



UNIVERSIDAD DE CASTILLA-LA MANCHA
ESCUELA SUPERIOR DE INGENIERÍA INFORMÁTICA

GRADO EN INGENIERÍA INFORMÁTICA
TRABAJO FIN DE GRADO
TECNOLOGÍA ESPECÍFICA DE INGENIERÍA DE COMPUTADORES

mHolter: Prototipo basado en microcontrolador para
recogida de datos clínicos

Juan Manuel Puñal Bargueño

Julio de 2014



UNIVERSIDAD DE CASTILLA-LA MANCHA
ESCUELA SUPERIOR DE INGENIERÍA INFORMÁTICA

Departamento de Sistemas Informáticos

TRABAJO FIN DE GRADO
TECNOLOGÍA ESPECÍFICA DE INGENIERÍA DE COMPUTADORES

**mHolter: Prototipo basado en microcontrolador para
recogida de datos clínicos**

Autor: Juan Manuel Puñal Bargueño
Directores: José Luis Sánchez García
Francisco José Alfaro Cortés

Julio de 2014



UNIVERSIDAD DE CASTILLA-LA MANCHA
ESCUELA SUPERIOR DE INGENIERÍA INFORMÁTICA

Declaración de autoría

Yo, Juan Manuel Puñal Bargueño, con DNI 03906126-J, Declaro que este trabajo es original, de mi autoría, que se han citado las fuentes correspondientes y que en su ejecución se respetaron las disposiciones legales que protegen los derechos de autor vigentes.

En Albacete, a 3 de julio de 2014

Fdo.: Juan Manuel Puñal Bargueño

Resumen

Los avances tecnológicos están presentes en multitud de ámbitos y se aplican en la mayoría de los casos con indudable éxito. Este es el caso, por ejemplo, del campo de la salud en el cual, además, se está comprobando que es totalmente necesario apoyarse en la tecnología más avanzada para obtener mejores resultados en el diagnóstico y tratamiento de los pacientes.

Aunque hospitales y centros de salud han experimentado cambios significativos en este sentido, todavía queda un largo camino que recorrer para alcanzar los niveles de implantación que serían deseables. En el ámbito de la salud es muy común el hecho de que ciertos sistemas e incluso dispositivos tengan un coste elevado y por tanto no es fácil su adquisición. Es cierto, sin embargo, que en el mercado hay productos más económicos pero por contra no ofrecen todas las funciones deseables.

En este trabajo hemos realizado un dispositivo para la recogida de datos biomédicos. Se trata de un dispositivo muy económico para el que se han usado componentes totalmente accesibles y de bajo coste, así como software libre.

El prototipo recoge un numeroso conjunto de parámetros clínicos y de movimiento que podrán ser utilizados en diferentes ámbitos y con diversos propósitos. Además, con él, se quiere mostrar que es posible obtener dispositivos de monitorización de bajo coste y con una aceptable funcionalidad.

*A Paulino,
que en sólo seis años dio
un nuevo sentido a mi vida.*

*A mi familia,
que me han apoyado en todas mis
decisiones y ayudado con sus consejos.*

Agradecimientos

En primer lugar quiero expresar mi más sincero agradecimientos a José Luis Sánchez y Francisco José Alfaro, directores de este Trabajo de Fin de Grado, sin los cuales no habría sido posible su planteamiento y desarrollo. Gracias a ellos por su trabajo y dedicación.

A todo el personal docente de la Escuela Superior de Ingeniería Informática, aunque no haya tenido el honor de recibir alguna clase de ellos, la calidad humana que atesoran es incalculable y su pasión por la docencia hacen avanzar a cualquier alumno tanto a nivel académico como en lo personal.

A Miguel Iniesta, Antonio Quintanilla y Juan José Bravo, no por sus horas de clase, sino por su pasión y conocimientos en electrónica que tanto me han ayudado.

A todo el personal no docente que con su amabilidad, simpatía y profesionalidad han hecho que el paso por esta etapa de nuestra vida sea más agradable, ya sea con un simple *“buenos días”* junto a una sonrisa o librándonos del papeleo burocrático.

A todos mis compañeros durante estos cuatro años, en especial a Enrique de la Torre, David Jiménez, Javier Lapeña, Jaime Antonio Monje, Santiago Montoya y Sergio Novillo que gracias al apoyo que hemos tenido entre nosotros hemos superado todas las adversidades ya sean académicas o personales.

A mi familia, que no han dudado de mí, en especial a mi abuela y mi madre aunque las he tenido en la distancia las he sentido muy cerca en todo momento.

A María, y toda su familia, en especial a Paulino, su padre, y Consuelo, su madre, que me animaron y apoyaron incondicionalmente en esta gran aventura.

A todos mis Némesis por hacerme la vida más injusta y dura, gracias a esto me hicieron mucho más fuerte y astuto.

Índice general

ÍNDICE DE FIGURAS	IX
Lista de Figuras	IX
ÍNDICE DE TABLAS	XIII
Lista de Tablas	XIII
1. INTRODUCCIÓN Y MOTIVACIÓN	1
1.1. Estructura de la memoria	3
2. OBJETIVOS Y METODOLOGÍA	5
2.1. Objetivos	5
2.2. Metodología	6
3. ESTADO DEL ARTE	9
3.1. Antecedentes	9
3.2. Estado actual	11
4. DESCRIPCIÓN DEL PROTOTIPO	17
4.1. ECG	18
4.2. Oximetría	23
4.3. Presión sanguínea	29
4.4. Temperatura	31
4.5. Aceleración y Posición angular	37
4.6. RTC	41
4.7. Almacenamiento de datos	44
4.8. Software	49
5. EVALUACIÓN	55
5.1. Pruebas de funcionamiento	55
5.2. Consumo	62
5.3. Coste	63

6. CONCLUSIONES Y TRABAJO FUTURO	65
6.1. Conclusiones	65
6.2. Trabajo futuro	67
Bibliografía	69
BIBLIOGRAFÍA	70
CONTENIDO DEL CD	71
A. ESQUEMAS ELÉCTRICOS	73
A.1. Esquema eléctrico de la oximetría	75
A.2. Esquema eléctrico del electrocardiograma	77
A.3. Esquema eléctrico del microcontrolador, RTC, acelerómetro y tarjeta SD	79
B. PARTES DEL PROTOTIPO	81
C. MANUAL DE USUARIO	83
C.1. Preparación del prototipo	83
C.2. Puesta en marcha del prototipo	86
C.3. Extracción de datos	88

ÍNDICE DE FIGURAS

2.1. Diagrama de flujo de la metodología	8
3.1. Primer transmisor del Dr. Holter.	10
3.2. Monitor holter de principio de los años 80 con grabación en cinta magnética.	11
3.3. Monitor holter DMS-840-3D de Tecnomed2000.	12
3.4. Monitor holter h3+ de Mortara Instruments.	13
3.5. Monitor holter Merlin de Mediatech.	13
3.6. Radiografía de un monitor holter implantable.	14
3.7. Monitor holter Reveal plus de Medtronic.	14
3.8. Proyecto Mobilecg.	15
3.9. E-Healt Arduino Shield.	15
4.1. Representación gráfica de la actividad eléctrica del corazón.	19
4.2. Distintas fases del electrocardiograma.	19
4.3. Detalle del INA321 en el esquema.	20
4.4. Posición de los puntos de la ECG.	22
4.5. Simulación del funcionamiento de la ECG en Proteus.	22
4.6. Implementación del circuito de la ECG en el prototipo final.	23
4.7. Captura generada por el osciloscopio para la ECG.	24
4.8. Extinción (absorción) de la luz por la hemoglobina oxigenada y reducida [14].	25
4.9. Curva empírica sobre la saturación del oxígeno (vs. R) [20].	26
4.10. Simulación de la oximetría con el software Proteus.	27
4.11. Implementación del circuito del oxímetro en el prototipo final.	28
4.12. Prueba de concepto del oxímetro en hardware real y Labview.	28
4.13. Conexión del LM35.	34
4.14. Simulación del funcionamiento del LM35 en Proteus.	35
4.15. Implementación del LM35 en el prototipo final.	36
4.16. Prueba del hardware de la temperatura con Labview.	36
4.17. Orientación y polaridad de los ejes del MPU-6050.	38

4.18. Orientación de los ejes del giroscopio en el cuerpo humano.	39
4.19. Un esquema representativo de un bus I^2C con un maestro y 3 esclavos.	40
4.20. Implementación del MPU-6050 en el prototipo final.	41
4.21. Pinout del DS3231.	42
4.22. Simulación del funcionamiento del DS3231 en Proteus.	43
4.23. Implementación del DS3231 en el prototipo final.	44
4.24. Memoria EEPROM de 256 Kbits (32 KBytes) fabricada por Microchip.	45
4.25. Microdrives del fabricante Hitachi.	45
4.26. Memoria FLASH de Intel con una capacidad de 16 gigabytes.	46
4.27. Simulación del funcionamiento de la tarjeta SD en Proteus.	47
4.28. Opción del WinImage en el cual podemos crear un fichero de imagen con formato FAT32.	48
4.29. Implementación del módulo SD en el prototipo final.	49
4.30. Diagrama de flujo de la ISR <code>dmpDataReady</code>	50
4.31. Diagrama de flujo de la función <code>setup</code>	51
4.32. Diagrama de flujo de la ISR <code>ADC_vect</code>	52
4.33. Diagrama de flujo del bucle principal.	53
5.1. Colocación del prototipo en un voluntario.	56
5.2. Gráfica representando las pulsaciones por minuto en el tiempo.	57
5.3. Gráfica representando la saturación de oxígeno en el tiempo.	57
5.4. Gráfica representando la temperatura corporal en el tiempo.	58
5.5. Gráfica representando el PTT en el tiempo.	59
5.6. Colocación del prototipo para la prueba del movimiento.	59
5.7. Gráfica representando el movimiento en la prueba con el voluntario an- dando.	60
5.8. Gráfica representando el movimiento en la prueba con el voluntario co- rriendo.	60
5.9. Gráfica representando el movimiento en la prueba con el voluntario si- mulando una caída junto con el pulso.	61
5.10. Gráfica representando los datos del giroscopio durante la simulación de una caída.	62
5.11. Consumo de nuestro prototipo.	63
A.1. Esquema eléctrico de la oximetría	75
A.2. Esquema eléctrico del electrocardiograma	77
A.3. Esquema eléctrico del microcontrolador, RTC, acelerómetro y tarjeta SD	79
B.1. Partes del prototipo.	82

C.1. Conexión del ECG.	83
C.2. Conector de la sonda pulsioximétrica.	84
C.3. Posición de los parches de la ECG.	84
C.4. Colocación de la sonda pulsioximétrica en el paciente.	85
C.5. Colocación de la tarjeta SD.	85
C.6. Colocación del prototipo en el paciente.	86
C.7. Conector de alimentación de prototipo.	86
C.8. Diodos LED de depuración.	87
C.9. Lector de tarjetas.	88

ÍNDICE DE TABLAS

3.1. Clasificación de los monitores Holter	12
4.1. Principales características del ATmega2560.	18
4.2. Comparación entre diferentes tipos de memoria para almacenamiento. .	46
5.1. Datos de referencia de los voluntarios.	55
5.2. Consumo teórico del prototipo.	62
5.3. Lista de componentes y precios.	64

Capítulo 1

INTRODUCCIÓN Y MOTIVACIÓN

Como es de sobra conocido por todos, la tecnología está presente en numerosas, si no todas, las actividades de nuestra vida cotidiana: trabajo, compras, transportes, estudios..., y también en aquellas relacionadas con la salud. Este campo se está beneficiando especialmente de los avances tecnológicos que cada vez se suceden a una mayor velocidad.

Muchas son las áreas de la salud en las que progresivamente se están incorporando nuevas herramientas y metodologías basadas en las tecnologías. Desde las recetas o las historias clínicas electrónicas hasta la salud virtual, pasando por los sistemas de información de la salud, la telemedicina o la sanidad electrónica móvil están ya implantadas en numerosos centros hospitalarios y de atención primaria.

A nivel europeo, por ejemplo, el 93 % de los médicos tiene acceso a servicios básicos de Historia Médica Electrónica (EHR) [2]. Y, en general, el desarrollo en los hospitales de la práctica de cuidados sanitarios apoyada en tecnologías de la información y las comunicaciones (eSalud o *eHealth*) ha crecido un 42 % en el período 2010-2013 [18].

Es cierto, sin embargo, que todavía la inversión no es muy alta y que muchos médicos son reacios al uso de las nuevas tecnologías. Así, el 63 % de los hospitales europeos dedica menos de un 3 % de su presupuesto anual a las TIC, o sólo un 10 % de los médicos generalistas europeos está usando en la actualidad las posibilidades de consulta en línea. Además, la conectividad es uno de los temas pendientes de los hospitales, ya que la mayoría de los centros no comparten información [2]. En definitiva, aunque todavía le queda mucho recorrido a la aplicación de las nuevas tecnologías en el campo de la salud, su presencia en hospitales y centros de salud es cada vez más evidente, y pacientes y profesionales se están beneficiando de ello.

Una de las líneas que ha emergido con fuerza y tiene un gran potencial es la que tiene que ver con la práctica médica apoyada en dispositivos móviles tales como teléfonos móviles, PDAs, dispositivos de monitorización, u otros dispositivos inalámbricos. Es lo que se ha dado en llamar Salud móvil (mSalud o *mHealth*), y que también incluye aplicaciones relacionadas con estilo de vida y bienestar que pueden conectarse a dispositivos médicos o sensores (por ejemplo, pulseras o relojes), así como sistemas de guía personal, de información sanitaria y recordatorios de medicamentos mediante SMS y telemedicina de forma inalámbrica [3].

Esta irrupción de mSalud ha venido propiciada por la creciente difusión de los teléfonos móviles y las redes 3G y 4G, así como la disponibilidad de tecnologías de navegación por satélite en dispositivos móviles. Diferentes tipos de sensores y aplicaciones móviles hacen posible la recogida de múltiples datos médicos, fisiológicos, de la actividad diaria, del entorno o del estilo de vida, que pueden ser utilizados tanto para el cuidado de la salud como para actuaciones de investigación.

Así, en el primer caso, las dolencias de los pacientes pueden ser mejor diagnosticadas, dichos pacientes pueden recibir medicación y tratamiento personalizados y más eficaces, y hacerlo en un ambiente más adecuado. En definitiva, la mSalud puede contribuir a conseguir una prestación sanitaria de mayor calidad. En cuanto a la investigación, parece indudable el beneficio que puede tener disponer de datos casi continuos de ciertos parámetros clínicos de una persona y poder relacionarlos con las posibles causas de las deficiencias en su salud.

Al margen de los aspectos estrictamente sanitarios, la mSalud tiene también implicaciones sobre otras múltiples facetas como pueden ser el análisis de la gran cantidad de datos generados, la protección de dichos datos, la fabricación de los diferentes tipos de dispositivos que recojan los datos... Es por ello que a este área relacionada con la salud se están dedicando importantes esfuerzos de financiación, investigación e innovación con el fin de obtener dispositivos y aplicaciones que puedan suponer avances significativos en el campo de la Salud.

Aunque hay muchos de esos productos que resultan bastante accesibles, no suelen ser los que mejores prestaciones ofrecen. Es decir, aquellos dispositivos que recogen un grupo numeroso de parámetros clínicos y con alto valor médico, se fabrican en su mayoría por grandes multinacionales, y por tanto suelen tener costes elevados. Eso implica que sólo unos pocos centros hospitalarios pueden disponer de ellos, y en ocasiones con muy pocas unidades.

Por tanto, sería muy importante que se siga invirtiendo en mSalud para que la demanda de los productos que se vayan obteniendo continúe creciendo y con ello también las empresas que se dedican a diseñar y fabricar dichos productos. Eso contribuirá sin duda a que haya mayor competitividad y repercuta en una bajada de costes, totalmente necesaria para que una inmensa mayoría se aproveche de ellos.

Este Trabajo Fin de Grado es fruto del interés de su autor por el diseño y creación de dispositivos móviles en general, y de aquellos en particular destinados a la monitorización de pacientes. Otro hecho ha motivado también especialmente la realización de este trabajo, y es el interés mostrado desde el Servicio de Geriátrica del Centro Hospitalario Universitario de Albacete por disponer de un dispositivo de monitorización con unas determinadas características. Eso, unido al hecho de que dispositivos con parecida funcionalidad ya existentes tienen un coste demasiado elevado como para que dicho centro pueda disponer de varios de ellos, acabó por impulsar definitivamente el desarrollo de este trabajo.

1.1. Estructura de la memoria

Se indica en esta sección la estructura en la que se ha organizado esta memoria.

- **Capítulo 1: Introducción y Motivación.** En este capítulo se realiza una pequeña introducción y se indican los motivos que llevaron al autor a realizar este proyecto.
- **Capítulo 2: Objetivos y Metodología.** En este capítulo se proponen los objetivos que debe cumplir este proyecto y la metodología seguida para lograr cumplir estos objetivos.
- **Capítulo 3: Estado del Arte.** En este capítulo se explica los orígenes de los dispositivos de monitorización y el estado actual de estos dispositivos.
- **Capítulo 4: Descripción del Prototipo.** En este capítulo se describen todos los detalles del prototipo desarrollado.
- **Capítulo 5: Evaluación.** En este capítulo se evalúa la funcionalidad del prototipo, a través de diversas pruebas de funcionamiento.
- **Capítulo 6: Conclusiones y Trabajo Futuro.** En este capítulo se incluyen las conclusiones finales más importantes del trabajo desarrollado. También se apuntan posibles trabajos futuros y modificaciones realizables al prototipo desarrollado en este trabajo.

Capítulo 2

OBJETIVOS Y METODOLOGÍA

2.1. Objetivos

En este apartado se indican los objetivos de este trabajo. El principal objetivo es el desarrollo de un prototipo funcional de un dispositivo de monitorización, de ahora en adelante mholter, para recoger diversos parámetros médicos y de movimiento. Este objetivo quedará cubierto cuando se alcancen los siguientes sub-objetivos:

1. El prototipo debe ser el resultado de un proceso de diseño previo adecuado a las características que se le requieren.
2. El prototipo debe recoger los siguientes datos:
 - **Electrocardiograma:** Nuestro mholter debe ser capaz de medir los latidos del corazón mediante un electrocardiograma, a partir de ahora ECG.
 - **Oximetría:** Debe ser capaz de realizar una oximetría no invasiva mediante la utilización de una sonda colocada en un dedo o un lóbulo de la oreja del paciente. Mediante esta sonda también se puede obtener el pulso cardiaco por lo que se puede obtener una pulsioximetría.
 - **Presión arterial:** El mholter debe ser capaz de medir la presión arterial ya sea de una manera directa o indirecta.
 - **Temperatura:** Debe ser capaz de medir la temperatura corporal.
 - **Aceleración:** La posición del cuerpo y las fuerzas G aplicadas a este es un aspecto importante para poder saber si el paciente ha sufrido algún tipo de desmayo. El dispositivo debe ser capaz de medir estas magnitudes.
3. El prototipo debe almacenar todos los datos recogidos, de acuerdo con unas especificaciones temporales, para su posterior tratamiento.

4. El prototipo debe tener una cierta autonomía, y en consecuencia se buscará minimizar el consumo.
5. El prototipo debe ser cómodo, no debe suponer problemas de movilidad y debe ser fácil de usar.
6. El prototipo debe tener un coste lo suficientemente bajo como para que pueda plantearse su producción.

2.2. Metodología

En este apartado se indican la metodología y las herramientas usadas para el desarrollo del prototipo. La metodología seguida para lograr el objetivo principal del trabajo está dividida en distintas fases: Objetivos, obtención de información, diseño, simulación, implementación y evaluación. Es necesario completar una fase dada para continuar con la siguiente. Las fases de recopilación de información, diseño, simulación e implementación se repiten para cada elemento del prototipo. Si se encuentra un fallo en la fase de simulación o implementación se vuelve a la fase de diseño de ese elemento para poder subsanar el error desde el principio. Una vez implementado todos los elementos se evalúa el prototipo. Si no se consigue el cumplimiento de algún objetivo se vuelve a la fase de objetivos por si hace falta una revisión de alguno, para después pasar por todas las fases siguientes hasta volver a la de evaluación. En la Figura 2.1 se puede observar un diagrama de flujo con las fases y las diferentes decisiones que se deben tomar en función de los resultados obtenidos en cada fase.

Seguidamente se indican las tareas a realizar en cada una de las distintas fases de la metodología:

Objetivos: En esta fase se definen los objetivos que debe cumplir nuestro prototipo. También se pueden definir unos objetivos secundarios los cuales añadirían valor a nuestro prototipo. Esta fase es la Sección 2.1.

Obtención de información: Esta fase consiste en recopilar información, ya sea técnica o médica, para comprender el funcionamiento de los elementos que formarán parte del prototipo y saber como se pueden obtener los parámetros clínicos necesarios. Esta información nos ayudará a tomar decisiones en la fase de diseño.

Diseño: En este punto con la ayuda de la información recopilada en la fase anterior se diseña tanto el hardware como el software para cumplir los diferentes objetivos.

Simulación: En esta fase se simula mediante software especializado el diseño obtenido en la fase anterior. Esto supone un ahorro de tiempo y coste debido a que podemos detectar fallos en el diseño antes de llevarlos al prototipo.

Implementación: Esta es la fase final, en ella se integra el hardware funcional en el prototipo. También se añade la parte de software correspondiente a cada objetivo.

Evaluación: En este punto se evalúa la funcionalidad de nuestro prototipo y se comprueba si se han cumplido los objetivos. Si no se cumplieran se volvería a la fase de objetivos para revisar las causas.

En relación a las herramientas usadas para la realización de este trabajo, se indican las más importantes:

Arduino IDE: Es un software de código abierto utilizado para transformar el código escrito en C++ y generar un archivo en formato hexadecimal que contiene las instrucciones entendibles por el microcontrolador.

