviación estándar de los datos correspondientes.

En cuanto a las mediciones realizadas a los pacientes, se obtuvieron los resultados detallados en la figura 4.8. Cabe destacar que cada valor detallado representa la VOP promedio en cada paciente. Puede observarse un claro comportamiento creciente de la VOP a medida que se incrementa la edad. En cuanto a la magnitud de las velocidades, en la figura 4.9 se observa la comparación de dichas velocidades con los valores de referencia para cada localidad. Las líneas de cada una de las poblaciones fueron construidas con los datos provistos en la tabla 4.8 y con regresiones lineales de segundo grado.

Los valores obtenidos se mostraron dentro de los intervalos de referencia correspondientes a la población de Uruguay, en menor medida a los de Europa y por último a los de Argentina. En cuanto a estos últimos, es necesario destacar que los datos de la población argentina poseen una desviación bastante inferior en relación al resto, por lo que el rango de valores aceptables para cada década de edades es mucho más estrecho que para el resto de las poblaciones.

Con respecto a los otros parámetros como el peso, la talla y la presión arterial, se realizaron múltiples visualizaciones en manera de identificar algún comportamiento o correlación. Particularmente, la talla y el peso se utilizaron para determinar el índice de masa corporal de los pacientes dado por la ecuación  $imc = peso/talla^2$ , dicho índice y los



**Figura 4.8. Resultados de VOP en relación a la edad de los pacientes**

valores de VOP se presentan en la figura 4.10. Puede notarse que la velocidad tiende a ser mayor a medida que el índice aumenta.

Se observó también la relación entre los datos de presión arterial y velocidad de pulso. La presión sistólica y la diastólica fueron utilizadas para calcular la presión arterial media por medio de la ecuación 4.1. En la figura 4.11 se muestra el comportamiento entre la presión arterial media y la velocidad de pulso para los distintos pacientes. Aunque con un grado de correlación menor que las variables detalladas por medio de las figuras 4.8 y 4.10 (probablemente debido a la poca precisión del equipo de medición de presión arterial), se observa un incremento de la velocidad de pulso conforme a aumentos en la presión arterial media.

$$PAM = \frac{2}{3}(PAD) + \frac{1}{3}(PAS) \quad (4.1)$$

En base a los resultados puede concluirse que el equipo desarrollado cumple el desempeño requerido para la medición de la VOP. Por otra parte también, se observó la influencia que ejerce la edad, el índice de masa corporal y la presión arterial en la velocidad de la

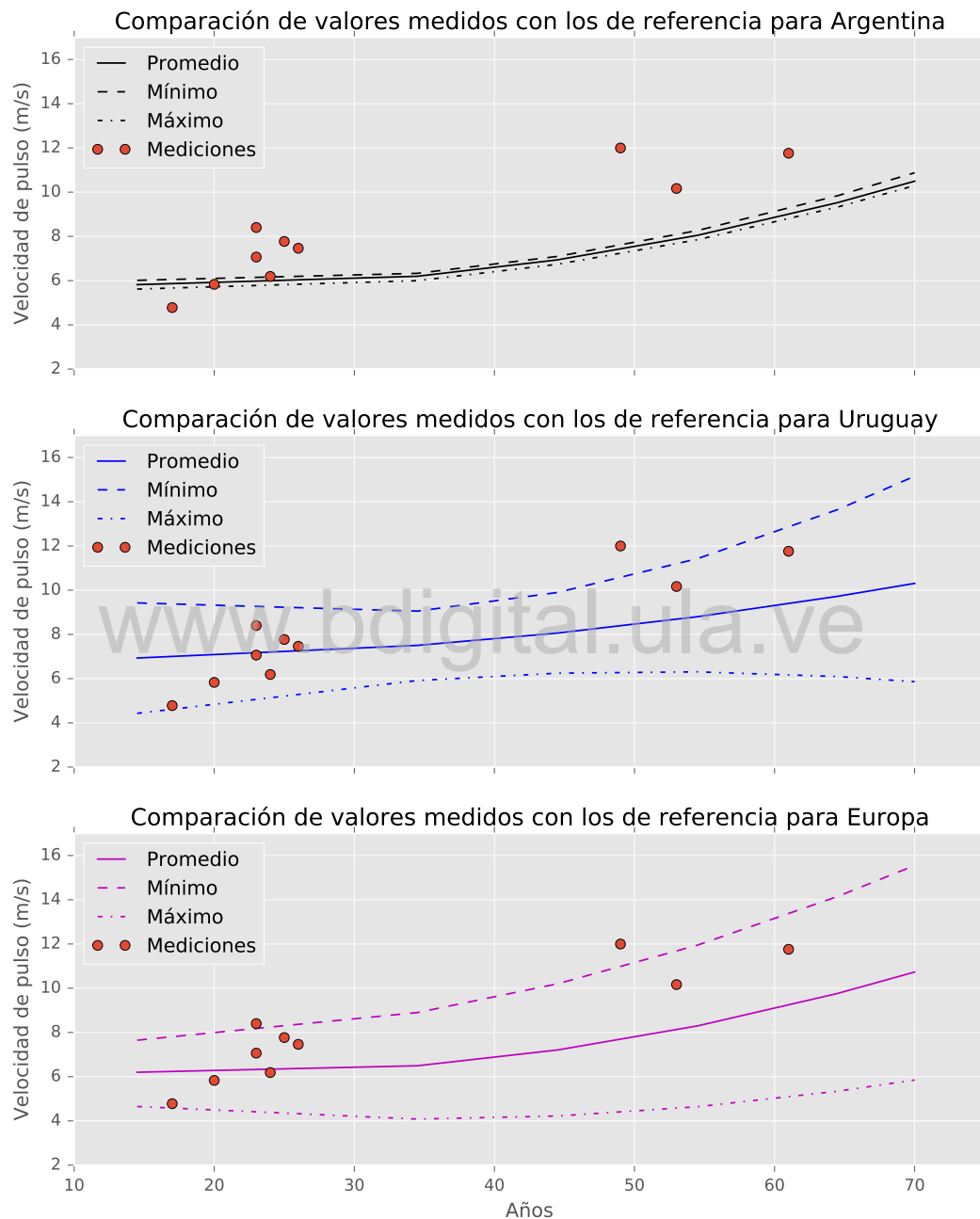


Figura 4.9. Comparación de resultados con valores de referencia para las distintas poblaciones

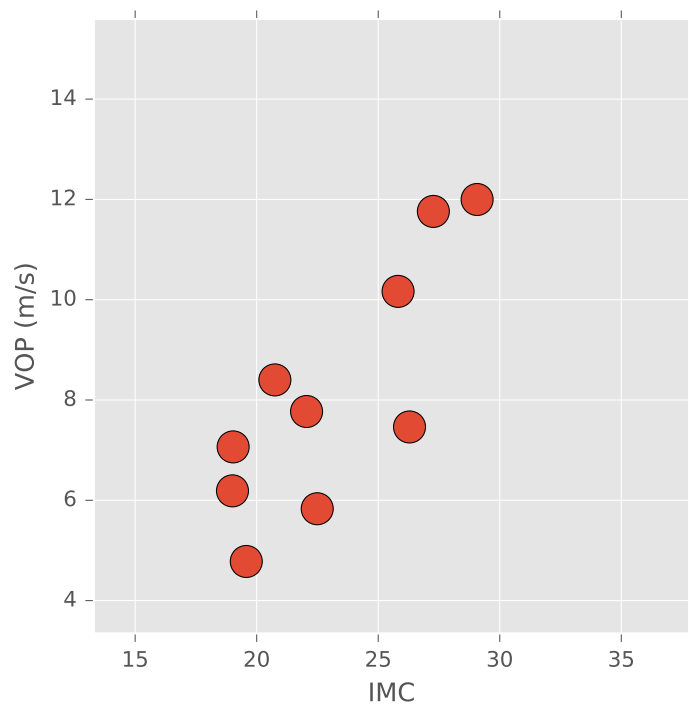


Figura 4.10. Resultados de VOP en relación al índice de masa corporal del paciente

[www.bdigital.ula.ve](http://www.bdigital.ula.ve)