Arduino UNO: Es el modelo de Arduino más utilizado y está basado en un Atmega328p de Atmel. En este trabajo se ha utilizado como tarjeta de adquisición de datos junto al software Labview.

Programador TinyUSB: Es un programador USB capaz de grabar la memoria usando el puerto ICSP del microcontrolador.

Simulador Proteus: Es un software de código privativo capaz de simular el funcionamiento de varios microcontroladores, entre ellos el usado en este trabajo.

Labview 2013: Es un software privativo que permite, mediante una programación visual basada en bloques, generar programas que pueden tratar con señales y datos de una manera sencilla.

Osciloscopio: Es una herramienta muy útil en los laboratorios de electrónica. Su función principal es mostrar de una manera visual una señal eléctrica.

Multímetro: Otra herramienta esencial en un laboratorio de electrónica. Este dispositivo es fruto de la unión de varios dispositivos en uno solo. Este dispositivo puede contener un voltímetro, amperímetro, capacímetro, etc. Según los dispositivos implementados se pueden medir diferentes magnitudes eléctricas.

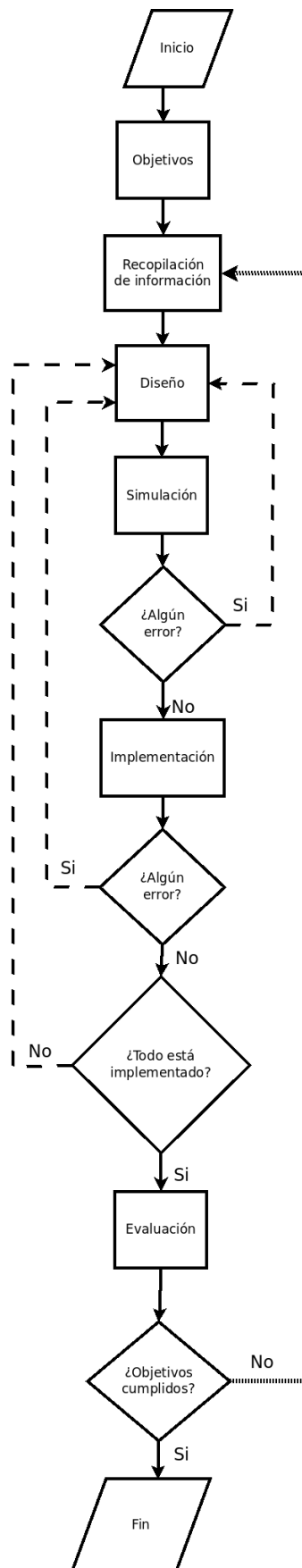


Figura 2.1: Diagrama de flujo de la metodología

Capítulo 3

ESTADO DEL ARTE

3.1. Antecedentes

Los principios de la monitorización se remontan a 1887 cuando Ludwig y Waller fueron los primeros en descubrir el diferencial de potencial eléctrico en el pecho humano, iniciando de esta manera la electrofisiología cardíaca. El electrocardiograma práctico como prueba comenzó en 1893 con el trabajo del galvanómetro de cuerda de Einthoven. El cuerpo del trabajo siguió en este campo, pero el desarrollo de dispositivos que permitían una escritura directa en papel revolucionaron y promovieron el uso de una nueva herramienta de diagnóstico, esta herramienta sería el electrocardiógrafo [10].

En la realización de estas pruebas el paciente debía estar inmóvil o evitar hacer ejercicio debido a que estaba enchufado a una máquina mediante electrodos. Este problema sirvió de inspiración al profesor Norma J. Holter. Ya en los años 40 realiza una serie de estudios en los que consigue estimular preparaciones de músculo por control remoto y conseguir registrar potenciales eléctricos de sistemas biológicos a distancia. Con la colaboración de Joseph A. Gengerelli aprendieron a emitir señales de radio al cerebro de ratas, posteriormente, amplificar la respuesta emitida por el cerebro del animal, transmitiéndola a distancia. En 1947 logran, trabajando ya en humanos, la primera transmisión de un electroencefalograma mientras el sujeto se encontraba realizando ejercicio físico [11; 15].

Las señales emitidas por el músculo cardíaco eran 10 veces más potentes que las señales de cualquier otro músculo, por lo que decide continuar sus investigaciones en el campo de la cardiología. Esta decisión fue tomada no solo por la facilidad que conlleva tratar una señal más fuerte, sino también por el aumento de interés de la comunidad médica por los problemas cardiovasculares [9].

El primer éxito del profesor Holter es un transmisor de radio de 43 Kgs de peso con un alcance de una manzana de casas (Figura 3.1). Sin embargo, con la colaboración de un físico llamado W.R. Glasscock, desarrolla un transmisor más pequeño con un receptor con grabadora que iba incorporado en un maletín que llevaba el propio sujeto de la investigación [6; 7].



Figura 3.1: Primer transmisor del Dr. Holter.

A partir de mediados de los 60, los avances técnicos favorecidos por el descubrimiento del transistor, permitieron disminuir el tamaño del registrador, que ya pudo ser conectado directamente a los electrodos que se colocaban sobre el pecho del paciente. Así se eliminó la necesidad de transmitir la señal por radio y el sistema se hizo totalmente portátil [10; 12]. Por este invento el profesor Norman J. Holter consiguió algo infrecuente en la medicina, prestar su nombre a un método diagnóstico y que este método se convierta en pieza fundamental del diagnóstico en cardiología.

En un principio los registradores permitían 10 horas de grabación en cintas magnéticas, pero el desarrollo comercial del sistema impulsó el desarrollo de nuevos diseños que han conducido a grabadoras de tan solo unos gramos de peso y con capacidad de grabaciones ininterrumpidas de hasta 48 horas en una memoria de estado sólido.

El trabajo del Dr. Holter y sus colaboradores no se paró solamente en el grabador, también idearon un sistema para el análisis rápido de la señal de la ECG grabada en las cintas magnéticas. Desarrollaron en un osciloscopio de rayos catódicos un sistema que permitía la sobreimposición de los complejos cardiacos a una velocidad 60 veces superior a la grabada. La posterior adición de una señal de audio que aumentaba

con los incrementos de la frecuencia cardiaca, completó el sistema que se denominó: AVSEP, del inglés: audiovisual superimposed electrocardiographic presentation [6]. La introducción de este nuevo método de diagnóstico en la medicina amplió enormemente el estudio de los problemas del ritmo cardiaco.



Figura 3.2: Monitor holter de principio de los años 80 con grabación en cinta magnética.

3.2. Estado actual

Los monitores holter actuales son el resultado de la evolución de la tecnología. Hasta mediados de los años 80 los monitores holter grababan en cinta magnética. En la Figura 3.2 podemos ver un ejemplo de un monitor holter con grabación en cinta magnética. En los monitores actuales se graba en memorias de estado sólido. Actualmente disponemos de monitores con capacidad de analizar la ECG y tomar decisiones. Esta evolución también ha creado diferentes tipos de monitores haciendo necesaria una clasificación. El Dr. Igor N. Posokhov en 2012 propone una clasificación de los monitores holter. Esta clasificación está representada en la Tabla 3.1 y varía en función del criterio utilizado para obtenerla.

Actualmente en los hospitales el modelo de monitor holter más usado es el externo, debido a que se puede reutilizar con facilidad. Estos monitores están compuestos de una grabadora digital y unos electrodos externos a la grabadora. En la Figura 3.3 podemos observar un monitor holter modelo DMS-840-3D de la empresa Tecnomed2000. Este monitor es el usado por el servicio de salud de la comunidad autónoma de Andalucía. Dispone de 512 MB de memoria que le permiten grabar 3 canales durante 3 días continuados. El precio de este monitor es de 1.650 € sin el software de análisis.

Otro modelo de monitor holter es el h3+ de Mortara Instruments, en la Figura 3.4. Este monitor puede grabar 2 ó 3 canales durante 48 horas. También dispone de una

CRITERIO	CLASIFICACIÓN
peridodo de monitoriazación	• 24h • 48h • de semanas a meses
Tipo de grabación	• Grabación continua • Grabación intermitente: a) Grabación en bucle b) Grabación de eventos
Localización del dispositivo durante la grabación	• Grabador implatado • Grabador portable
Número de canales	• 1 • 2 • 3 • 12
Características de los electrodos	• Externos • Intracardiacos • Electrodo en el cuerpo del dispositivo

Tabla 3.1: Clasificación de los monitores Holter



Figura 3.3: Monitor holter DMS-840-3D de Tecnomed2000.

característica adicional que es la detección automática de marcapasos. El precio de este monitor es de 1.200 €.

Otro tipo de monitores holter son lo de tipo de grabación de eventos. Estos monitores solamente graban cuando se produce un evento o el paciente lo activa. Normalmente graban unos minutos antes de la activación y unos minutos posteriores. Aquí es donde muchos expertos no coinciden ya que algunos dicen que graban continuamente y otros durante solamente unos eventos. La realidad es que el monitor está grabando cons-



Figura 3.4: Monitor holter h3+ de Mortara Instruments.

tanamente, pero si no se ha producido ningún evento va desechando los datos más antiguos debido a que su memoria es muy limitada y solamente puede grabar unos pocos minutos. Un ejemplo de este tipo de monitor es el modelo Merlin de la marca Mediatech, en la Figura 3.5, el cual tiene forma de reloj de pulsera y se activa poniendo la mano sobre él. En precio de estos dispositivos ronda los 800 €. La desventaja de estos dispositivos es que se depende del paciente para que active el monitor, pudiéndose dar el caso de que se obvie la activación del dispositivo perdiendo la captura del evento.



Figura 3.5: Monitor holter Merlin de Mediatech.

La siguiente escala en la evolución de los monitores holter nos lleva a los monitores holter implantables. Este tipo de monitores se implantan quirúrgicamente en el paciente. Un ejemplo de un holter implantable lo podemos observar en la Figura 3.6. Al precio del monitor hay que sumar el precio de la intervención quirúrgica. Este tipo de monitores son grabadores de eventos debido a que su memoria es muy limitada. La activación de estos dispositivos puede ser manual o automática cuando se detecta una anomalía. La lectura de los datos se realiza mediante medios inalámbricos, por lo que no es necesario la extracción para la lectura. La vida de estos dispositivos puede llegar a los 3 años, que es la duración de la batería. Después de esto es necesaria su extracción debido a que la batería no es recargable ni intercambiable. Un ejemplo de este tipo de

monitores es el Reveal plus de Medtronic usado en el servicio de salud de la Junta de Andalucía (Figura 3.7), con un precio de 3.600 € sin contar con la intervención quirúrgica [17].

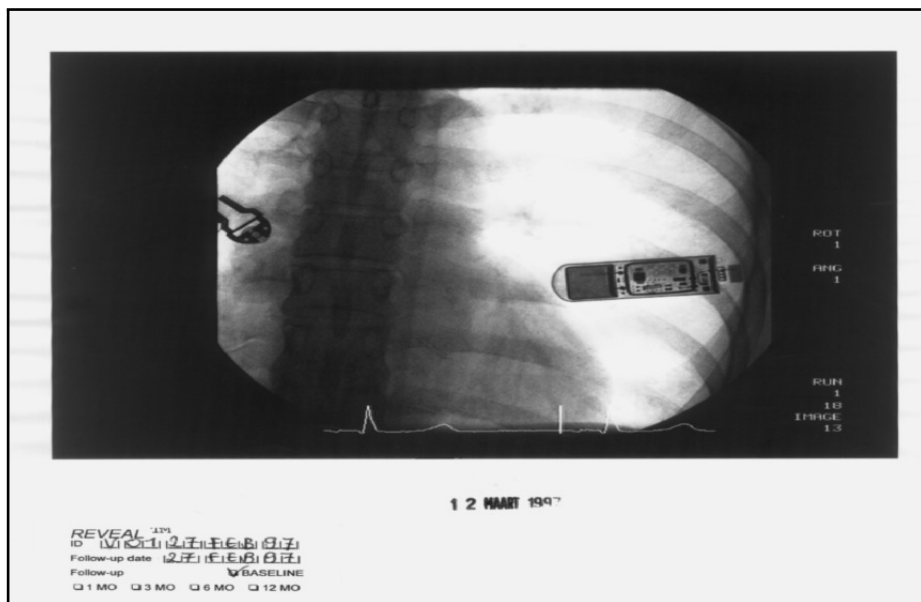


Figura 3.6: Radiografía de un monitor holter implantable.



Figura 3.7: Monitor holter Reveal plus de Medtronic.

También existen muchos otros monitores holter no comerciales, la mayoría proyectos “open source”. Entre ellos destacaremos Mobilecg¹, en la Figura 3.8 y E-health², en la Figura 3.9. El primero no es un monitor en sí pero es capaz de leer 12 canales y a unos niveles de ruido muy bajos. Este proyecto es además de código abierto, es “open hardware” por lo que podemos conseguir todos los esquemas y software relacionado con este proyecto. El precio de este dispositivo es de 299 \$.

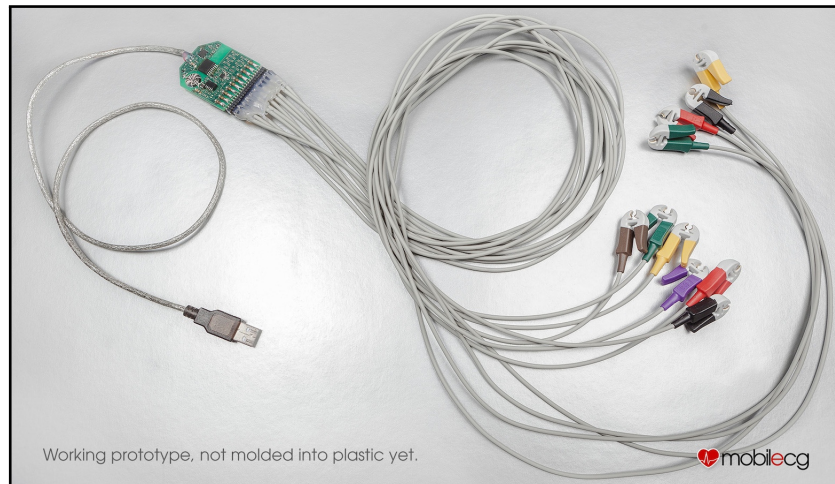


Figura 3.8: Proyecto Mobilecg.

El segundo es un shield para Arduino el cual nos permite recoger varias magnitudes médicas, entre las que se encuentran la ECG. El proyecto es libre, tanto el software como el hardware. El precio si optamos solamente al shield es de 75 €, pero si queremos el kit completo el precio es de 450 €.

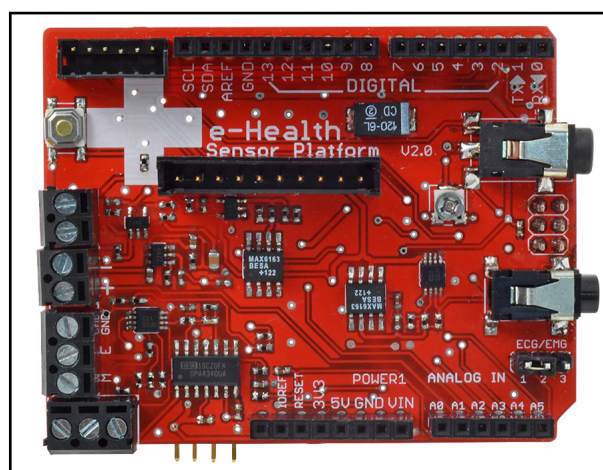


Figura 3.9: E-Healt Arduino Shield.

¹<http://mobilecg.hu/>

²<http://www.cooking-hacks.com/>

Capítulo 4

DESCRIPCIÓN DEL PROTOTIPO

El prototipo desarrollado está compuesto por varios circuitos independientes entre sí. Cada uno de esos circuitos está diseñado para obtener un dato biométrico distinto, por lo que se pueden aplicar la metodología explicada en la sección [2.2](#) por separado.

Los distintos circuitos son: ECG, oximetría, temperatura, aceleración/giroscopio, RTC y almacenamiento de datos. A cada uno de estos circuitos se les aplica la metodología explicada, pero existen otros elementos que no son circuitos aunque se les puede aplicar algunas fases de la metodología. Estos elementos son el software de representación de los datos almacenados y la medición de la presión sanguínea.

Otro apartado importante que no debemos olvidar es la elección del microcontrolador a usar para la realización del prototipo. En este caso se va a usar una placa Arduino Mega 2560, provista de un microcontrolador ATmega2560. Este microcontrolador dispone de suficiente memoria para almacenar todo el software que implementemos. Las características principales de este microcontrolador las podemos encontrar en la Tabla [4.1](#). Un aspecto importante que nos ha ayudado a decantarnos por esta plataforma es la cantidad de librerías disponibles en la red ofrecidas libremente por la comunidad de usuarios de esta plataforma.

A continuación se incluyen los detalles más significativos de cada uno de los parámetros considerados. Para cada uno de ellos se describe su diseño, las características de la simulación y los aspectos relevantes de la implementación.

Voltaje operacional	5V
Pines digitales de E/S	54
Salidas PWM	15
Entradas analógicas	16
Corriente suministrada por un pin	40 mA
Memoria flash	256 KB
Memoria usada para el bootloader	8 KB
SRAM	8 KB
EEPROM	4 KB
Velocidad del reloj	16 Mhz
Número de puertos UART	4
Número de interrupciones externas	6
Número de puertos SPI	1
Número de puertos de comunicación paralela	1

Tabla 4.1: Principales características del ATmega2560.

4.1. ECG

El corazón es el órgano principal del aparato circulatorio del ser humano. Es un músculo hueco y piramidal situado en la cavidad torácica. Funciona como una bomba aspirante e impelente, impulsando la sangre a todo el cuerpo. El corazón impulsa la sangre mediante los movimientos de sístole y diástole. Se denomina **sístole** a la contracción del corazón (ya sea de una aurícula o de un ventrículo) para expulsar la sangre hacia los tejidos. Se denomina diástole a la relajación del corazón para recibir la sangre procedente de los tejidos [16].

Un ciclo cardiaco está formado por una fase de relajación y llenado ventricular (diástole) seguida de una fase de contracción y vaciado ventricular (sístole). El corazón al realizar este ciclo genera un impulso eléctrico el cual usaremos para detectar la actividad del corazón. La prueba médica que utiliza este efecto para detectar problemas cardiacos es el electrocardiograma o ECG. Podemos ver una representación gráfica de esta actividad en la Figura 4.1, donde en rojo está representada la actividad eléctrica dentro del corazón y en azul la onda leída por electrocardiograma [4].

La onda generada por el corazón está compuesta de 6 fases y dos intervalos. De estas fases la única que nos interesa para medir el pulso es el complejo QRS, el cual es más potente que cualquier otra fase debido a que es causada por la contracción de los ventrículos derecho e izquierdo implicando así mucha más masa muscular, produciendo

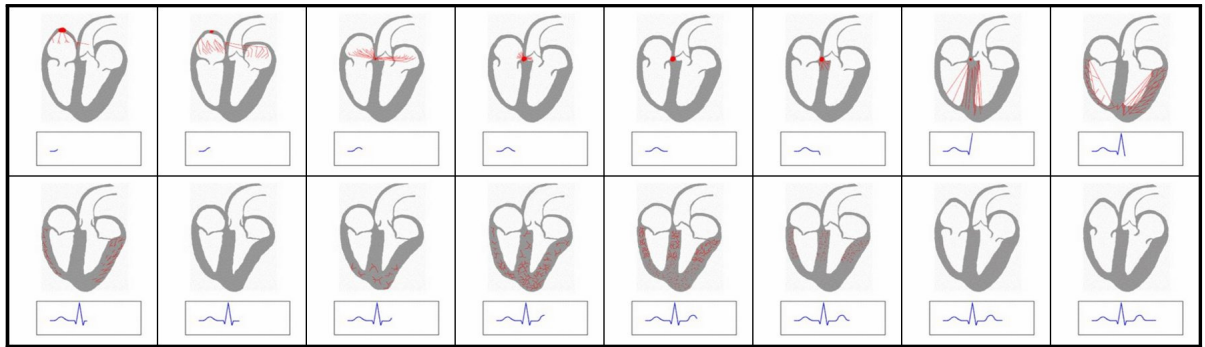


Figura 4.1: Representación gráfica de la actividad eléctrica del corazón.

mayor deflexión en el electrocardiograma. La duración normal de esta fase es de 60 a 100 milisegundos [1].

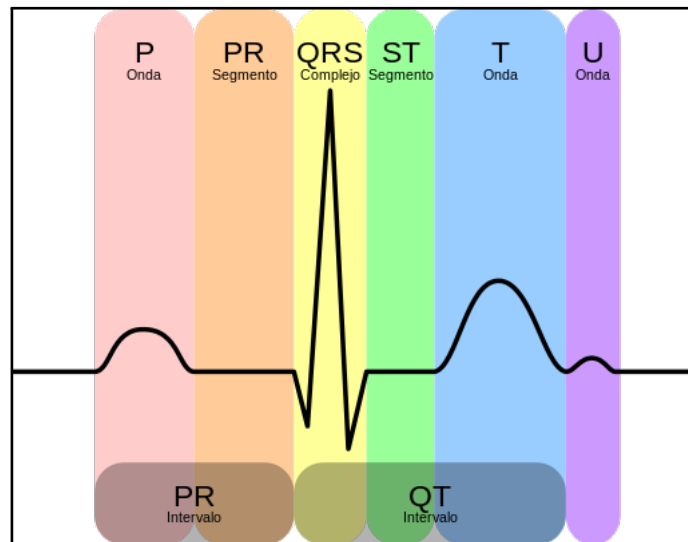


Figura 4.2: Distintas fases del electrocardiograma.

4.1.1. Diseño

La onda eléctrica generada por el corazón no tiene un voltaje mayor a 3,5 mV, por lo que necesitaremos amplificar la señal. También debemos filtrar la señal debido a que se pueden introducir otras señales y sus armónicos, por ejemplo las ondas generadas por la red eléctrica de las instalaciones. A este proceso (amplificación y filtrado) se le denomina acondicionamiento de la señal en la adquisición de señales analógicas para posteriormente hacer una conversión a digital.

Para el diseño nos basaremos en el proyecto de OpenHardware e-Health de Cooking Hacks¹. El corazón de este proyecto son dos amplificadores instrumentales de National Instruments, en concreto, los modelos INA321 y OPA4340. También dispone de un filtro de paso alto, el cual quita las frecuencias que no nos interesan.

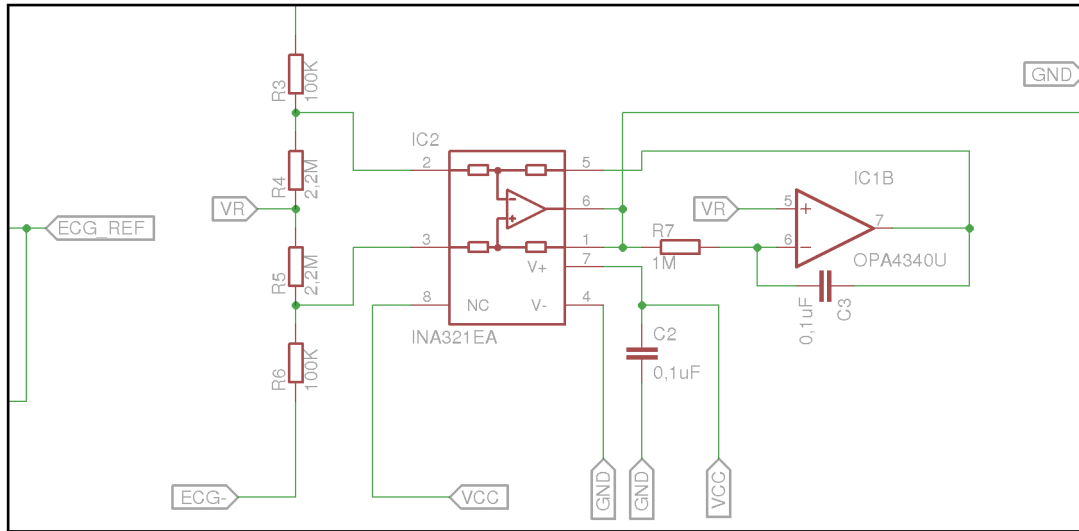


Figura 4.3: Detalle del INA321 en el esquema.

Para la parte de software se ha usado el convertor A/D interno del Arduino, el cual es capaz de convertir a una velocidad de 1 Ksamples por segundo. La conversión se programa para que realice la conversión a una frecuencia determinada de una manera automática y usando las interrupciones que saltan al acabar la conversión. Este método de realizar las conversiones es el más efectivo debido a que deja la CPU libre mientras se realiza la conversión.

La frecuencia a la cual muestreemos tiene que ser mayor que la frecuencia dada por el teorema de Nyquist. Este teorema fundamental indica que si la frecuencia más alta contenida en una señal analógica $x_a(t)$, es $F_{max} = B$, y la señal se muestrea a una tasa $F_s > 2F_{max} \equiv 2B$, entonces $x_a(t)$ se puede recuperar totalmente a partir de sus muestras.

La frecuencia que nos interesa detectar viene dada por el tiempo del complejo QRS, que puede tener una duración de 60 a 100 milisegundos. Es decir, puede tener una frecuencia aproximada de 10 Hz a 17 Hz, por lo que tendremos que muestrear a una velocidad de 34 Hz o más.

¹<http://www.cooking-hacks.com>

Debido a que el microcontrolador sólo dispone de un conversor A/D, necesitaremos multiplexarlo para poder realizar la conversión en todas sus entradas analógicas, al haber más sensores que necesitan usar el conversor (oximetría y temperatura). Es decir, deberemos repartir el tiempo de uso del conversor entre las entradas siempre cumpliendo el teorema de Nyquist. Para ello debemos saber los tiempos críticos para cada medida. Para el caso de la oximetría, si deseamos detectar el pulso, la frecuencia que tiene la señal es la generada por el pulso humano, siendo éste de 40 ppm a 250 ppm en sus casos más extremos, lo que nos da un 1 latido cada 240 milisegundos (en el peor caso), es decir, una frecuencia de 4,17 Hz. Por lo tanto, debemos tomar muestras cada 120 milisegundos. Por último, la medición de la temperatura corporal es otra medida que usa el conversor A/D. Esta medida no varía rápidamente en el tiempo por lo que no es necesario una frecuencia muy alta de muestreo. Se ha considerado que una muestra por segundo es suficiente para esta medida.

Como podemos ver, la medida que nos condiciona el tiempo de las otras medidas es la ECG, la cual debemos muestrear a una velocidad de 34 Hz como mínimo, lo que obliga a tomar una muestra cada 29 milisegundos o menos.

En nuestro caso usaremos un pre-escalado interno, el cual divide el reloj del Arduino entre un determinado valor guardado en un registro. Este valor puede ser de 8, 16, 64, 128. En nuestro caso es de 128, pues cada 128 pulsos del reloj del Arduino se generará una conversión de analógico a digital. Sabemos que el reloj del Arduino funciona a 16 Mhz, si lo dividimos entre 128 obtenemos 125 Khz. Pero debemos multiplexar el conversor para poder usar las distintas entradas por lo que la conversión de la ECG se realizará cada 2 para ir alternandola con la oximetría, por lo que la frecuencia final de muestreo es de 62,5 KHz, mucho mas superior que los 34 Hz necesarios [13]. Los puntos necesarios para tomar las medidas del electro cardiograma son 3: +, - y neutro o referencia. La posición de estos puntos se pueden consultar en la Figura 4.4.

El flujo de datos generado por la medición de esta señal es de 2 bytes por segundo, debido a un número de tipo entero, pero al pasarlo a una representación en caracteres ASCII, en el peor de los casos serán 3 dígitos, debido a que el pulso máximo humano tiene 3 cifras, por lo que se guardarán en la memoria de almacenamiento 3 bytes por segundo. El consumo de potencia de este circuito en el peor de los casos es de 25mW, que es resultado de la suma de los circuitos integrados y toda la electrónica restante.

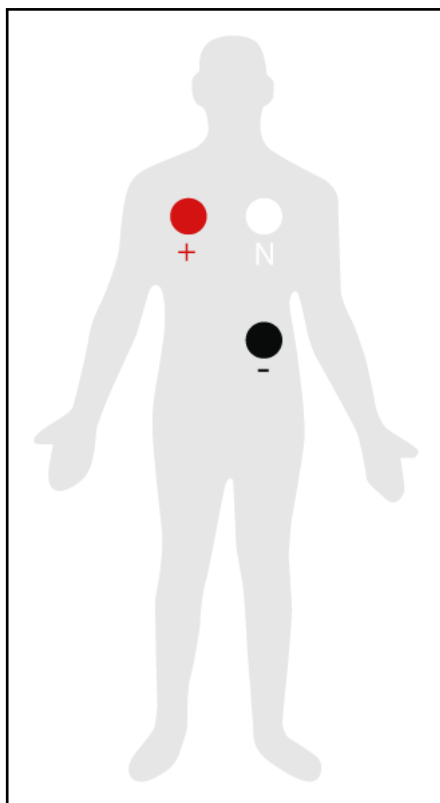


Figura 4.4: Posición de los puntos de la ECG.

4.1.2. Simulación

Para la simulación nos basaremos en el software Proteus, el cual dispone de un generador de funciones que usaremos para simular los latidos del corazón. Un Arduino MEGA se encarga de realizar los cálculos y un LCD de 16 caracteres por 2 líneas para poder representar de una forma gráfica el pulso actual.

Tanto el proyecto de Proteus como el código fuente de la prueba se adjuntan en el CD anexo a esta memoria.

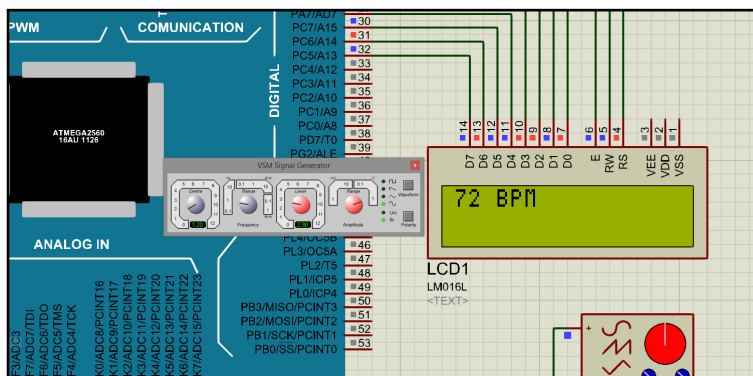


Figura 4.5: Simulación del funcionamiento de la ECG en Proteus.

En la Figura 4.5 se puede observar una captura de la pantalla del software Proteus con la simulación ejecutándose y pudiendo comprobar que si variamos el contador del generador de funciones este cambio se refleja en nuestro LCD.

4.1.3. Implementación

La implementación en el prototipo final se ha realizado en una placa perforada, en la cual las pistas son realizadas mediante la técnica Manhattan, montando integrados como el INA321 y el OPA4340 sobre adaptadores que transforman sus conexiones SMD a PDIP.

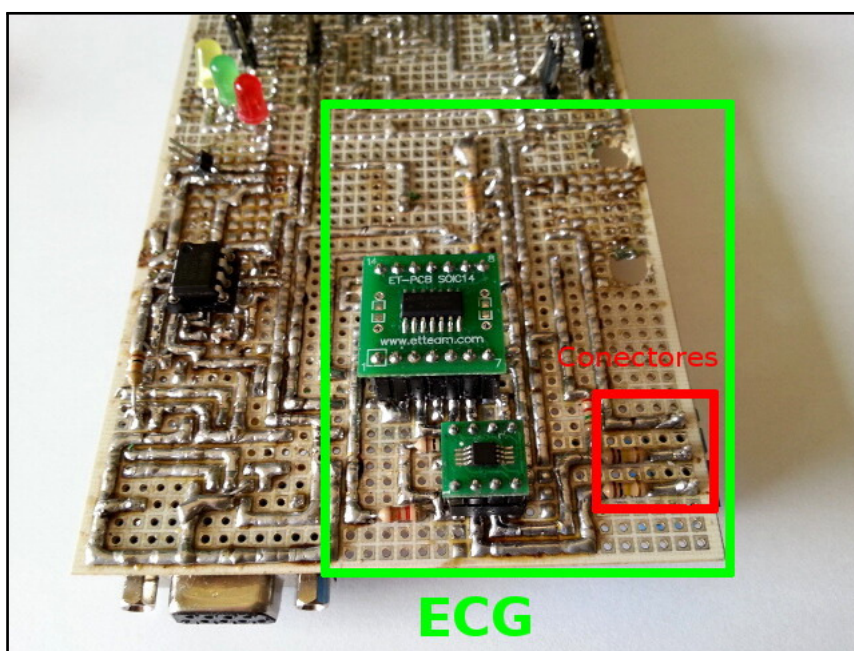


Figura 4.6: Implementación del circuito de la ECG en el prototipo final.

En la Figura 4.6 se observan los conectores a los cuales van enchufados los parches para poder tomar el valor de la ECG. Para confirmar su funcionamiento se ha usado un osciloscopio del laboratorio, en la Figura 4.7 se observa la captura generada por el osciloscopio al probar el funcionamiento de la nueva parte implementada.

4.2. Oximetría

La Oximetría de pulso (SpO_2) es la estimación de la saturación arterial de oxígeno (SaO_2) en forma no invasiva, usando dos emisores de luz y un receptor colocados a través de un lecho capilar pulsátil.

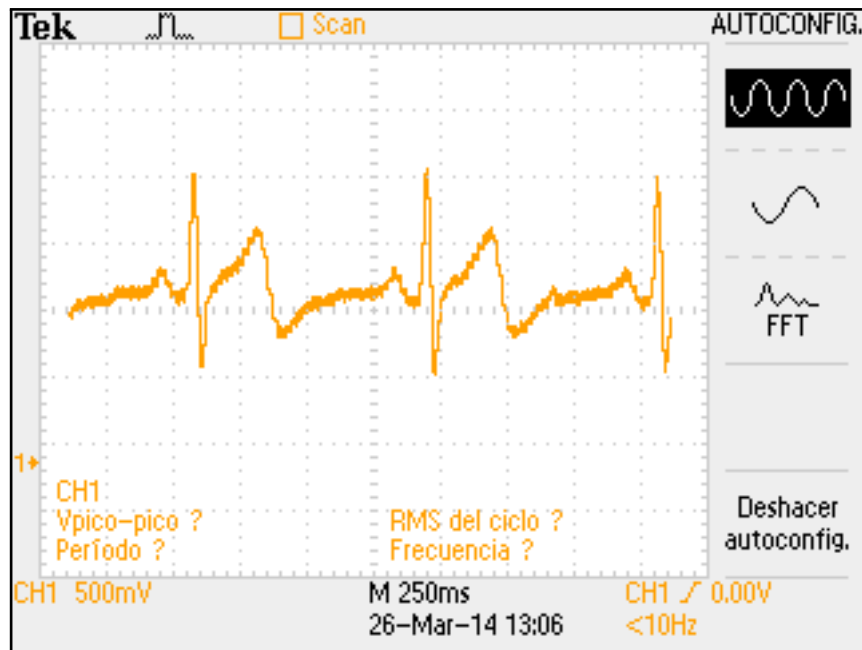


Figura 4.7: Captura generada por el osciloscopio para la ECG.

El método que se utiliza para su medición se denomina Pulsioximetría, que permite determinar el porcentaje de saturación de oxígeno de la hemoglobina en sangre de un paciente con ayuda de métodos fotoeléctricos.

Para realizar esta técnica se coloca el pulsioxímetro en una parte del cuerpo que sea relativamente translúcida y tenga un buen flujo sanguíneo, por ejemplo los dedos de la mano o del pie (en caso de lactantes) o el lóbulo de la oreja. El pulsioxímetro emite luces con diferente longitud de onda, roja e infrarroja, que pasan secuencialmente desde un emisor hasta un fotodetector a través del paciente. Se mide la absorbancia² de cada longitud de onda causada por la sangre arterial (componente pulsátil), excluyendo sangre venosa, piel, huesos, músculos, grasa. Con estos datos será posible calcular la saturación de oxígeno en sangre [21].

El pulsioxímetro se sirve de diferentes longitudes de onda de la luz para diferenciar las moléculas de un fluido en función de sus propiedades. El objetivo es diferenciar la hemoglobina oxigenada (oxihemoglobina) y la desoxigenada (desoxiemoglobina) en sangre pulsátil.

La oxihemoglobina y la desoxihemoglobina son dos moléculas distintas que absorben y reflejan distintas longitudes de onda de luz. La oxihemoglobina absorbe la luz de espectro infrarrojo, y transmite las longitudes de onda del espectro rojo a través de

²Medida de la atenuación de una radiación al atravesar una sustancia, que se expresa como el logaritmo de la relación entre la intensidad saliente y la entrante.

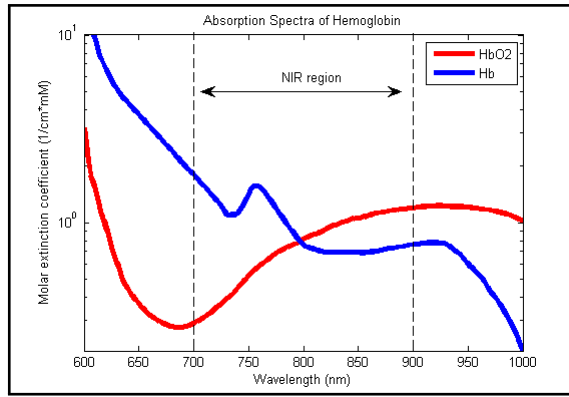


Figura 4.8: Extinción (absorción) de la luz por la hemoglobina oxigenada y reducida [14].

ella. Por el contrario, la desoxihemoglobina absorbe la luz del espectro rojo, y deja pasar la del infrarrojo. El espectrofotómetro del pulsioxímetro emite luz en el espectro rojo (660 nanómetros) y en el infrarrojo (920 nanómetros). Los dos tipos de onda se transmiten a través de un lecho vascular pulsátil y se detectan en el extremo opuesto con un fotodetector (que también podría recibir la luz reflejada y no la recibida, en cuyo caso se situaría en el mismo extremo de emisión), midiendo la cantidad de luz de cada longitud de onda recibida y transmitiendo la información mediante impulso eléctrico. El procesador calcula, para cada longitud de onda, la diferencia entre la luz emitida y la recibida, indicándonos la cantidad de luz que ha absorbido la sangre pulsátil. Este dato nos sirve para calcular el ratio de oxihemoglobina y desoxihemoglobina en circulación o, lo que es lo mismo, la saturación de la hemoglobina, mediante la siguiente fórmula:

$$SaO_2 = HbO_2 / (HbO_2 + Hb) \quad (4.1)$$

donde HbO₂ es la hemoglobina oxigenada, oxihemoglobina, y Hb es la hemoglobina desoxigenada, desoxihemoglobina [14; 19].

4.2.1. Diseño

Para tomar las medidas se decide usar una sonda de medición Nellcor DS-100A, ya que la conexión y pinout son casi un estándar seguido por la gran mayoría de fabricantes. Esta sonda contiene en su interior dos diodos LED, uno trabaja en 660 nanómetros (luz roja visible) y otro a 920 nanómetros (Infrarrojos). También dispone de un foto-transistor el cual es saturado en su base mediante luz, produciendo en su emisor una señal semejante a la de la base.

La señal proporcionada por el foto-transistor es muy pequeña por lo que debemos filtrarla y amplificarla. En el proceso de filtrado se quitan las frecuencias que no nos interesan y que pueden interferir en la medición, por ejemplo, la frecuencia generada por la red eléctrica. En el proceso de amplificado la señal se aumenta a los niveles en los que el microcontrolador pueda trabajar con la señal.

El corazón de este circuito es un amplificador operacional de Texas Instruments, concretamente el modelo LM358. Este es un amplificador asimétrico debido a su alimentación que es de 5V.

La salida del circuito se conecta a una de las entradas analógicas del Arduino y como se ha explicado en la Sección 4.1.1, se muestrea a una frecuencia de 20 milisegundos alternando los canales del infrarrojos y del visible. Un valor intermedio, conocido como ratio normalizado R, se calcula desde las señales leídas de la sonda pulsioximétrica.

$$R = \frac{\left(\frac{AC_{rms660nm}}{DC_{660nm}}\right)}{\left(\frac{AC_{rms940nm}}{DC_{940nm}}\right)} \quad (4.2)$$

Este valor representa una relación de reducción de la hemoglobina arterial oxigenada. Usando este valor, y en base a unos datos empíricos, se puede calcular un valor de saturación de oxígeno.

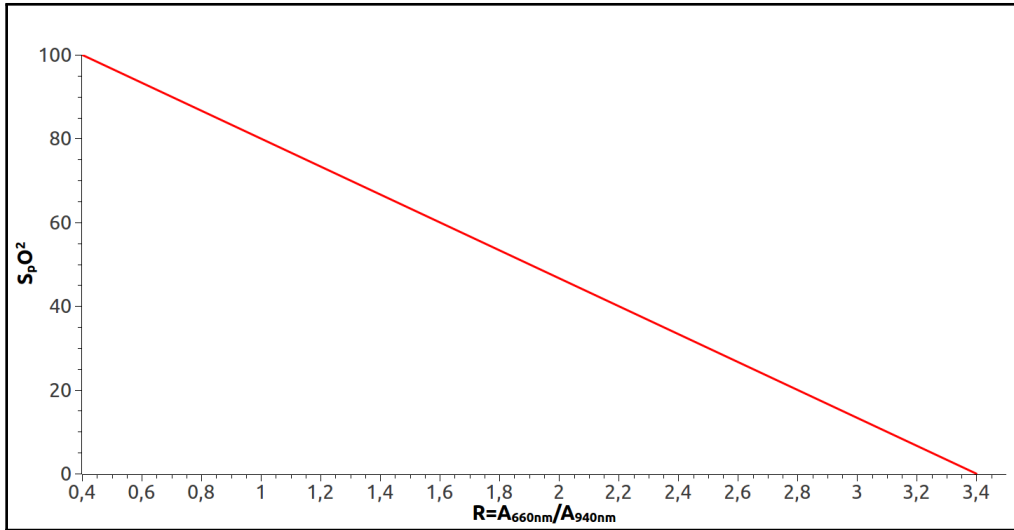


Figura 4.9: Curva empírica sobre la saturación del oxígeno (vs. R) [20].

Esta curva empírica puede ser aproximada por una ecuación lineal [21]:

$$S_pO_2 = 110 - 25 \times R \quad (4.3)$$

El consumo de este circuito es de 125 mW, resultado de la suma del amplificador, la sonda y el resto de la electrónica.

4.2.2. Simulación

Para la simulación se usó el software Proteus. En esta simulación se han añadido dos generadores de frecuencia para simular la onda generada por la luz infrarroja y la luz roja visible. Se han puesto a una frecuencia de 1,2 Hz y con una amplitud diferente para simular la absorción de la luz. En la Figura 4.10 podemos ver una captura de la simulación. En el LCD virtual se puede observar el pulso y la oximetría.