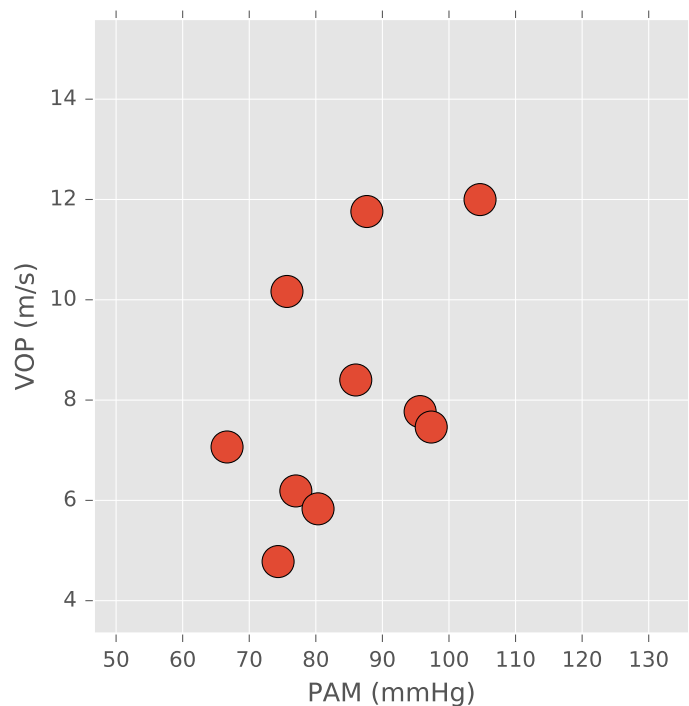


Figura 4.11. Velocidad de pulso en relación con la presión arterial media de los pacientes

C.C.Reconocimiento

onda de pulso.

## 4.5 Cálculo de VOP en interfaz gráfica

En base a los resultados de las secciones 4.2 y 4.3 el algoritmo seleccionado fue el de segunda derivada. Por ende, dicho algoritmo fue implementado en la interfaz gráfica para el cálculo de la velocidad.

En la interfaz gráfica, el cálculo de la VOP sólo es habilitado cuando ha finalizado exitosamente el proceso de muestreo.

Cuando el usuario presiona el botón *Calcular*  $\Sigma$ , se inicia proceso de identificación de puntos y posteriormente el cálculo de la velocidad. Cuando éste finaliza se muestran los resultados de velocidad y frecuencia cardíaca del paciente.

La frecuencia cardíaca es calculada simplemente contando los puntos máximos de la señal electrocardiográfica y el tiempo total de muestreo (que es aproximadamente de 22 segundos), luego se realiza una regla de tres simple para estimar el resultado a 60 segundos.

En la figura 4.12 se observa el resultado final del procesamiento para una medición de un paciente luego de presionarse el botón *Calcular*  $\Sigma$ . Los resultados de velocidad y frecuencia cardíaca son mostrados en el campo de texto de mensajes. Adicionalmente, las señales son graficadas nuevamente luego de su procesamiento (que incluye filtrado) y se muestran los puntos identificados en cada una de ellas.



Figura 4.12. Cálculo de velocidad de pulso en interfaz gráfica



## CONCLUSIONES

- Se ha desarrollado un equipo para la medición de la velocidad de onda de pulso por fotopleletismografía. El dispositivo incluye la adquisición de la señal electrocardiográfica y dos señales de pulso y su comunicación en el computador. Además de una aplicación donde el usuario puede realizar la medición de la velocidad de la onda de pulso y la frecuencia cardíaca del paciente. También puede:
  - Guardar la información recopilada en una base de datos local.
  - Visualizar las mediciones y pacientes registrados en dicha base de datos.
  - Generar archivos de reporte PDF con la información del paciente en conjunto con las bioseñales obtenidas.
- En cuanto a los estudios con los algoritmos, se ha seleccionado el método de segunda derivada para la identificación de puntos de medición porque ofrece los resultados con menor dispersión (desviación estándar por periodos de las señales de 3.9969 frente a 7.1782 del algoritmo de la tangente y 12.2601 de puntos máximos).
- Los resultados de VOP carótida-femoral estuvieron dentro de los intervalos de valores de referencia para dos grupos poblacionales importantes y además guardaron relación con referencias provistas en la bibliografía [1]. Los resultados por otra parte, reiteraron un comportamiento creciente de la VOP ante el incremento de la edad, el índice de masa corporal y en menor medida, la presión arterial media.