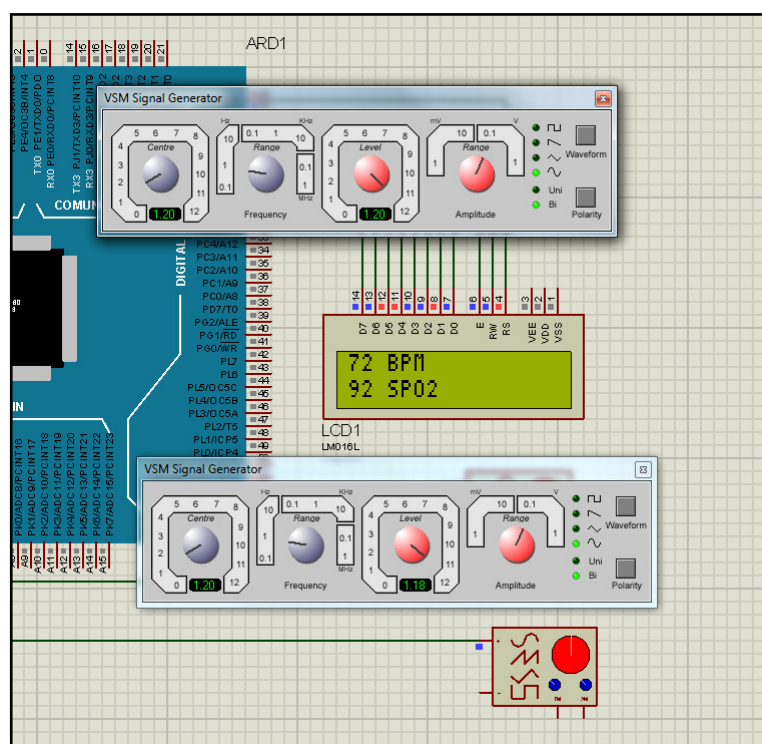


Figura 4.10: Simulación de la oximetría con el software Proteus.

4.2.3. Implementación

La implementación en el prototipo final se ha realizado en una placa perforada, en la cual las pistas se realizan mediante la técnica Manhattan, montando integrados sobre zocalos PDIP.

En la Figura 4.11 se observa el conector al cual va enchufado la sonda de pulsioximetría. Para comprobar la parte recién implantada podemos usar el software Labview 2013 junto a un Arduino UNO preparado para realizar la adquisición de datos.

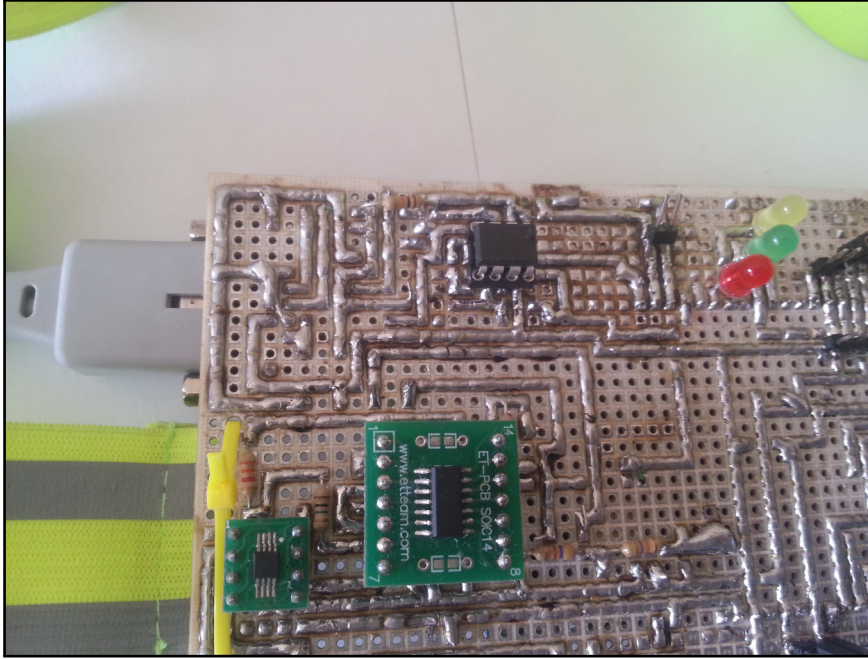


Figura 4.11: Implementación del circuito del oxímetro en el prototipo final.

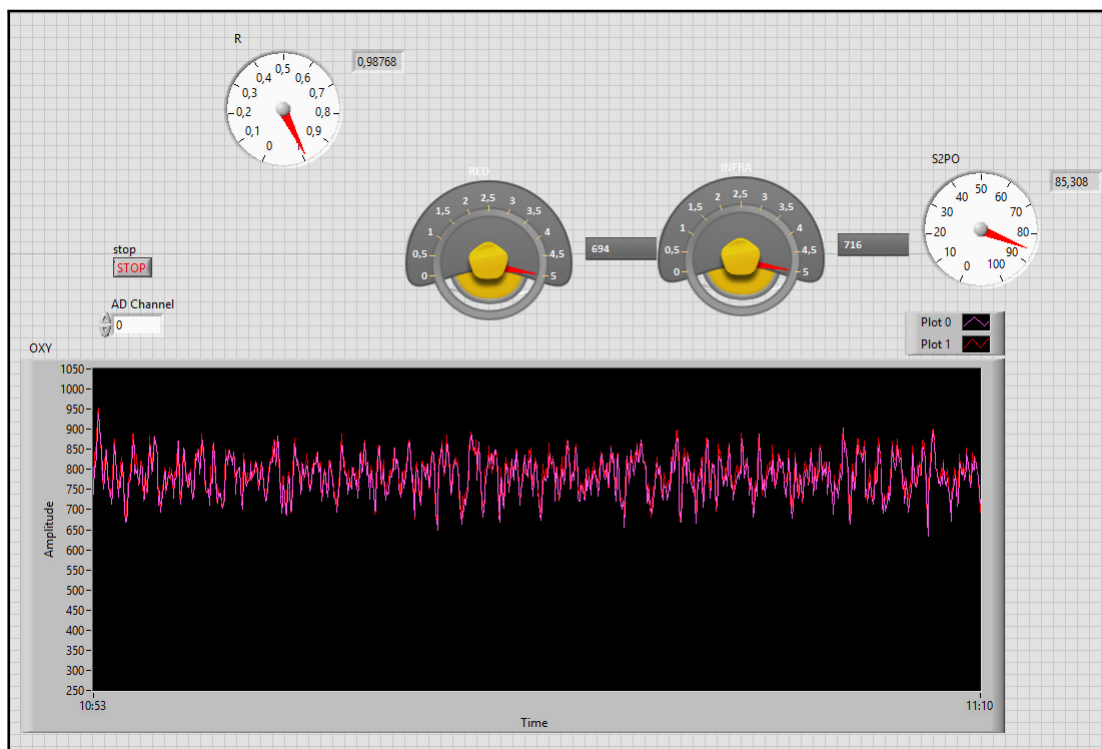


Figura 4.12: Prueba de concepto del oxímetro en hardware real y Labview.

Como se observa en la Figura 4.12 la nueva parte implementada funciona correctamente.

4.3. Presión sanguínea

La presión arterial está compuesta por dos presiones. Una presión sistólica y otra diastólica. La presión sistólica es la presión máxima que se alcanza en la sístole. Esta depende fundamentalmente del débito sistólico, la volemia y la distensibilidad de la aorta y las grandes arterias. La presión diastólica es la mínima presión de la sangre contra las arterias y ocurre durante la diástole. Depende fundamentalmente de la resistencia vascular periférica.

La presión de pulso es la diferencia entre la presión sistólica y la diastólica. La presión arterial varía en las personas a lo largo de las 24 horas. Los factores que influyen son las emociones, actividad física, presencia de dolor, estimulantes como café, tabaco, drogas, etc.

Una vez se han obtenido los valores tanto de la presión sistólica como la diastólica, esos valores se representan separados con una barra entre los valores de las dos presiones y seguidos de mmHg (milímetros de mercurio). Por ejemplo: para una presión sistólica de 120 y diastólica de 80, su representación sería: 120/80 mmHg.

La medición de la presión arterial habitualmente se realiza con un esfigmomanómetro. Los más usados son los de mercurio y los de tipo aneroides. Constan de un sistema para ejercer presión alrededor del brazo y una escala que permite conocer la presión. Los esfigmomanómetros de mercurio son más fiables en su calibración. Los aneroides, que registran la presión mediante un reloj, son menos fiables, son más fáciles de transportar, pero con el tiempo se pueden descalibrar [8].

La presión arterial conviene medirla en el brazo, estando el paciente sentado o acostado, cómodo y relajado. Debe haber descansado unos cinco minutos y no haber consumido café o haber fumado en los 30 minutos anteriores.

La presión arterial se expresa con la presión sistólica y la diastólica. Por ejemplo, una presión de 120/80 mmHg, significa que la sistólica es de 120 mmHg y la diastólica es de 80 mmHg. Además del registro numérico, se debe especificar en qué parte del cuerpo se tomó la presión y en qué posición se encontraba el paciente. Un registro de 120/80/70 mmHg significaría que a los 80 mmHg los ruidos se atenuaron y que a los 70 mmHg se dejaron de escuchar, siendo este último valor la presión diastólica.

Los valores normales de la presión arterial son:

- Presión sistólica: entre 100 y 140 mmHg (lo ideal sería tener una presión sistólica que no superara los 120 mmHg, o a lo sumo, los 130 mmHg).
- Presión diastólica: entre 60 y 90 mmHg (lo ideal sería tener una presión diastólica por debajo de los 90 mmHg).

Se considera que un paciente está comenzando a ser hipertenso cuando su registro es igual o superior a 140/90 mmHg. Algunas personas, especialmente mujeres jóvenes, tienen presiones que normalmente son bajas (100/60 mmHg o incluso menores). En otras situaciones, la presión baja es una manifestación de Shock o colapso circulatorio, pero en estos casos, se prestan signos de mala perfusión tisular (compromiso de conciencia, extremidades frías, diuresis escasa, ...).

Otra manera de medir la presión arterial es de manera indirecta mediante PTT (Pulse Transit Time). Mediante esta técnica se usa el tiempo que tarda en transmitirse el pulso por el sistema circulatorio (PTT) y la velocidad de la onda del pulso (pulse wave velocity, PWV). Estos dos valores tienen una correlación con la medición indirecta de la presión sanguínea [5].

4.3.1. Diseño

En este punto debemos valorar la elección en nuestro diseño debido a que si optamos por diseñar la lectura de la presión sanguínea de manera directa, mediante un esfigmomanómetro, deberíamos implementar un manguito y una bomba de aire. Dicha bomba de aire nos comprometería nuestro dispositivo debido a que su consumo sería excesivo y añadiría ruido al sistema.

Si se elige la forma de medir indirectamente, usando los datos del ECG y el oxímetro para medir el PTT. El consumo no varía, pero para calcular el PWV, como se puede observar en la Ecuación 4.4, son necesarios unos parámetros que varían según el paciente, como son la altura (H) y el factor de correlación del cuerpo (BDC).

$$PWV \text{ (cm/ms)} = \frac{BCD \times H \text{ (cm)}}{PTT \text{ (ms)}} \quad (4.4)$$

Esto conllevaría a cargar esos datos en el dispositivo cada vez que se usa. Se va optar por guardar el PTT cada segundo, para luego a la hora de realizar la presentación de los datos guardados del dispositivo se realicen los cálculos ya que el software externo puede vincularse de una forma sencilla con el historial médico del paciente.

El flujo de datos generado por esta medición es de 32 bits (4 bytes) pero al pasarlo a una representación en caracteres ASCII éstos pueden llegar, en el peor de los casos, a 10 caracteres, por lo que se guardarán en el dispositivo de almacenamiento 10 bytes por segundo.

4.4. Temperatura

La temperatura corporal es la medida relativa de calor o frío asociado al metabolismo del cuerpo humano y su función es mantener activos los procesos biológicos. Esta temperatura varía según la persona, la edad, la actividad y el momento del día y normalmente cambia a lo largo de la vida.

La temperatura corporal depende de las condiciones de temperatura y de actividad física, ya que de la energía total liberada durante el metabolismo se emplea aproximadamente una quinta parte en el trabajo y lo demás se libera en forma de calor; este calor debe ser disipado para mantener la condiciones de temperatura adecuadas en el cuerpo humano [16].

Hay dos tipos de temperaturas, la **temperatura central** (núcleo: cerebro, grandes vasos, vísceras, músculo profundo, sangre), que se mantiene constante. La **temperatura periférica** (piel, mucosas, extremidades, etc) que es variable.

La temperatura normal del cuerpo de una persona varía dependiendo de su género, actividad reciente, consumo de alimentos y líquidos, hora del día y, en las mujeres, de la fase del ciclo menstrual en el que se encuentren.

La temperatura corporal, de acuerdo a la Asociación Médica, puede oscilar entre 36.5°C y 37.2°C. Aumenta lentamente a lo largo del día, hasta alcanzar un máximo de 37.2°C (algo más en algunas personas) entre las 6 y las 10 de la tarde, y desciende lentamente hasta un mínimo entre las 2 y las 4 de la madrugada. La temperatura con ejercicio intenso suele tener grandes subidas.

El organismo humano es homeotermo, lo que significa que para mantener de forma constante la temperatura corporal dentro de unos límites tan estrechos (de 36 a 37°C), necesita, ante elevaciones importantes de la temperatura exterior, poner en marcha una serie de mecanismos de termorregulación.

Para mantener constante esta temperatura, existen múltiples mecanismos. Estos están controlados por el hipotálamo, que es donde se centraliza el control de la temperatura. El hipotálamo se encarga de regular las propiedades del medio interno, como la temperatura. El hipotálamo funciona de forma parecida al termostato de una casa. Cuando la temperatura del cuerpo es menor que la apropiada, éste se pone en marcha hasta que la temperatura llega a la indicada. El hipotálamo mide la temperatura en el propio hipotálamo, en cierta región del hipotálamo hay neuronas que son sensibles a la temperatura. Además el hipotálamo recibe información de temperaturas de diferentes partes del cuerpo, sobre todo de la temperatura de la piel, y esta información le llega procedente de fibras nerviosas sensoriales sensibles a la temperatura. El hipotálamo compara la temperatura en el hipotálamo y en la piel con el valor de 37°C , si la temperatura corporal es mayor de 37°C pone en marcha mecanismos para que disminuya, si es menor de 37°C hace que ascienda. Cuando existe una discrepancia entre la temperatura central en el hipotálamo y la temperatura en la piel, toma como referencia la temperatura central [4].

La información cutánea permite al hipotálamo anticiparse a los cambios. Si la temperatura es baja, quiere decir que estamos en un ambiente frío, y que conviene conservar el calor, así que el hipotálamo ponen en marcha los mecanismos correspondientes antes de que la temperatura en el interior del organismo empiece a cambiar. Por otro lado el cerebro se daña fácilmente con los cambios de temperatura, por lo que si la temperatura en el hipotálamo empieza a aumentar, se ponen en marcha inmediatamente mecanismos para bajarla, sin importar la temperatura cutánea.

En general, en los humanos están mejor desarrollados los mecanismos para resistir al calor que aquellos para resistir el frío, lo que probablemente refleja el hecho de que la especie evolucionó en un clima cálido.

Puede suceder que el sistema termorregulador fracase en su función de mantener la temperatura corporal, y que ésta disminuya o aumente por debajo o por encima de 37°C . En el primer caso se produce **hipotermia**. A medida que la temperatura corporal disminuye se enlentece la respiración y la frecuencia cardíaca y se pierde la consciencia. Cuando la temperatura corporal baja a 28°C , el hipotálamo deja de funcionar y la temperatura comienza a descender rápidamente hasta la muerte del sujeto. Sin embargo, si en este punto se le aplica calor extremo todavía puede recuperarse. Puede disminuirse la temperatura corporal hasta cerca del punto de congelación, y recuperarse si luego se aplica calor. Pero si la temperatura disminuye por debajo de 0°C se forman cristales de hielo que rompen los tejidos y producen daños irreversibles.

Cuando la temperatura aumenta excesivamente se produce **hipertermia o choque de calor**. En este caso aparece dolor de cabeza, confusión, pérdida de la consciencia, aumento de la frecuencia cardiaca, disminución de la presión arterial (todas las arterias se dilatan para eliminar calor), y si la temperatura aumenta a 43°C se produce daño cerebral. El choque de calor es más grave si el sujeto está deshidratado, porque entonces su capacidad de eliminar el calor por medio del sudor es menor.

Lo que hay que hacer en caso de **hipotermia o hipertermia** es bien simple: en la hipotermia hay que calentar al sujeto (abrigándolo, colocándolo cerca de un radiador, etc.) y en la hipertermia enfriarlo (mojándolo con agua fría o aplicándole hielo). No se debe confundir hipertermia con fiebre, en los dos casos hay un aumento de la temperatura corporal, pero es causado por diferentes agentes.

La fiebre es un mecanismo de emergencia, es un aumento en la temperatura del cuerpo por encima de los niveles normales. La fiebre se puede definir generalmente como una temperatura que es superior a los $37,2^{\circ}\text{C}$ por la mañana temprano o una temperatura superior a los $37,8^{\circ}\text{C}$ en cualquier momento del día. Este aumento de la temperatura corporal puede ser debido a cualquier tipo de infección [19].

4.4.1. Diseño

Gracias a la evolución de la tecnología y la integración de ésta podemos obtener soluciones a problemas de una forma sencilla y económica. La medición de la temperatura es uno de estos problemas, el cual si anteriormente se necesitaba un sensor que generara la señal, un filtro para quitar los posibles ruidos y adaptar la señal a los valores que necesitábamos, en estos momentos con la tecnología actual y los productos existentes en el mercado un único integrado es capaz de realizar todas estas operaciones.

El integrado elegido para realizar esta operación es el LM35 de Texas Instruments. El LM35 es un sensor de precisión de temperatura con todo lo necesario para su funcionamiento integrado en un único encapsulado. Este integrado puede ser alimentado con un voltaje de 4V a 30V, no variando por ello su salida. La salida del sensor es lineal y analógica con factor en su escala de $10\text{ mV}/^{\circ}\text{C}$. Por lo que una vez realizada la conversión A/D la fórmula usada para conseguir saber la temperatura actual mediante una conversión A/D es la mostrada en la Ecuación 4.5, en la cual VT es el voltaje de trabajo del microcontrolador, R es la resolución del conversor y x es la medida tomada. En nuestro caso $VT = 5$ y $R = 1023$, debido a que nuestro microcontrolador trabaja con un voltaje de 5 V y el conversor que dispone tiene una resolución de 10 bit, por lo que la ecuación se simplifica y la operación de obtención de la temperatura se resume

```

#define TEMP_CONS 0.488758553

int read_temp;
double temp;

read_temp = analogRead(entrada);
temp = TEMP_CONS*read_temp;

```

Código 4.1: Ejemplo para la obtencion de la temperatura en grados centígrados.

en un único cálculo, $T = 0,488758553x$, ya que la primera parte de la ecuación se transforma en una constante.

$$T = \frac{\left(\frac{VT}{R}\right)}{0,01}x \quad (4.5)$$

El conexionado del LM35 es muy sencillo. Únicamente son necesarias 3 líneas: alimentación, masa y salida, esta última conectadas a una entrada analógica de nuestro microcontrolador.

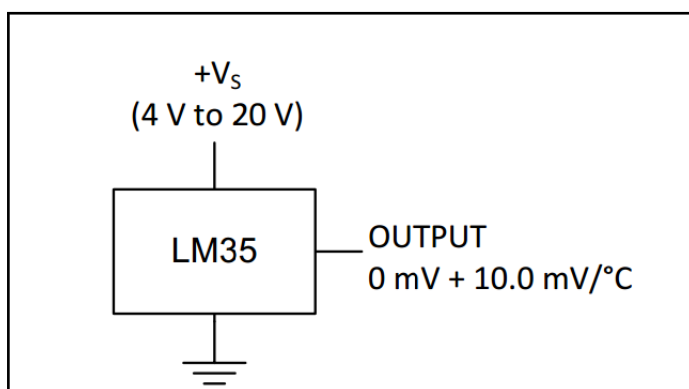


Figura 4.13: Conexionado del LM35.

Respecto al software, es muy simple ya que gracias a la simplificación de la Ecuación 4.5, solamente debemos definir una constante y realizar una única multiplicación con el dato obtenido de la conversión. Un ejemplo de ello es el Código 4.1.

Respecto a la tasa de muestreo de la temperatura, al no ser una medida que varíe muy rápido no es necesario muestrear a una frecuencia muy alta, por lo que como se ha explicado en la Sección 4.1.1, la tasa de refresco de la temperatura es de 1 Hz, o lo que es lo mismo de una muestra por segundo.

El flujo de datos a almacenar en la memoria es de 4 caracteres ASCII cada segundo lo que corresponde con 4 bytes por segundo. El consumo de potencia de este circuito es de $750 \mu W$.

4.4.2. Simulación

Para la simulación nos basaremos en el software Proteus el cual ya tiene implementada la librería necesaria para la simulación del sensor LM35. Para realizar correctamente esta simulación hemos añadido a nuestro proyecto de Proteus un sensor LM35, un arduino mega y un LCD de 16 caracteres por 2 líneas para poder representar de una forma gráfica los datos obtenidos del sensor.

Tanto el proyecto de Proteus como el código fuente de la prueba se adjuntan en el CD anexo a esta memoria.

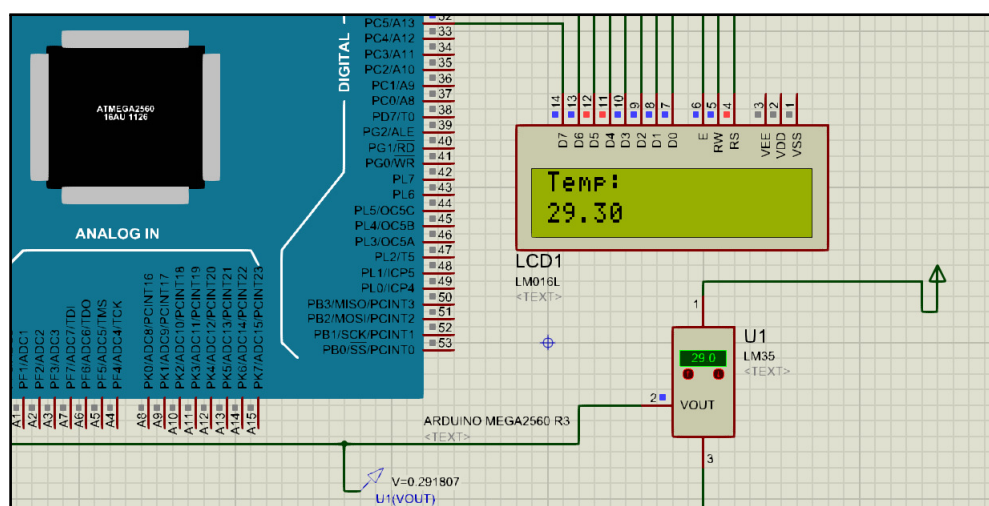


Figura 4.14: Simulación del funcionamiento del LM35 en Proteus.

En la Figura 4.14 se puede observar una captura de la pantalla del software Proteus con la simulación ejecutándose y pudiendo comprobar que si variamos la temperatura del sensor el microcontrolador transforma correctamente la medida leída por el conversor A/D en grados centígrados.

4.5. Aceleración y Posición angular

Un acelerómetro es un instrumento para medir aceleraciones. Esta aceleración no es necesariamente la misma que la aceleración de coordenadas (cambio en el espacio), sino es el tipo de aceleración asociada con el fenómeno de peso experimentado por una masa de prueba que se encuentra en el marco de referencia del dispositivo.

Para medir la posición angular en un determinado momento se usa un giroscopio. Este instrumento sirve para medir o asignar un momento angular a un objeto. Este dispositivo se usa cuando no funcionan los instrumentos magnéticos, por ejemplo las brújulas, o la precisión dada por éstas es insuficiente. Estos dispositivos son usados por satélites o aparatos de vuelo.

Gracias a la tecnología de fabricación actual de los Sistemas Microelectromecánicos, a partir de ahora MEMS (Microelectromechanical Systems), se pueden construir elementos que contienen partes mecánicas y partes electrónicas a una escala de fabricación nanométrica, por lo que la escala de integración en un mismo encapsulado es muy alta.

Usando esta tecnología, la mayoría de los giroscopios se basan en el efecto del péndulo de Foucault, mientras que los acelerómetros se basan en el traspaso térmico, por convección natural. Estos dispositivos miden cambios internos, de la transferencia de calor causada por la aceleración, ofreciendo ventajas significativas sobre el empleo de una estructura tradicional sólida de masas de prueba, ya que la masa de prueba en el diseño de los sensores MEMS son moléculas de gas, las estructuras móviles mecánicas son eliminadas dentro del acelerómetro. También se puede usar la técnica basada en el traspaso térmico para fabricar inclinómetros o giroscopios.

4.5.1. Diseño

Nosotros vamos a aprovechar la tecnología de fabricación MEMS, y así por ejemplo, el integrado MPU-6050 de InvenSense ya tiene integrado en un único encapsulado un giroscopio y un acelerómetro, además de toda la electrónica necesaria para tratar la señal de los sensores. Las capacidades de este integrado no se quedan ahí sino que también implementa un procesador digital de movimiento y sensibilidad ajustable de ambas medidas. El acelerómetro puede trabajar con sensibilidades de $\pm 2g$, $\pm 4g$, $\pm 8g$, y $\pm 16g$, y el giroscopio puede trabajar con sensibilidades de ± 250 , ± 500 , ± 1000 , y $\pm 2000^\circ/\text{sec}$. En la Figura 4.17 podemos observar la posición de los ejes según los tiene ordenados de fábrica el MPU-6050, por lo que debemos tener en cuenta esto en su colocación final en el prototipo.

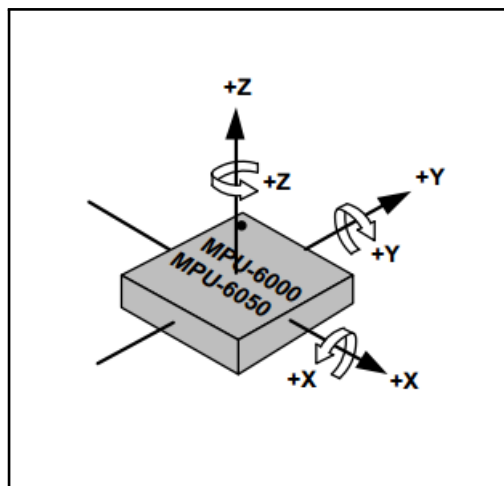


Figura 4.17: Orientación y polaridad de los ejes del MPU-6050.

El procesador digital de movimiento es de gran utilidad ya que nos ahorra unos cálculos muy complejos en el microcontrolador. Este procesador transforma los valores analógicos obtenidos de los 6 ejes (3 ejes del acelerómetro y 3 del giroscopio) en 9 ejes: 3 representan la dirección (heading o yaw), la elevación (pitch) y ángulo de alabeo (roll); otros 3 representan la aceleración en los 3 ejes con la influencia de la gravedad; y los últimos 3 representan la aceleración sin la influencia de la gravedad y avisándonos mediante una interrupción enviada por un terminal de conexión cuando ha realizado los cálculos. En la Figura 4.18 podemos ver la relación de los ejes del acelerómetro con el cuerpo humano.

Al representar los datos del giroscopio con ángulos de Euler aparece un efecto llamado Gimbal Lock. El Gimbal Lock es un efecto bastante molesto que se produce cuando rotas cualquier vértice o entidad en el eje Y 90 ó -90 grados. La consecuencia de esto es que el eje X y el Z acaban apuntando en la misma dirección, con lo que pierdes el control sobre ellos. Y esto ocurre con cualquier paquete de animación (3D Studio MAX, Maya, Softimage XSI, etc) y con cualquier motor 3D que utilice para las rotaciones los llamados ángulos de Euler.

Para visualizarlo podemos coger un bolígrafo apuntando hacia el horizonte, de tal forma que el eje Z recorre todo el largo del bolígrafo. Podemos girarlo en Z, y seguirá apuntando hacia el horizonte. Si lo giramos en X apuntará al suelo o al techo. Si lo giramos en Y apuntará a la izquierda o a la derecha.

Partiendo de la posición inicial, vamos girando el bolígrafo en Z, y sin dejar de hacerlo, giramos 90 grados en X. Ahora el bolígrafo gira igual que antes pero apuntando todo el rato al techo, no olvidemos que primero estamos aplicando el giro en Z y luego en X.

Sin embargo el giro en Y lo estamos calculando después del giro en X, es decir que en la posición actual del bolígrafo, si giramos en el eje Y, podemos ver que es el mismo giro que provoca el giro en Z. La razón es que el giro de 90 grados en X hace que el giro en Z que estamos aplicando antes de girar en X, coincida con el giro en Y que aplicaríamos después de girar en X. El efecto de todo esto es que en vez de tener posibilidad de maniobra en los tres ejes, cuando se solapan dos de ellos es como si perdiéramos un eje.

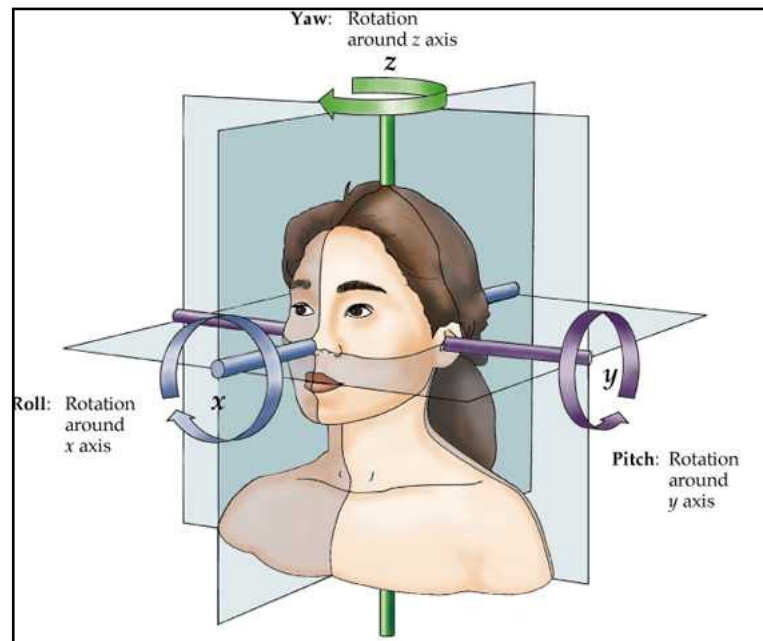


Figura 4.18: Orientación de los ejes del giroscopio en el cuerpo humano.

El bus de comunicaciones que usa este integrado es el I^2C . Este bus fue diseñado por Philips en 1991, la velocidad en modo estándar es de 100 kbits/s , pero también permite velocidades de $3,4\text{ Mbit/s}$. Es un bus muy usado en la industria, principalmente para comunicar microcontroladores con sus periféricos en sistemas integrados. Una principal característica del I^2C es que solamente utiliza dos líneas para transmitir la información independientemente del número de periféricos que controle el microcontrolador, estas líneas son: SDA, para datos, y SCL para reloj.

Estas líneas son de drenador abierto, por lo que necesitan resistencias de pull-up, esto hace que sea compatible con cualquier tecnología de fabricación (ttl, cmos, etc.). Los dispositivos conectados al bus I^2C tienen una dirección única para cada dispositivo. Estos dispositivos pueden ser maestros o esclavos. El dispositivo maestro inicia la transferencia de datos y además genera la señal de reloj, pero no es necesario que el maestro sea siempre el mismo dispositivo. Es decir, si hay varios maestros en un mismo bus se pueden ir pasando el control entre ellos pero no puede haber dos maestros

activos simultáneamente. Esto hace que el bus I^2C sea multimaestro. Un ejemplo de configuración lo podemos encontrar en la Figura 4.19 en la que tenemos un dispositivo que tiene el rol de maestro y dispositivos con el rol de esclavo.

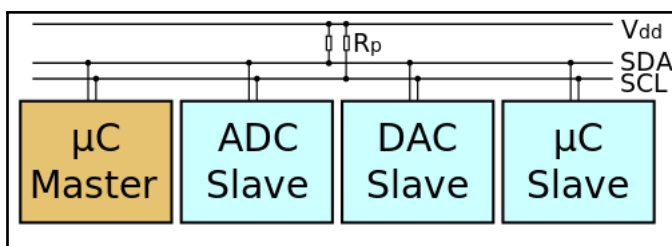


Figura 4.19: Un esquema representativo de un bus I^2C con un maestro y 3 esclavos.

El flujo de datos generado por el dispositivo que nos interesa grabar es de 12 bytes, debido a 3 ejes del acelerómetro y 3 del giroscopio, necesitando cada eje 2 bytes. Sin embargo al pasarlos a una representación con caracteres ASCII éstos pueden llegar a ocupar 5 caracteres como máximo, lo que supone $5 \times 6 = 30$ bytes cada 50 milisegundos. El consumo de potencia máximo de este circuito es de 20 mW, teniendo todo activado (acelerador + giroscopio + DMP).

4.5.2. Simulación

Debido a que el funcionamiento interno es desconocido y el sistema de cálculo de los ejes derivados del procesamiento está registrado comercialmente no existe librería alguna para poder realizar una simulación antes de empezar a usar hardware real.

4.5.3. Implementación

La implementación de este dispositivo es muy sencilla debido a que necesita muy pocas líneas para trabajar. Como se ha explicado, sólo necesita alimentación, masa, SDA, SCL, una señal de interrupción y un selector de dirección.

En la Figura 4.20 se observa el conector al cual va enchufado el MPU-6050, las conexiones de izquierda a derecha son VCC, GND, SCL, SDA, XDA, XCL, AD0 e INT. Se conecta SCL y SDA a los terminales del Arduino Mega homónimos, Vcc a +5V y GND a masa; la conexión INT al pin IO 3 del Arduino; AD0 a masa o VCC según la dirección de esclavo que le queramos dar y los terminales XDA y XCL son para conectar un esclavo de I^2C y que realice las funciones de esclavo y maestro a la vez.

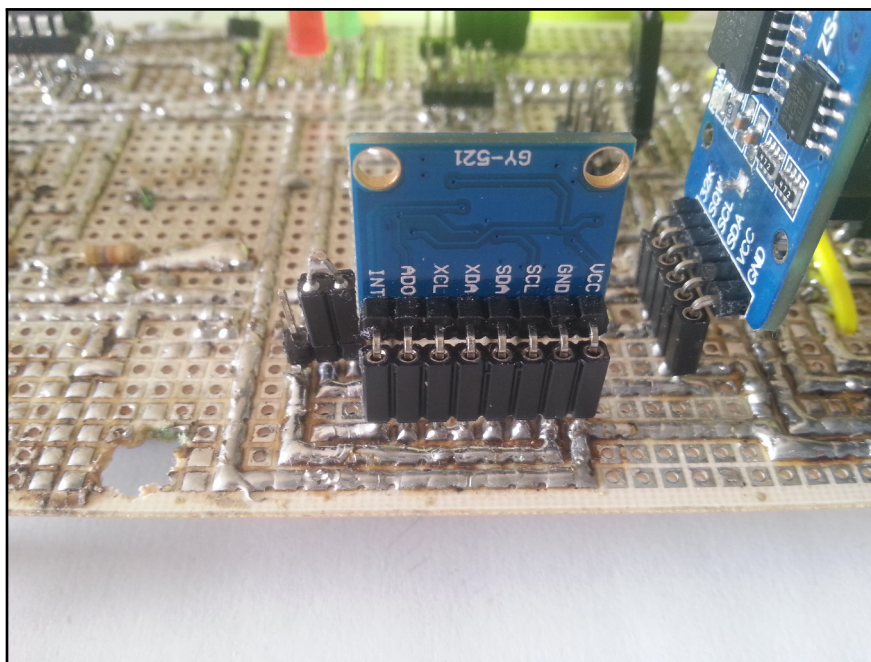


Figura 4.20: Implementación del MPU-6050 en el prototipo final.

4.6. RTC

El control del tiempo es algo importante en cualquier sistema. En sistemas autónomos, como nuestro dispositivo, en los cuales se almacenan datos es esencial saber en qué instante de tiempo se ha capturado el dato para luego poder hacer una representación de los datos correctamente.

La mayoría de microcontroladores no llevan un control de tiempo real, es decir, no saben qué hora es, pero sí controlan el tiempo que han estado encendidos. Por lo tanto podemos saber en cualquier momento cuánto tiempo ha pasado desde que se arrancó. Pero, ¿qué pasa si necesitamos saber qué hora es?

Para solucionar esta necesidad se usan los integrados RTC. Estos integrados son capaces de mantener un contador en el cual se mantiene una hora predefinida que sigue funcionando incluso si se apaga el sistema debido a que mantienen activo el contador con una pequeña batería que les puede durar varios años. Estos integrados son muy usados. A sí por ejemplo, todos los computadores llevan uno, y por eso cuando apagamos nuestro PC se sigue manteniendo la hora aunque lo desenchufemos de la red eléctrica.

Nosotros usaremos este tipo de integrados con el mismo objetivo que los PC, es decir, saber qué hora es al arrancar el sistema. Después usaremos el contador interno del microcontrolador para saber el instante de tiempo actual desde el arranque. Para

saber la hora en cada momento sólo hay que sumar a la hora leída en el inicio el tiempo actual o volver a preguntar al integrado por la hora actual.

4.6.1. Diseño

Para obtener la hora nosotros usaremos un integrado de Maxim, en concreto el modelo DS3231, el cual integra una gran parte de la electrónica necesaria dentro de su encapsulado. Además, este modelo ofrece una precisión extrema y la posibilidad de añadir una batería externa que mantendría el tiempo durante años ya que en este modo solamente consume $0,84 \mu A$.

El bus de comunicación es I^2C . Es el mismo bus que el usado para el Giroscopio/Acelerómetro en la Sección 4.5, por lo que este dispositivo tiene el rol de esclavo en el bus de comunicaciones.

El conexionado del DS3231 es muy sencillo, incluso más sencillo que el usado en el Giroscopio/Acelerómetro. Únicamente son necesarias 4 líneas, si nos fijamos en la Figura 4.21 nos damos cuenta que estas son: alimentación, masa, SDA y SCL, estas dos últimas pertenecientes al bus I^2C .

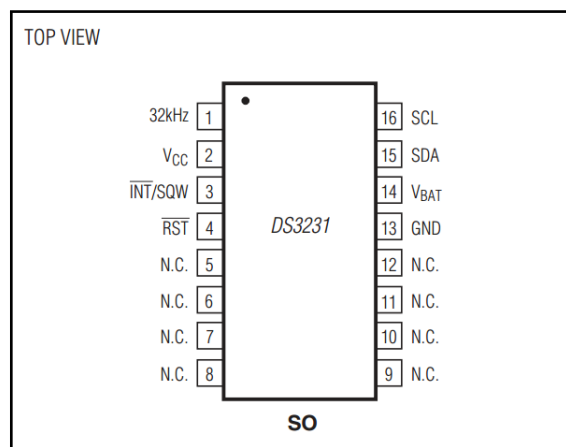


Figura 4.21: Pinout del DS3231.

Para poder ajustar el contador de tiempo interno del dispositivo, se dispone de un método realmente sencillo. En la raíz de la tarjeta SD del prototipo se crea un directorio con 6 bytes (hora, minutos, segundos, día, mes, año). Cuando se inicia el prototipo se comprueba si ese fichero existe. Si es así se cargan los valores en el dispositivo RTC antes de su lectura, y si no se continúa con una carga normal.

El valor del contador interno del RTC sólo se lee en el arranque del dispositivo, posteriormente si se necesita saber en qué instante de tiempo se está se usará el contador interno del microcontrolador debido a que éste es más rápido de consultar que el del RTC. El consumo de potencia de este circuito es de 1,5 mW.

4.6.2. Simulación

Para la simulación nos basaremos en el software Proteus el cual ya tiene implementada la librería necesaria para la simulación del dispositivo DS3231. Para realizar correctamente esta simulación hemos añadido a nuestro proyecto de Proteus un dispositivo DS3231, un Arduino MEGA y un LCD de 16 caracteres por 2 líneas para poder representar de una forma gráfica los datos obtenidos del sensor. Tanto el proyecto de Proteus como el código fuente de la prueba se adjuntan en el CD anexo a esta memoria.

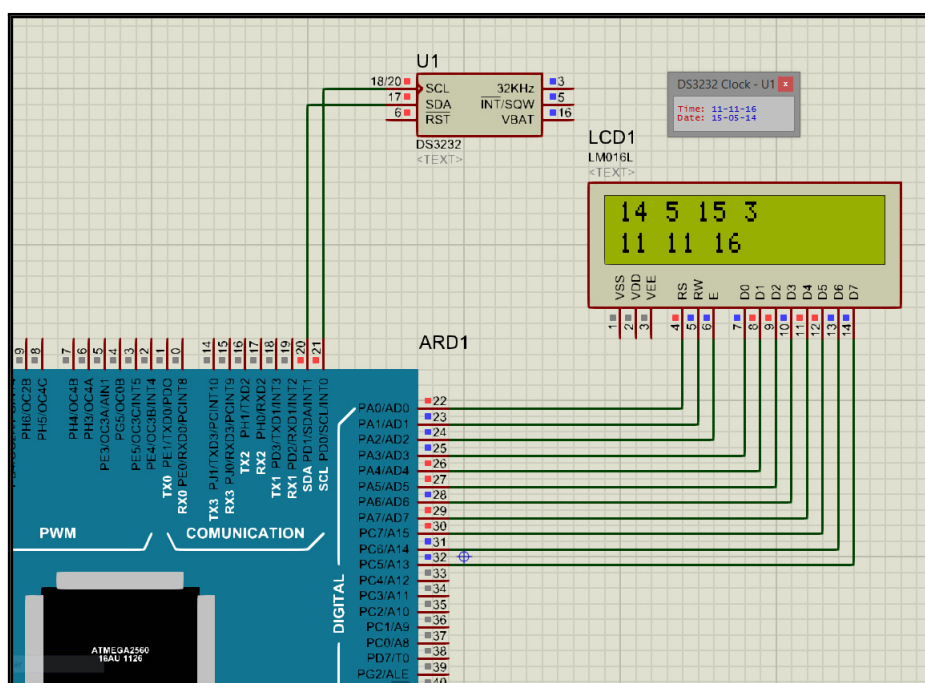


Figura 4.22: Simulación del funcionamiento del DS3231 en Proteus.

En la Figura 4.22 se puede observar una captura de la pantalla del software Proteus con la simulación ejecutándose y pudiendo comprobar que si variamos el contador del DS3231, este cambio se refleja en nuestro LCD y en la representación gráfica del DS3231 del Proteus.

4.6.3. Implementación

La implementación de este dispositivo es muy sencilla debido a que necesita muy pocas líneas para trabajar. Como se ha explicado, sólo necesita alimentación, masa, SDA y SCL.