## RECOMENDACIONES

El equipo desarrollado en el presente trabajo es un prototipo para la medición de la onda de pulso. A pesar del buen rendimiento que presenta (considerando la calidad de las señales adquiridas y la reproducibilidad de los datos), es importante recalcar que para una posible aplicación comercial o para su uso en investigaciones científicas en cardiología, es necesario realizar múltiples mejoras que pueden formar parte del trabajo futuro para próximas investigaciones. Entre ellas las siguientes:

- Los filtros analógicos activos en la tarjeta de adquisición de datos son amplificadores LM324. Estos integrados son de uso general y fueron seleccionados por la disponibilidad de éstos en el mercado. Uno de los problemas con este amplificador operacional es que se satura a partir de 3.5 V si es alimentado en el rango de tensión 0–5 V. Principalmente esta característica afectó la resolución efectiva del convertidor analógico digital, pues el rango real de la señal estuvo limitado a 0 –3.5 V.
- Se observó durante el presente trabajo que la diferencia de amplitud de la señal electrocardiográfica entre pacientes puede variar significativamente. Por esta razón pudiera ser favorable la adición de un circuito de control de ganancia para la amplificación de ECG en manera de normalizar las amplitudes de dicha onda.
- El sensor tipo pinza de Marca Nellcor seleccionado en la sección 3.2 provee una alta estabilidad en la adquisición de la bioseñal del pulso pero no así el sensor por reflexión. Es necesario un estudio para la elaboración de un sensor óptimo para la medición del pulso en zonas donde hay mayor tejido o grasas, o donde se requiere mayor potencia de la luz incidente y mayor robustez ante movimientos del paciente y ruidos luminosos del ambiente.
- El uso del Arduino Uno rev 3 facilitó la implementación de comunicaciones vía puerto serial con el computador, así como entre otros dispositivos como el DAC y al ADC contenidos en la tarjeta de adquisición de datos. Sin embargo, el microcontro-

lador embebido del Arduino Uno es considerado de gama media. La sustitución del dispositivo Arduino por un DsPIC o una computadora de tarjeta única como Raspberry pudiera habilitar mayores funcionalidades y una mayor rapidez de adquisición y transferencia de datos.

- El desarrollo del software para el computador fue realizado en Python. Este es un lenguaje interpretado y aunque sencillo de aprender y con múltiples librerías científicas, puede resultar un poco lento en el procesamiento de datos. Una aplicación desarrollada en un lenguaje compilado pudiera acelerar significativamente la adquisición y el procesamiento de datos mencionado.
- La base de datos implementada es básicamente una estructura de almacenamiento de archivos y carpetas. Esto es muy poco escalable e ineficiente cuando el número de pacientes y mediciones aumenta significativamente. Debe diseñarse una base de datos relacional por medio de lenguajes de manejo de datos como SQL para atender a la escalabilidad y facilitar futuros estudios con mayor número de mediciones y pacientes.

## REFERENCIAS

- [1] G. y Hall., *Tratado de Fisiología Médica*, 12ta edición. Elsevier B. V., 2011, cap. Distensibilidad vascular y funciones de los sistemas arterial y venoso, pp. 167–176.
- [2] A. Díaz, A. J. Ramírez, E. Ignacio, y C. Fischer, “Reference values of pulse wave velocity in south america: A call to action,” *Annals of Clinical and Experimental Hypertension*, vol. 3, pp. 1–2, 2015.
- [3] G. Tortora y B. Derrickson, *Principios de Anatomía y Fisiología*, 11ta edición. Editorial Panamericana, 2015.
- [4] P. J. Lynch, “Coronaire,” [Sep 2, 2015]. [Online]. Available: <https://commons.wikimedia.org/wiki/File:Coronaire.png#/media/File:Coronaire.png>
- [5] B. R. Teja, “Calculation of blood pulse transit time from ppg,” Tesis, National Institute of Technology Rourkela, 2012.
- [6] P. V. R. Farina, “Diseño, implementación y validación de un nuevo método para la medición del intervalo qt,” Tesis de Maestría, Universidad Simón Bolívar, 2013.
- [7] World Health Organization, “Top 10 causes of death,” 2012, [Feb 24, 2015]. [Online]. Available: [http://www.who.int/gho/mortality\\_burden\\_disease/en/](http://www.who.int/gho/mortality_burden_disease/en/)
- [8] Sociedad Chilena de Obesidad, “Rigidez arterial y riesgo cardiovascular,” 2007, [Feb 24, 2015]. [Online]. Available: <http://www.sochob.cl/web1/rigidez-arterial-y-riesgo-cardiovascular/>
- [9] C. Estadella, S. Vázquez, y A. Oliveras, “Hipertensión y riesgo vascular,” *Elsevier Doyma*, vol. 27, pp. 203–210, 2009.