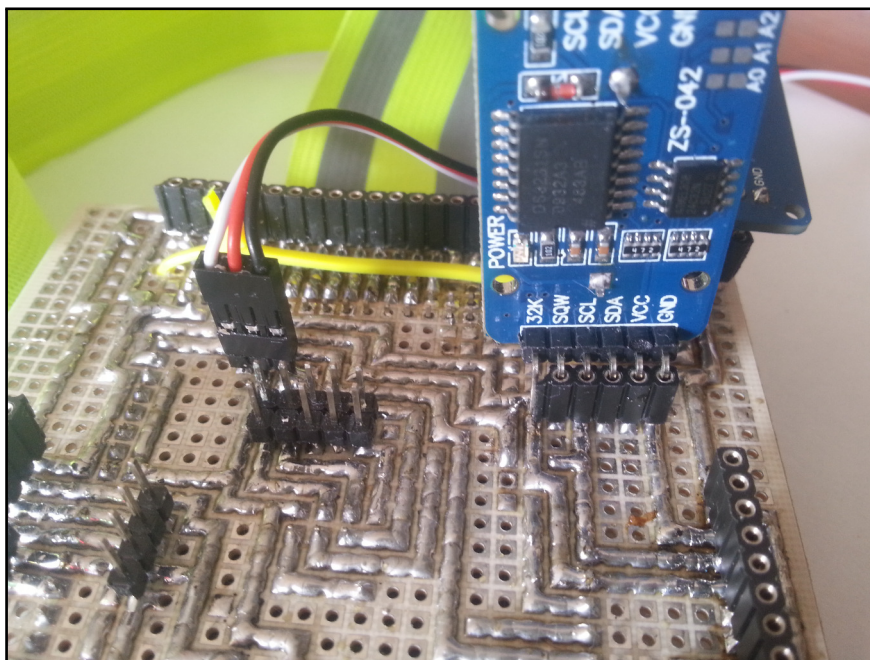


Figura 4.23: Implementación del DS3231 en el prototipo final.

En la Figura 4.23 se observa el conector al cual va enchufado el DS3231. Las conexiones de izquierda a derecha son SQW, SCL, SDA, VCC y GND. Conectando SCL y SDA a los terminales del Arduino Mega homónimos, Vcc a +5V y GND a masa. La conexión SQW no se utiliza en este proyecto pero sirve para generar una onda cuadrada o una interrupción programada mediante una alarma ajustada en un determinado momento de tiempo.

4.7. Almacenamiento de datos

El almacenamiento de datos es algo esencial para nuestro prototipo debido a que gracias a los datos almacenados se puede obtener un historial de la evolución del estado de salud del paciente.

En la búsqueda del medio de almacenamiento de datos más adecuado podemos encontrar varias soluciones entre los que podemos citar:

Memorias EEPROM: Este tipo de memoria es muy común e incluso muchos microcontroladores incorporan una dentro de su mismo encapsulado, permitiendo realizar escrituras y lecturas en ellas de una forma muy sencilla. Sin embargo, suelen ser de muy poca capacidad, del orden de KBs, por lo que la cantidad de datos que podríamos almacenar sería pequeña.

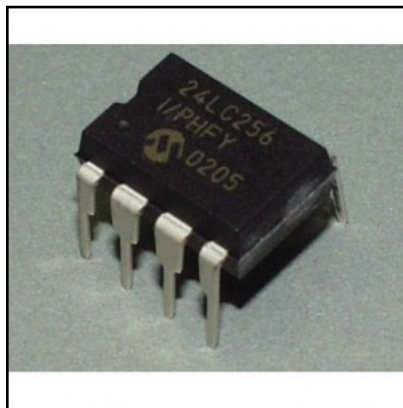


Figura 4.24: Memoria EEPROM de 256 Kbits (32 KBytes) fabricada por Microchip.

Microdrives: Es una evolución de los discos duros convencionales hacia una miniaturización del dispositivo. Usado en dispositivos como por ejemplo el reproductor de audio Ipod de Apple, son descartados en los dispositivos alimentados con baterías por su gran coste y consumo de energía. Estos dispositivos han logrado llegar al orden de los cientos de gigabytes.



Figura 4.25: Microdrives del fabricante Hitachi.

Memorias FLASH: Este tipo de memoria es la más usada actualmente debido a que ofrece una gran velocidad en el acceso a los datos y dispone de varias interfaces para el acceso a la información. La relación capacidad/precio también es una ventaja, debido a que en la actualidad se ofrecen capacidades en el orden de los gigabytes por precios que no superan los 5 euros.

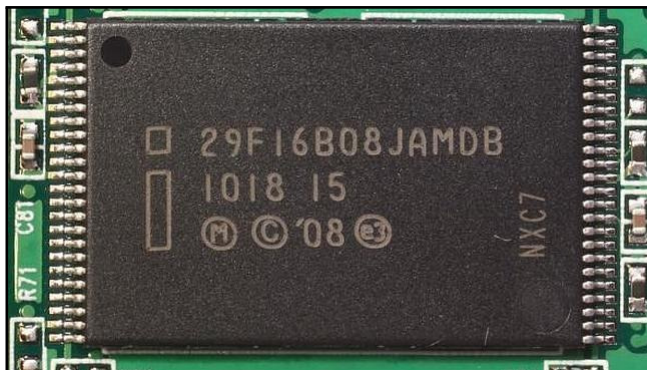


Figura 4.26: Memoria FLASH de Intel con una capacidad de 16 gigabytes.

Viendo estas posibles opciones y considerando los datos de la Tabla 4.2, nos decantaremos por las memorias FLASH. En concreto, por una tarjeta SD, la cual en su interior lleva una memoria FLASH pero ofrece una interfaz SPI. Esta interfaz es muy sencilla de utilizar ya que la mayoría de microcontroladores poseen un módulo de comunicación de esta interfaz.

La capacidad de la tarjeta limitará el tamaño del historial a guardar. Sin embargo, si suponemos el caso más pesimista que es guardar 1 kibibyte³ de información en la tarjeta por segundo en una tarjeta de 4 gibibytes⁴, ésta podría estar guardando información del prototipo durante 48 días hasta llenar su capacidad.

Tipo	Capacidad máxima	Consumo	Relación Capacidad/Precio
EEPROM	Baja	Bajo	Alta
Microdrive	Alta	Alto	Media
Flash	Media	Bajo	Baja

Tabla 4.2: Comparación entre diferentes tipos de memoria para almacenamiento.

4.7.1. Diseño

La parte hardware de la tarjeta es sencilla debido a que proporciona una interfaz SPI. Hay un problema y es que debemos adaptar la lógica que usa el arduino de 5V (TTL) a la lógica usada por las tarjetas SD que es de 3,3V.

³Unidad definida por la Comisión Electrotécnica Internacional (IEC) en diciembre de 1998, estableciendo que un kibibyte es igual a 1024 bytes. Su nombre proviene de la contracción de kilo binary byte y su nomenclatura es KiB.

⁴Múltiplo del KiB. 1 GiB = 1024 MiB = 1048576 KiB.

Esta conversión la realizamos mediante divisores de tensión y solamente cuando el flujo de la señal va desde el del Arduino hacia la tarjeta SD, pero no cuando el flujo de los datos va de la tarjeta SD al Arduino debido a que los 3,3V de la tarjeta SD no dañarán la entrada del Arduino ya que ésta trabaja con 5V.

Otro aspecto importante es el software. Para esto utilizaremos la librería incluida en el IDE del Arduino, la cual es capaz de leer y escribir en el sistema de archivos fat32, en el cual podemos crear ficheros de 4 gibibytes como máximo, tal que en el peor caso se tienen 48 días continuos de almacenamiento.

Otro aspecto que importa en nuestro sistema es el consumo. El consumo de una tarjeta SD depende del modelo de tarjeta. Otro parámetro del cual depende el consumo de la tarjeta SD es la cantidad de escrituras y lecturas que se realicen, pero para aproximarnos a unas cifras, la mayoría de tarjetas gastan 0,05W estando en espera y 0,4W realizando alguna operación.

4.7.2. Simulación

Para realizar la simulación se ha usado el software Proteus, en el cual se incorpora un módulo que simula la tarjeta SD y otro para simular el Arduino. El software creado escribe en un fichero 100 líneas de texto. Antes de realizar el grabado se comprueba la existencia del fichero y en el caso de que se encuentre el fichero éste será borrado.

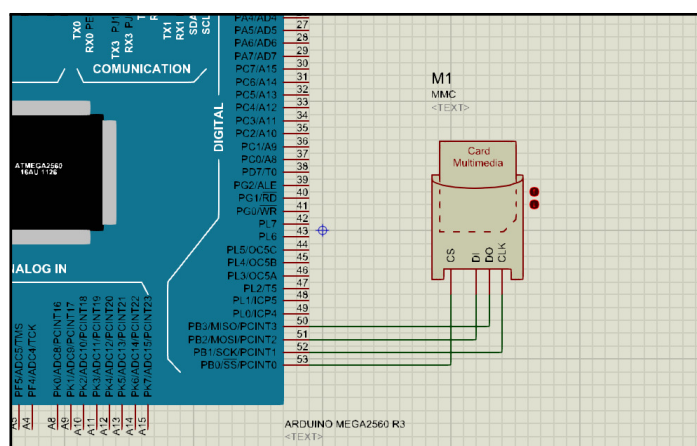


Figura 4.27: Simulación del funcionamiento de la tarjeta SD en Proteus.

En la Figura 4.27 se observa el circuito realizado en software de simulación. El módulo encargado de simular la tarjeta SD necesita un archivo especial para emular la tarjeta SD. Este archivo lo podemos manejar tanto en sistemas Windows como en sistemas UNIX. Los pasos a seguir en cada caso son los siguientes:

Windows: Para manejar el fichero en Windows necesitamos el software WinImage⁵, el cual dispone de una opción en su menú de archivo, mediante la cual podemos crear un archivo de imagen especificando qué sistema de archivo queremos y de qué tamaño lo queremos.

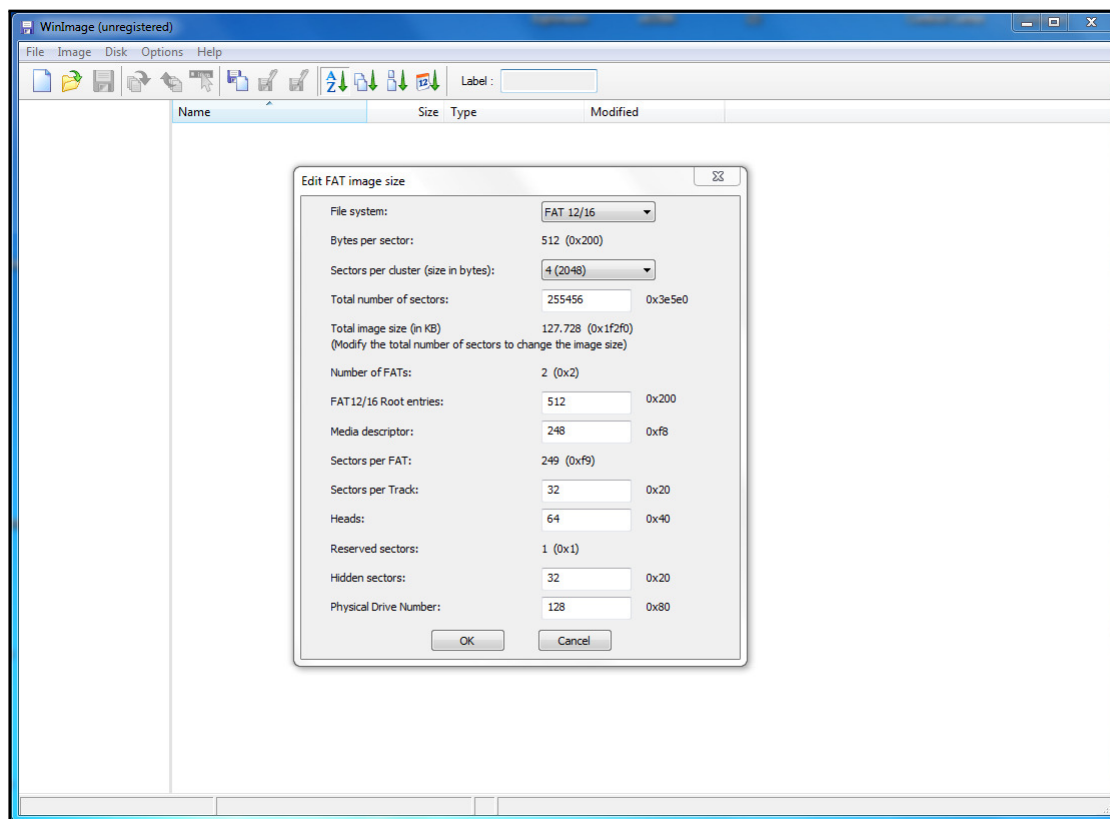


Figura 4.28: Opción del WinImage en el cual podemos crear un fichero de imagen con formato FAT32.

Para posteriormente leer el fichero y ver su contenido podemos abrirlo con el mismo software que lo hemos creado.

UNIX: En sistemas UNIX necesitamos una tarjeta SD del tamaño que queremos la imagen. Esta tarjeta la formatearemos en el sistema de archivos que queramos simular. Posteriormente realizaremos una imagen con la herramienta *dd* de UNIX.

Para poder leer el contenido del archivo de imagen en un sistema UNIX tenemos dos opciones. Una consiste en volver a grabar el archivo en la tarjeta SD por el proceso contrario a la creación mediante el comando *dd*, y la otra en montarlo con el comando *mount*. Esta última opción sólo funciona si el archivo de imagen tiene un sistema de archivos que reconozca nuestro sistema.

⁵<http://www.winimage.com/>

4.7.3. Implementación

Para la implementación en el prototipo final se ha usado un módulo que ya dispone de toda la electrónica necesaria para el uso de la tarjeta SD. Este módulo se ha conectado al puerto SPI del microcontrolador y usando el programa del apartado anterior, junto a una tarjeta SD formateada en FAT32. Esto se ha realizado dejando un conector en la placa del prototipo para poder hacer pruebas o realizar una sustitución rápida por otro módulo de almacenamiento, como se puede observar en la Figura 4.29.

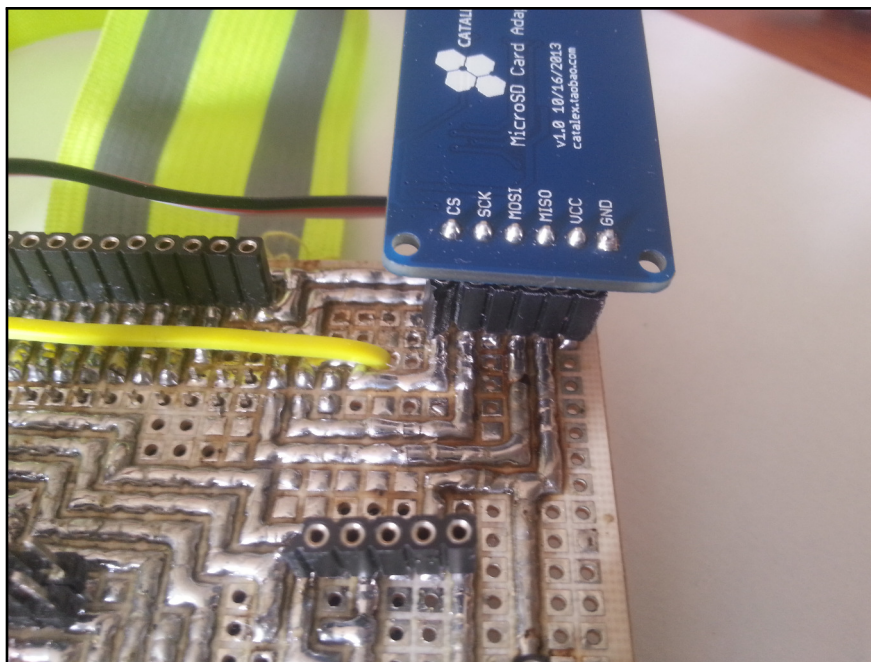


Figura 4.29: Implementación del módulo SD en el prototipo final.

4.8. Software

Para diseñar el software que llevará el microcontrolador hay que pensar de una manera un poco diferente a la programación para PC normal. Esto es debido a que un microcontrolador no disponemos de hilos secundarios. También hay que tener en cuenta el corazón de un programa de un microcontrolador es un bucle infinito, ya que un microcontrolador tiene normalmente la misión de monitorizar algún elemento.

En este apartado explicaremos a grandes rasgos el software utilizado para gestionar el monitor Holter. Esta explicación se realizará principalmente mediante diagramas de flujo, que ofrecerán de una manera clara y sencilla el funcionamiento del software.

Comenzaremos a explicar de la misma manera que arranca el microcontrolador. El microcontrolador cuando arranca ejecuta una sola vez la función `setup`. Esta función la usamos para inicializar todos los dispositivos y comprobar que estos funcionen correctamente. En la Figura 4.31 podemos observar el flujo que sigue el software en la función `setup`.

Las siguientes funciones que vamos a explicar son las ISR o rutinas de interrupción. En ellas se tratan eventos como la terminación del procesamiento de datos en el acelerómetro o la terminación de una conversión de analógico a digital. En la Figura 4.30 se representa la rutina de interrupción asociada al evento que genera el acelerómetro al terminar de realizar los cálculos en su procesador. Como se puede observar es muy sencilla ya que solo se activa un “flag”.

En la Figura 4.32 podemos observar el diagrama de flujo de la ISR que controla la interrupción que se genera cuando se finaliza una conversión de analógico a digital. Como se puede observar es más compleja que la anterior. El corazón de esta ISR es un `switch` el cual selecciona en que fase se encuentra. Cada 125000 conversiones se realiza una para la temperatura, es decir, cada segundo aproximadamente.

Por último nos queda comentar el bucle principal. En la Figura 4.33 se encuentra su diagrama de flujo. La parte más importante en este bucle es la espera de un dato del acelerómetro. Esta espera se realiza con un bucle “while”, del cual no salimos hasta que no hay un dato listo. Mientras estamos en este bucle aprovechamos para calcular la oximetría, pero solo realizamos un cálculo en cada vuelta debido a que se deben realizar la media de 200 valores y esto dejaría al microcontrolador ocupado en esta tarea. Una vez que ha llegado un dato del acelerómetro se sale del bucle escribiendo en la tarjeta SD los datos recién recibidos del acelerómetro. Cuando se sale del bucle también se comprueba si hay datos médicos disponibles y si estos existen también se escriben en la tarjeta SD.

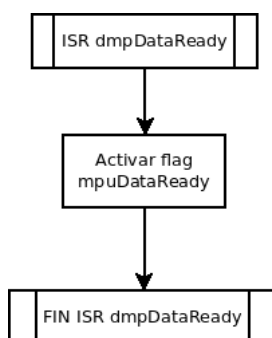
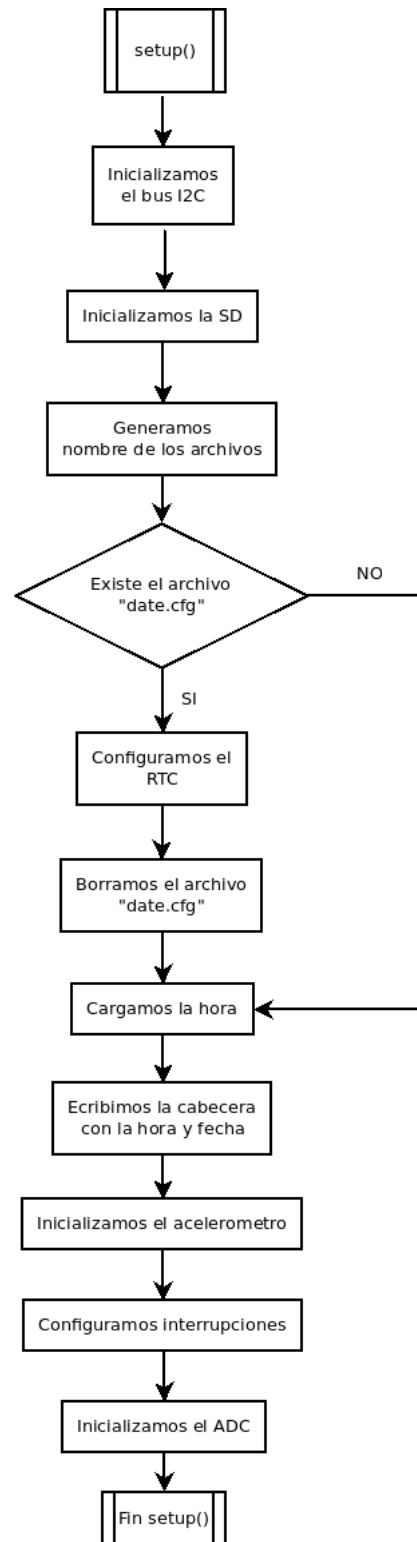


Figura 4.30: Diagrama de flujo de la ISR `dmpDataReady`.

Figura 4.31: Diagrama de flujo de la función `setup`.

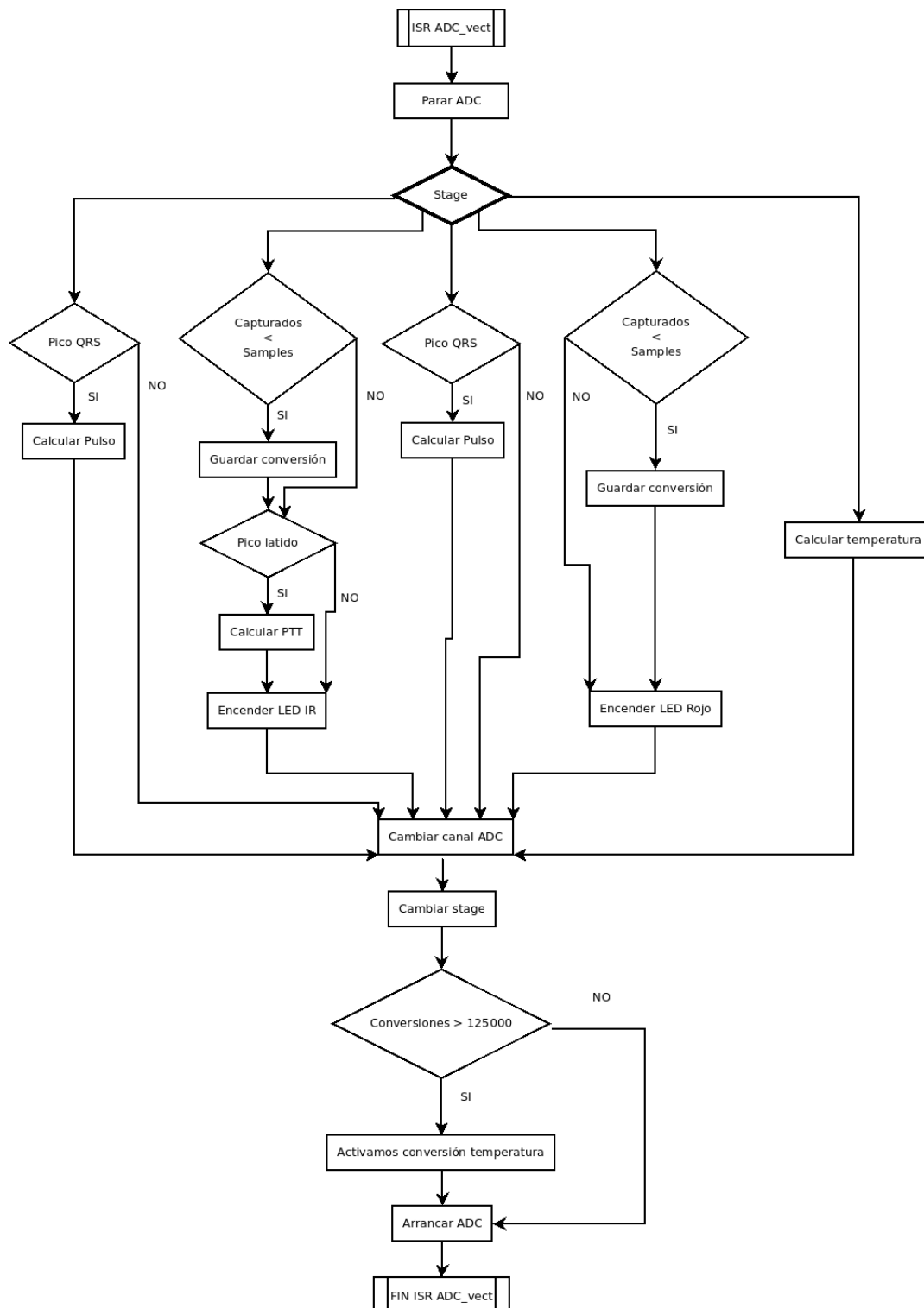


Figura 4.32: Diagrama de flujo de la ISR ADC_vect.

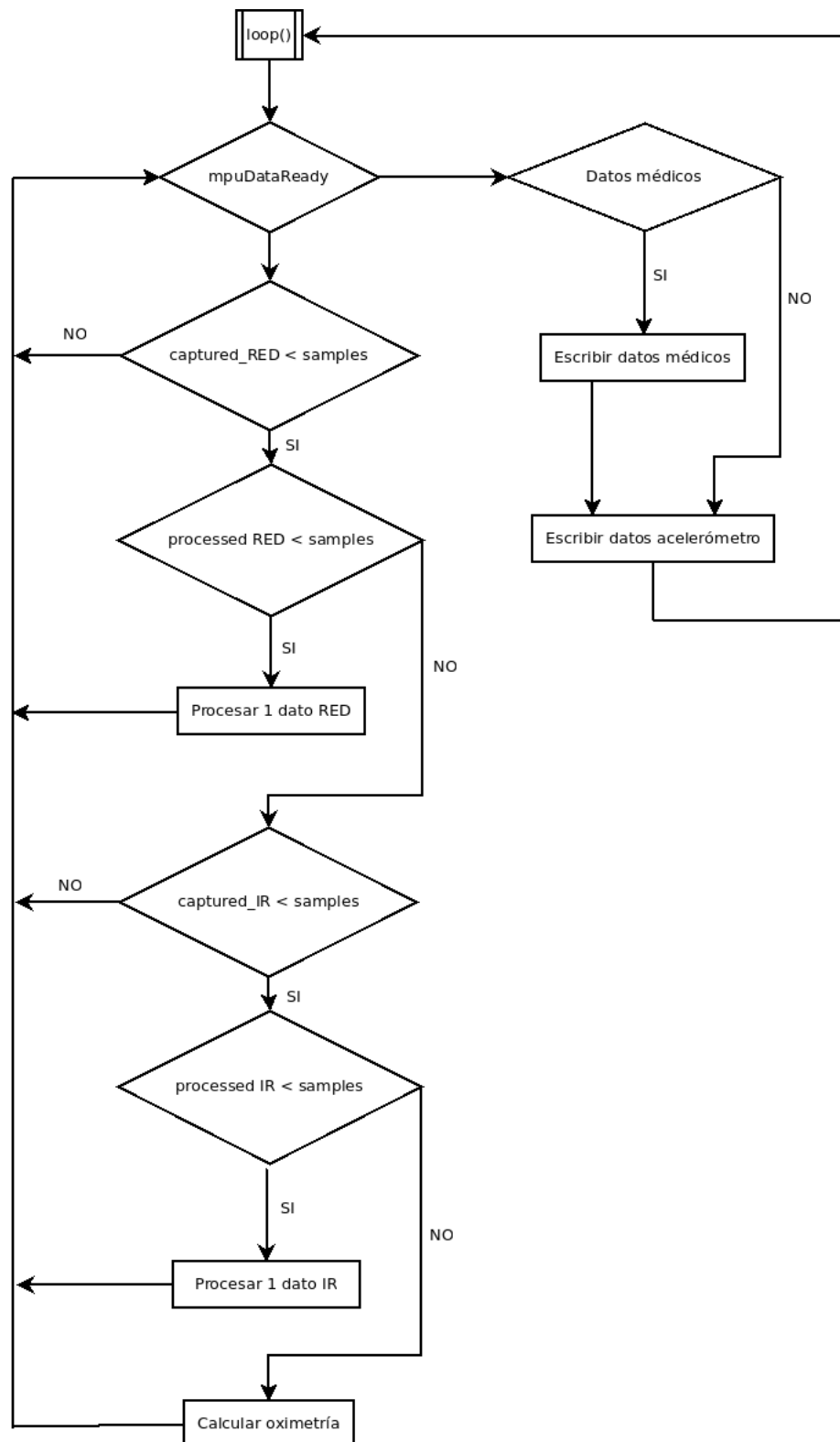


Figura 4.33: Diagrama de flujo del bucle principal.

Capítulo 5

EVALUACIÓN

En este capítulo se evaluará la funcionalidad de nuestro dispositivo. Para hacerlo se realizarán pruebas de dos tipos diferentes. Unas tendrán que ver con los aspectos médicos y las otras con la parte del movimiento. En el segundo caso se realizará una representación de los parámetros médicos junto los parámetros de movimiento. Para la realización de las pruebas ha sido necesario la colaboración de varios voluntarios. En la Tabla 5.1 se puede observar la relación de voluntarios con sus correspondientes edades, pesos, alturas y sexo. También hay que destacar que todos los voluntarios no tienen patologías conocidas que influyan en el resultado de estas pruebas.

Voluntario	Sexo	Edad	Peso	Altura
1	Varón	29	78 Kg	168 cm
2	Varón	21	63 Kg	178 cm
3	Varón	22	58 Kg	172 cm
4	Varón	22	82 Kg	165 cm
5	Mujer	34	55 Kg	169 cm

Tabla 5.1: Datos de referencia de los voluntarios.

5.1. Pruebas de funcionamiento

En la primera prueba se ha sometido a los voluntarios a 5 minutos de monitorización con el prototipo. Para ello se ha colocado el prototipo tal como se muestra en la Figura 5.1. Todos los voluntarios a los que se les ha realizado la prueba estaban sentados y con una posición relajada y cómoda.

Una vez cumplido el tiempo estipulado para la prueba se recogen los datos guardados en la tarjeta SD, procediendo posteriormente a su representación. Nuestro prototipo guarda los valores en texto plano y cada valor separado de su predecesor por “;”, este

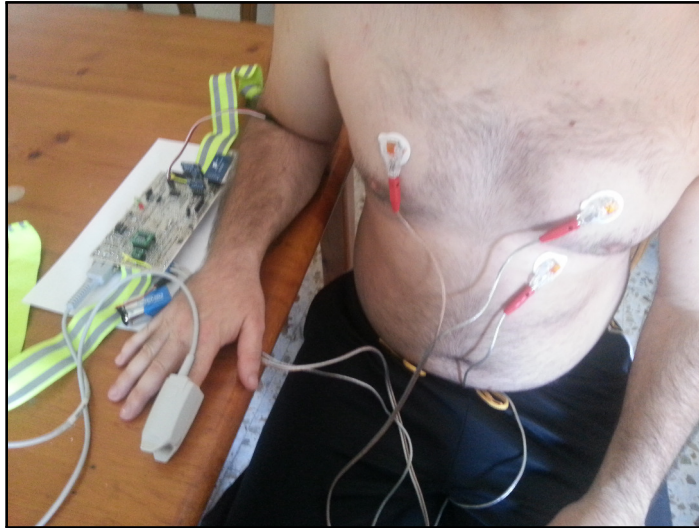


Figura 5.1: Colocación del prototipo en un voluntario.

formato se conoce como csv y es implementado por la mayoría de gestores de hojas de cálculo.

Los datos generados son muy extensos para su representación en una tabla. Pero en cambio, la representación de estos datos en una gráfica es muy sencilla. Por lo tanto, representaremos los valores en gráficas, usando para cada magnitud biológica una gráfica diferente.

En la primera gráfica (Figura 5.2) se observa las pulsaciones por minuto a lo largo de la duración de la prueba. Como se puede observar, dos de los voluntarios al comienzo de la prueba contaban con un pulso por encima del normal en un estado de relajación. Este pulso anormal es debido a una reacción muy común en los seres humanos y es debido al miedo. En la realización de algunas pruebas médicas es más habitual de lo que se cree, tanto que tiene nombre propio. A esta reacción se le conoce en el argot médico como “*síndrome de la bata blanca*”, Los resultados de la prueba médica se ven alterados por el miedo o un estado nervioso ante lo desconocido. En nuestro prototipo estos datos adulterados se deben, según los comentarios de los voluntarios, al miedo por sufrir algún tipo de descarga eléctrica. Pero como se puede observar en la gráfica, este temor desaparece con el paso del tiempo e incluso uno de ellos logra relajarse.

Como se puede observar, el algoritmo que mide el pulso responde de diferente manera con cada voluntario. Esto es debido a que dos individuos tienen diferente onda cardíaca. En la Figura 5.2 se puede observar que para algunos voluntarios el algoritmo tiene fluctuaciones, como por ejemplo el voluntario 1 y el voluntario 4. Para otros en cambio el algoritmo funciona perfectamente, como por ejemplo el voluntario 2 y el voluntario

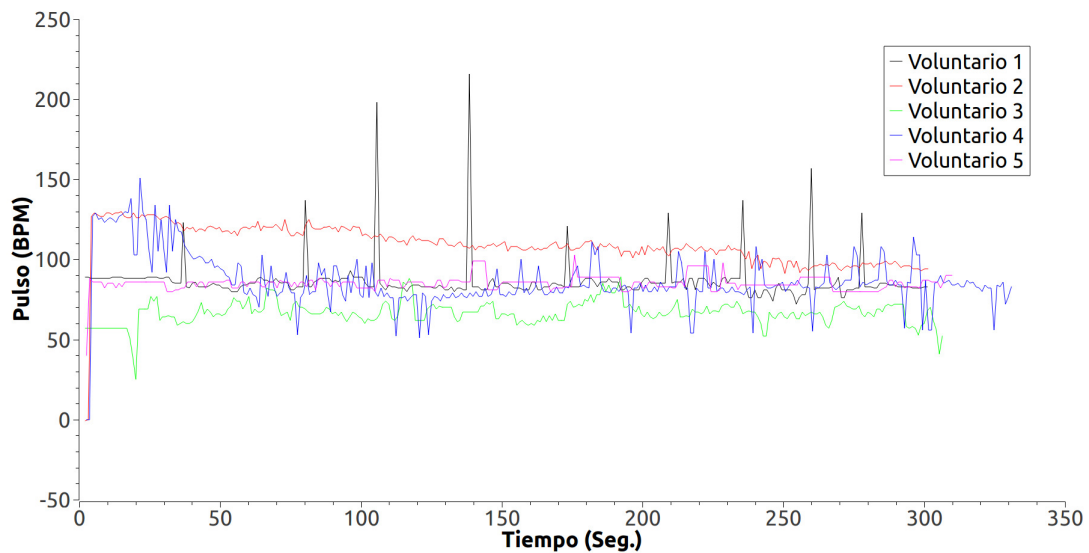


Figura 5.2: Gráfica representando las pulsaciones por minuto en el tiempo.

5.

La Figura 5.3 se representa la saturación de oxígeno durante la prueba. Todos los valores se mantienen en unos valores normales. Si alguno hubiese bajado del 90 % podríamos estar hablando de un principio de hipoxia¹.

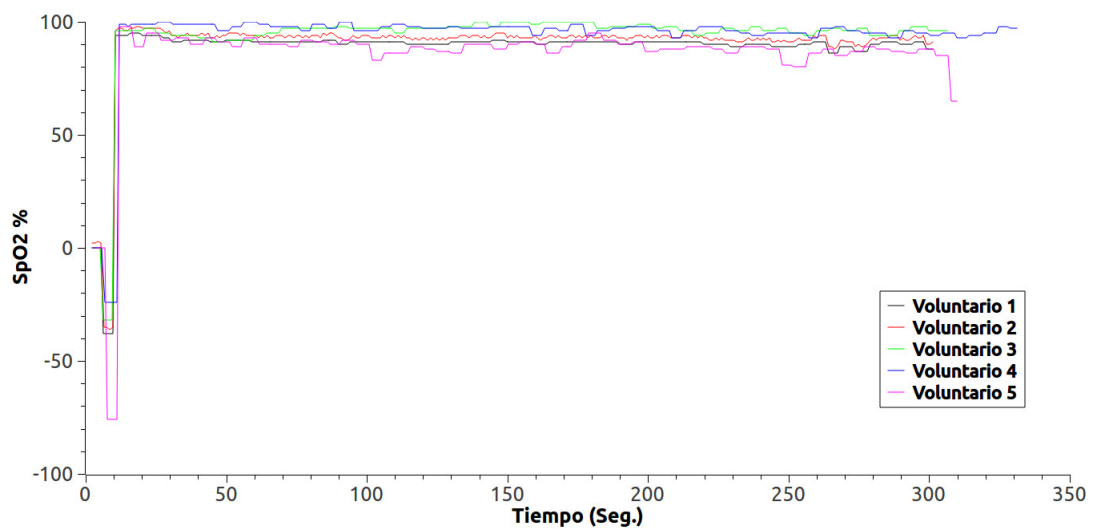


Figura 5.3: Gráfica representando la saturación de oxígeno en el tiempo.

Esta magnitud biológica necesita unos segundos para lanzar el primer valor válido debido a que la señal debe estabilizarse y la memoria intermedia o buffer debe llenarse

¹Es un estado en el cual el cuerpo completo o una región del cuerpo, se ve privado del suministro adecuado de oxígeno. La hipoxia está generalmente asociada con las alturas, siendo llamada «mal de montaña». Es especialmente grave en altitudes superiores a los 8.000 metros.

con valores de la señal estabilizada para que el dato calculado sea correcto. Si observamos la Figura 5.3 todos los valores se encuentran dentro de los márgenes aceptables. Hay valores del voluntario 5 que bajan del valor mínimo de saturación de oxígeno pero es debido a una mala colocación de la sonda pulsioximétrica.

La gráfica representada en la Figura 5.4 muestra la temperatura corporal durante la prueba. Como se puede observar todos los valores son normales y ninguno de nuestros voluntarios ha entrado en ningún estado de hipertermia o hipotermia.

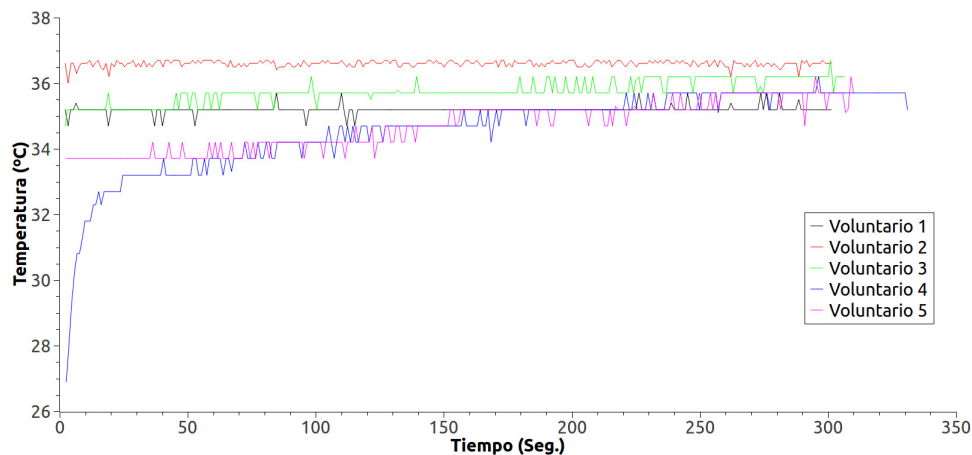


Figura 5.4: Gráfica representando la temperatura corporal en el tiempo.

En este caso la gráfica muestra, para la mayoría de voluntarios, pequeñas fluctuaciones debidas a los cambios de presión al sujetar el sensor térmico. Cualquier variación de presión afecta al intercambio de temperatura entre el cuerpo y el sensor. El sensor, al tener una rápida respuesta, refleja casi de inmediato este cambio de temperatura.

La gráfica de la Figura 5.5 contiene datos médicos, y en ella se observa la diferencia de tiempo entre el pulso de la ECG y el pulso detectado en la oximetría. Este valor se puede usar para calcular la presión arterial de una manera indirecta.

Como se observa, para esta magnitud se necesita un tiempo de inicialización muy parecido al tiempo de inicialización de la oximetría. Esto es debido a que tiene que estabilizarse la señal para poder detectar el pulso y calcular este valor.

Todos los archivos que se han generado en las pruebas se han adjuntado en el CD que acompaña a esta memoria.

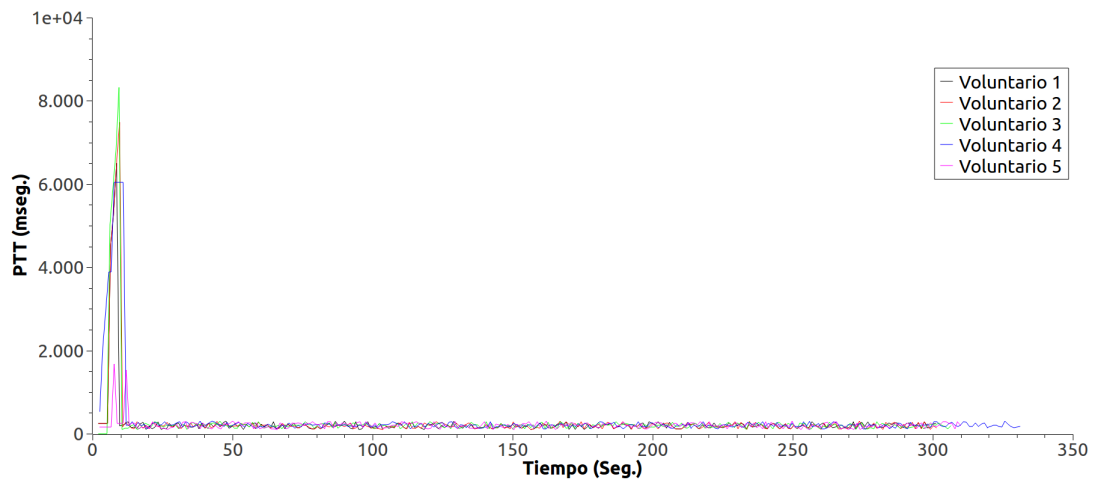


Figura 5.5: Gráfica representando el PTT en el tiempo.

La siguiente prueba se ha realizado con un único voluntario. En esta prueba se evalúa la captura de movimientos por parte de nuestro prototipo. Para ello se ha colocado el prototipo en el pecho del voluntario mediante un cinturón ajustable y por encima de una camiseta para evitar cortocircuitos, tal como se muestra en la Figura 5.6.

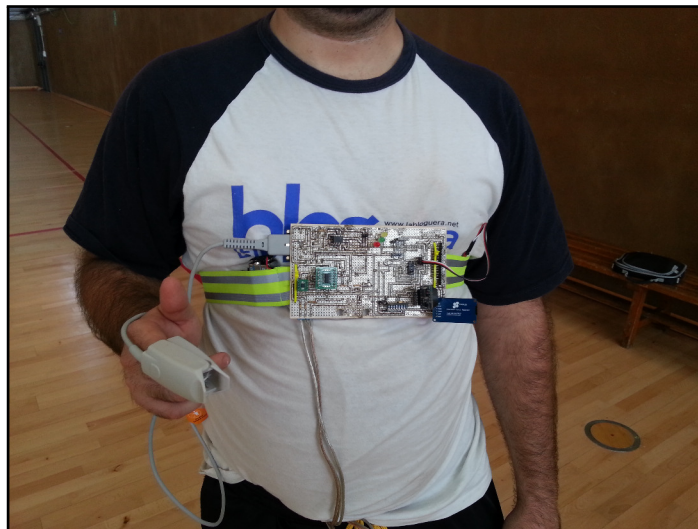


Figura 5.6: Colocación del prototipo para la prueba del movimiento.

Para probar la captura de los movimientos se consideran 3 situaciones diferentes. En una el voluntario andará, en la segunda correrá y por último se simulará una caída en una colchoneta.

La gráfica de la Figura 5.7 ha sido generada con los datos recogidos en la prueba de movimiento con el voluntario andando. Como se puede observar, los picos marcados corresponden a los pasos del voluntario.