- [10] C. Galli, J. Camus, E. Cabrera, y M. Risk, “La velocidad de la onda del pulso en la evaluación cardiovascular de pacientes con insuficiencia renal,” *Revista Federación Argentina de Cardiología*, vol. 33, pp. 212–217, 2004.
- [11] J. B. Sol, “Aplicación práctica de la medida de la onda de pulso en el estudio de la rigidez arterial,” *Revista Elsevier Saunders*, vol. 24, pp. 30–34, 2007.
- [12] R. Gonzales, O. Morales, J. Delgado, J. F. J.M. Padilla, y J. Saiz, “Automatic brachial ankle pulse wave velocity measurements for vascular damage assessments,” *Proc. Computers in Cardiology*, vol. 35, pp. 173–176, 2008.
- [13] J. Gu-Young, Y. Kee-Ho, y K. Nam-Gyun, “Continuous blood pressure monitoring using pulse wave transit time,” *Proc. 26<sup>th</sup> IEEE EMBS*, pp. 738–741, 2004.
- [14] Freescale Semiconductor, Appl. Note AN4327, pp.1-39. 2012.
- [15] E. Kazanavicius, R. Gircys, y A. Vrubliauskas, “Mathematical methods for determining the foot point of the arterial pulse wave and evaluation of propose methods,” *Information technology and control*, vol. 34, pp. 29–36, 2005.
- [16] J. C. Murphy, K. Morrison, J. McLaughlin, G. Manoharan, y J. Adgey, “An innovative piezoelectric-based method for measuring pulse wave velocity in patients with hypertension,” *The Journal of Clinical Hypertension*, vol. 13, pp. 497–505, 2011.
- [17] M. Chmelar y R. Ciz, “Problems of measuring pulse wave velocity,” *Proc. 8<sup>th</sup> WSEAS CSECS*, pp. 320–323, 2009.
- [18] E. Kipge-Kviesis, O. R. J Zaharans, y A. Grabovskis, “A photoplethysmography device for multipurpose blood circulatory system assessment,” *Proc. SPIE-OSA*, vol. 8090, pp. 1–7, 2011.
- [19] M. Elias, G. Dore, A. Davey, W. Abhayaratna, A. Goodell, y M. Robbins, “Norms and reference values for pulse wave velocity,” *The Journal of Bioscience and Medicine*, vol. 1, pp. 1–10, 2011.
- [20] A. Tucci, *Instrumentación Biomédica*, 1ra edición. Universidad de los Andes, 2005.

- [21] F. Saraví, “Presión y flujo en las arterias,” Reporte Técnico, 2005.
- [22] M. Khoo, *Physiological Control Systems*, 1ra edición. Wiley Interscience, 2000, cap. Mathematical Modeling, pp. 13–23.
- [23] F. Clara, M. Clayrol, A. Scandurra, G. Meschino, M. Garzillo, E. Moyano, y A. Introzzi., “La técnica de análisis de onda de pulso en la determinación del riesgo cardiovascular,” *Revista de la Federación Argentina de Cardiología*, vol. 34, pp. 213–220, 2005.
- [24] C. Estadella, S. Vázquez, y A. Oliveras., “Rigidez arterial y riesgo cardiovascular,” *Elsevier Doyma*, vol. 27, pp. 203–210, 2009.
- [25] S. Bagha y L. Shaw, “A real time analysis of ppg signal for measurement of spo2 and pulse rate,” *International Journal of Computer Applications*, vol. 36, pp. 45–50, 2011.
- [26] P. M. Middleton, G. S. H. Chan, E. O’Lone, E. Steel, R. Carroll, B. G. Celler, y N. H. Lovell, “Spectral analysis of finger photoplethysmographic waveform variability in a model of mild to moderate haemorrhage,” *Journal of Clinical Monitoring and Computing*, vol. 36, pp. 45–50, 2008.
- [27] J. Lee, W. Jung, I. Kang, Y. Kim, y G. Lee, “Design of filter to reject motion artifact of pulse oximetry,” vol. 26, 2004, pp. 241–249.
- [28] S. Borik, “Pulse wave velocity measurements,” en *XIV International PhD Workshop OWD*, 2012, pp. 324–327.