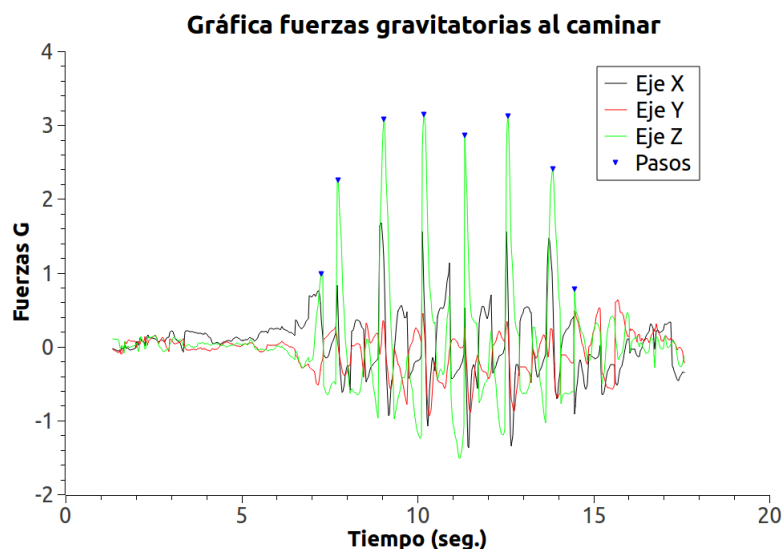


Figura 5.7: Gráfica representando el movimiento en la prueba con el voluntario andando.

La gráfica de la Figura 5.8 ha sido generada con los datos obtenidos en la prueba de movimiento con el voluntario corriendo. Como se puede observar, los picos marcados corresponden a las zancadas del voluntario. En concreto los puntos señalados corresponden al momento en el cual el voluntario impulsa su cuerpo para elevarlo, y por eso las fuerzas G son negativas.

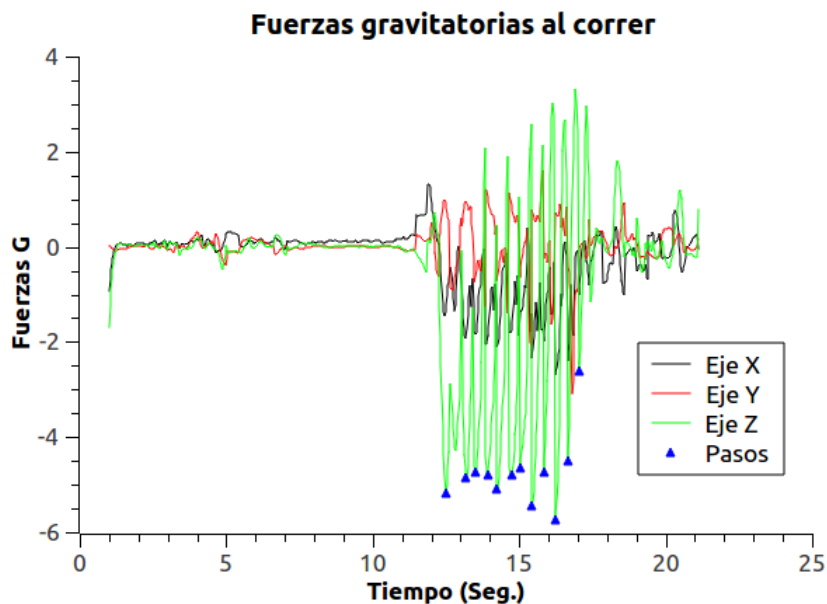


Figura 5.8: Gráfica representando el movimiento en la prueba con el voluntario corriendo.

Los archivos que almacenan los datos médicos y de movimiento comparten la misma referencia de tiempo. Por lo tanto, podemos representar un dato médico o varios junto a los datos de movimiento. Un ejemplo de esto es la gráfica de la Figura 5.9 en la cual se representa la captura de movimiento durante una caída simulada hacia atrás junto al pulso.

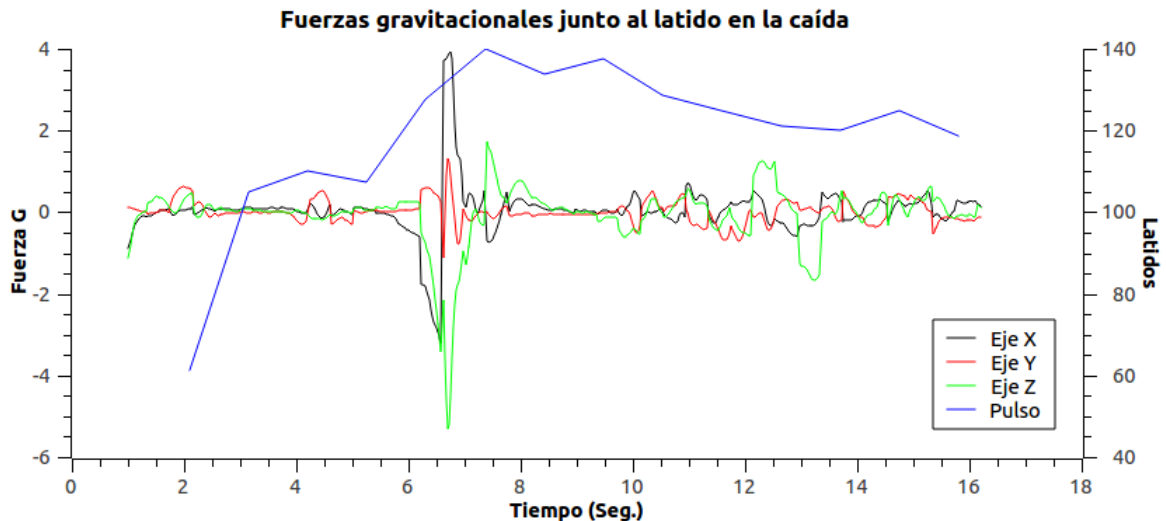


Figura 5.9: Gráfica representando el movimiento en la prueba con el voluntario simulando una caída junto con el pulso.

En la última prueba de movimiento también son interesantes los datos del giroscopio. La Figura 5.10 se puede observar que en la caída el pitch (eje Y) gira unos 90 grados. Pero en este giro podemos observar que el yaw (eje Z) también gira. Este efecto es debido al Gimbal Lock explicado en la Sección 4.5.1, por lo que el eje Z lo perdemos.

Las pruebas de movimiento han sido grabadas en vídeo e incluidas en el CD que acompaña a esta memoria. También se adjuntan los archivos de captura de los datos médicos y de movimiento de estas pruebas.

Indicar finalmente, con respecto, a la comodidad del prototipo, que a pesar de tratarse de un primer modelo, hay que indicar que es bastante cómodo y no molesta en ningún movimiento, prueba de ello son las pruebas de movimiento realizadas. En estas pruebas se ha podido caminar, correr y simular una caída.

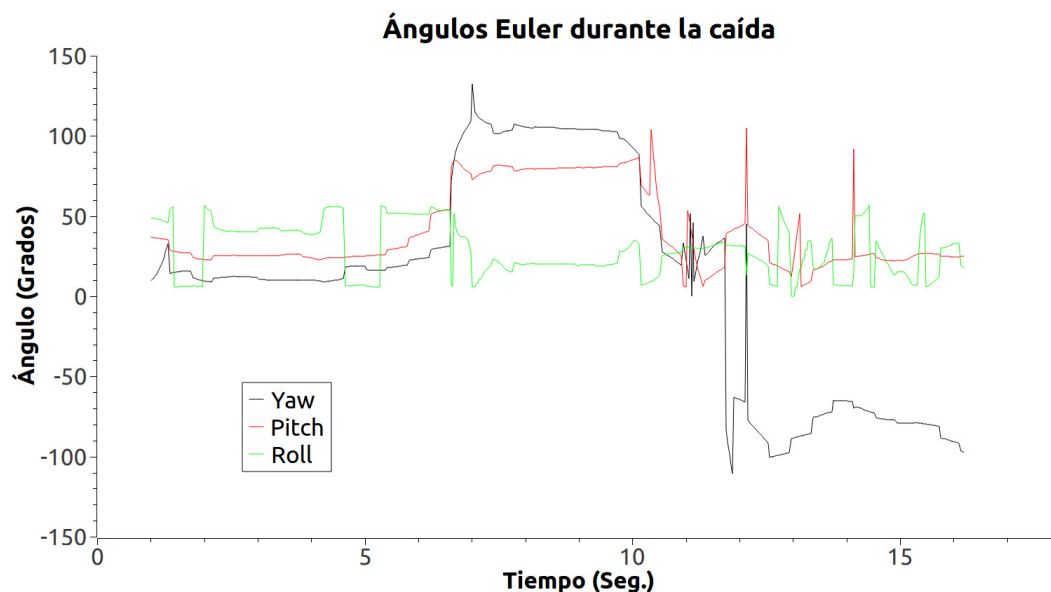


Figura 5.10: Gráfica representando los datos del giroscopio durante la simulación de una caída.

5.2. Consumo

Otro aspecto a evaluar de nuestro prototipo es el consumo de potencia. Este valor determinará la autonomía del prototipo. Si recopilamos todos los consumos de los sensores explicados en el Capítulo 4 podemos obtener un consumo teórico. En la Tabla 5.2 podemos ver el consumo teórico.

Dispositivo	Consumo
ECG	25 mW
Oximetría	125 mW
Temperatura	0,75 mW
Acelerómetro	20 mW
RTC	1,5 mW
Tarjeta SD	400 mW
MCU	150 mW
Total	722,25 mW

Tabla 5.2: Consumo teórico del prototipo.

Para comprobar si el consumo de nuestro prototipo está cerca de esos valores podemos usar un amperímetro. Colocando el amperímetro en serie con la alimentación de nuestro prototipo podemos obtener el consumo. Como podemos observar en la Figura 5.11 la corriente que absorbe, como máximo, nuestro prototipo es de 156 mA, como

está alimentado con 5 V, el consumo es de 780 mW. Este consumo tenemos que restar el consumo de los 5 diodos LED que usamos para funciones de depuración, por lo que el consumo final es de 680 mW.

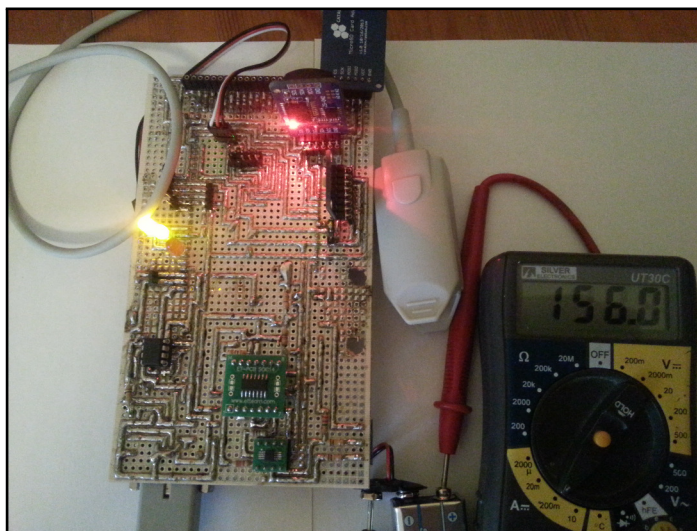


Figura 5.11: Consumo de nuestro prototipo.

5.3. Coste

Por último y no por eso menos importante hay que analizar el coste de nuestro prototipo. El objetivo siempre ha sido minimizar los costes. Como referencia para obtener los precios actuales se ha usado la página de la tienda online de Farnell². En la Tabla 5.3 se ha incluido una lista con los precios a fecha de 26 de junio de 2014 de la web citada.

Como se puede observar el precio final de los componentes usados en nuestro prototipo asciende a 75,01 €. A este precio hay que añadir el coste de la placa que varía según si tenemos capacidad para realizarla por nuestro medios o debemos adquirirla mediante algún servicio de prototipado. El precio de cada componente es el precio al adquirir una unidad del producto. En el caso de su producción masiva, el coste del prototipo sería muy inferior a estos precios, si bien es cierto que habría que considerar otros costes, como el de la propia producción.

²<http://es.farnell.com/>

Componente	Referencia	Precio unitario	Cantidad	Total
ATMEGA2560	1288330	15,36 €	1	15,36 €
MPU-6050	1864742	28,08 €	1	28,08 €
DS3231	2115248	7,83 €	1	7,83 €
OPA4340U	1212307	7,21 €	1	7,21 €
INA321EA	1206912	2,23 €	1	2,23€
Conector de alimentación	2293318	2,11 €	1	2,11 €
Portabatería CR2032	2115305	1,91 €	1	1,91 €
Conector SUB-D 9	1084697	1,42 €	1	1,42 €
Resistencia variable 10k	1784901	1,09 €	1	1,09 €
Conector microSD	2345279	0,81 €	1	0,81 €
Cristal resonador 16 MHz	2393158	0,59 €	1	0,59 €
NCP1117ST50T3G	2112617	0,50 €	1	0,50 €
LM358	7527007	0,50 €	1	0,50 €
Array de resistencias 10k	1377094	0,45 €	2	0,90 €
LP2985-33DBVR	2395925	0,43 €	1	0,43 €
Condensadores polarizados	2346368	0,25 €	2	0,50 €
Diodo M7	9843841	0,24 €	1	0,24 €
Condensador EUC0402K	9402063	0,19 €	10	1,90 €
Diodo 1N4148	8150192	0,10 €	2	0,20 €
Diodo CD1206-S01575	2211947	0,07 €	1	0,07 €
Resistencias R0603	2333548	0,03 €	31	0,93 €
Condensadores EUC0603	1813429	0,02 €	10	0,20 €
Total				75,01 €

Tabla 5.3: Lista de componentes y precios.

Si comparamos el coste de nuestro prototipo con los presentados en la Sección 3.2 del Capítulo 3 podemos observar que nuestro prototipo es mucho más barato. El único que se nos acerca a nuestro precio sería el E-Healt de Cooking Hack, que tiene un precio de 75 €, pero este dispositivo es solamente un shield de Arduino que de serie solo lee un ECG, mientras que nuestro dispositivo incluye microcontrolador, memoria, giroscopio, acelerómetro, etc.

Capítulo 6

CONCLUSIONES Y TRABAJO FUTURO

En este capítulo se resumen las conclusiones más importantes obtenidas tras la realización de este Trabajo Fin de Grado, y se indican algunas líneas de trabajo que podrían dar continuidad al mismo.

6.1. Conclusiones

El trabajo desarrollado ha requerido una importante dedicación tanto en lo que se refiere a la consulta de diversa documentación como a las propias tareas de diseño, realización y prueba del prototipo finalmente obtenido. Revisando el objetivo principal del trabajo, que fue expuesto en la Sección 2.1 del Capítulo 2, se puede concluir de una forma general que dicho objetivo ha sido cubierto de forma satisfactoria, en algún caso particular por encima de las previsiones iniciales.

Han sido muy numerosas las actividades realizadas y por tanto también las conclusiones particulares que se pueden extraer de ellas. Se indican a continuación las que consideramos más relevantes.

- Es posible obtener dispositivos de monitorización capaces de registrar un conjunto aceptable de parámetros clínicos utilizando componentes fácilmente accesibles y de bajo coste. Los diversos elementos que han dado forma al prototipo final se pueden adquirir, por ejemplo, en tiendas on-line. El coste total del prototipo no ha superado los 75 euros, pudiendo incluso haber sido menor.
- Hay diversas plataformas que permiten dar soporte al prototipo desarrollado. Además, en la mayoría de los casos, ofrecen suficiente soporte para encontrar la información necesaria para trabajar con ellas.

- Arduino es una plataforma muy adecuada para el desarrollo de este tipo de sistemas. Sus características y su coste son aspectos esenciales que hacen de ella una de las más utilizadas. Este hecho implica que hay una gran comunidad de usuarios y como consecuencia gran cantidad de información y software disponible.
- Existen en el mercado multitud de sensores capaces de recoger datos sobre diversos parámetros clínicos y mecánicos, algunos de los cuales han sido la base de este trabajo. En algunos casos no ofrecen datos directamente aprovechables y deben ser manejados adecuadamente. Así por ejemplo, se ha hecho necesario amplificar y filtrar algunas de las señales recogidas. De la misma forma, se ha requerido el uso de conversores analógicos/digitales.
- Algunos de los componentes disponibles, además de su funcionalidad específica, disponen de su propio procesador, que aunque tienen una capacidad limitada, permiten liberar de ciertos cálculos al microcontrolador. En nuestro prototipo ha sido el caso del componente utilizado para medir aceleraciones y posición angular.
- Es importante conocer que hay diversas formas de obtener el mismo dato correspondiente, por ejemplo, a un determinado parámetro clínico. Se trata de utilizar aquella que resulte más adecuada de acuerdo con las características del sistema a desarrollar. En este trabajo esto se ha tenido en cuenta a la hora de medir la presión arterial. Los requisitos exigidos al prototipo en cuanto a su tamaño, manejo y autonomía no permitían el uso de una bomba de aire debido a su tamaño y consumo. Por tanto, se ha optado por una medición indirecta basada en los pulsos del electrocardiograma y la oximetría.
- La autonomía es un aspecto esencial en este tipo de dispositivos, y por tanto hay que conseguir limitar el consumo y dejarlo en unos niveles mínimos que garanticen la total disponibilidad del sistema. En este sentido en este trabajo se han considerado eligiendo unos determinados componentes y no otros de mayor consumo.
- La mayor dificultad encontrada ha sido toda la información y conocimiento clínico que se debe asimilar para la realización de este proyecto. Por el contrario, la menor dificultad se ha tenido a la hora de realizar el software para el microcontrolador, debido a los conocimientos adquiridos en las asignaturas de Sistemas Empotrados y Adquisición de Señales.

6.2. Trabajo futuro

El sistema de monitorización desarrollado tiene una determinada funcionalidad que puede ser mejorada y ampliada. En este sentido se podrían iniciar algunas líneas de actuación a partir del trabajo ya realizado y que dieran continuidad al mismo. Por otro lado, este sistema es un prototipo y por tanto también caben ciertas actuaciones encaminadas a determinar la viabilidad de un futuro producto basado en el mismo. El prototipo puede considerarse un sistema genérico en el sentido de que puede ser la base de otros sistemas parecidos en su concepción, y diferentes en su funcionalidad. Por tanto, pueden surgir otros trabajos a partir de éste al considerar el ámbito de aplicación del mismo.

Se indican algunas de las tareas que podrían realizarse a partir del prototipo desarrollado, agrupándolas de acuerdo con la vía que conduciría dichas tareas:

1. Mejoras y ampliación de la funcionalidad:

- El afinamiento del algoritmo sería una mejora ya que ayudaría a detectar mejor el pulso de los pacientes.
- Se pueden añadir sensores que registren nuevos parámetros clínicos o de otro tipo. Por ejemplo, se podría añadir un glucómetro que nos permitiría medir el nivel de glucosa del paciente.
- Sería fácilmente incorporable un sistema de comunicaciones (red móvil, wifi, rf) con el fin de transmitir los datos en tiempo real.
- Se podría incorporar un GPS para obtener información en todo momento de la posición de la persona que lleve el dispositivo.

2. Desarrollo de productos:

- Realizar un proceso de validación homologado de la funcionalidad del sistema. Para poder utilizar este tipo de dispositivos en entornos médicos esta validación es un requisito.
- Reducir el tamaño, y el consumo. Esto se puede conseguir utilizando tecnología SMD, y algunos cambios en el diseño, eliminando ciertos elementos y tomando medidas de formas alternativas.

3. Aplicabilidad:

- Si usamos sólo la parte de ECG del proyecto podemos realizar un detector de arritmias o/y infartos de miocardio, este trabajo sería muy útil junto

un sistema de comunicación y un GPS para personas con un alto riesgo de parecer problemas cardiacos, ya que el dispositivo avisaría a los sistemas de emergencia en caso de alerta mandando la posición del afectado junto al aviso.

- Otro proyecto muy parecido al anterior sería uno con sólo el módulo de oximetría. En este caso detectaríamos un estado de hipoxia y podríamos lanzar alertas en caso de detectarlas. Este proyecto sería útil, por ejemplo, para alpinistas ya que al pasar de 3.000 metros algunas personas ya pueden padecer síntomas de hipoxia.
- Si se usa el acelerómetro y el giroscopio podemos realizar un detector de caídas, pudiendo guardar las magnitudes biológicas y realizar un análisis posterior para averiguar el motivo de la caída. Si tenemos incorporados el módulo de comunicaciones y de GPS podemos enviar una alerta para una intervención de los equipos de emergencia.
- En otros ámbitos distintos al de la salud también sería aprovechable. Por ejemplo, con pequeñas modificaciones se pueden realizar “trackers” de vehículos, monitorización de herramientas, etc. El límite en este caso es la imaginación.

Bibliografía

- [1] F. J. Medina Aragón y J. Gómez Salgado. *Fundamentos teóricos-prácticos de la enfermería de emergencias*. Fundación para el desarrollo de la enfermería, 2009. [19](#)
- [2] C. Codagnone y F. Lupiañez-Villanueva. Benchmarking deployment of ehealth among general practitioners 2013. Informe técnico, European Commission, 2013. [1](#)
- [3] European Commission. Green paper on mobile health ("mhealth"). Informe técnico, European Commission, 2014. [2](#)
- [4] Fundación FAE. *Manual de técnico en cuidados auxiliar de enfermería*. Fundación FAE, 2007. [18](#), [32](#)
- [5] H. Gesche, D. Grosskurth, G. Kuchler y A. Patzak. Continuous blood pressure measurement by using the pulse transit time: comparison to a cuff-based method. *European Journal of Applied Physiology*, 2010. [30](#)
- [6] J.S. Gilson, N. J. Holter y W.R. Glasscock. Clinical observations using the electrocardiographic-avsep continuous electrocardiographic system: tentative standards and typical patterns. *Ambulatory Journal Cardiol*, 1964. [10](#), [11](#)
- [7] J.S. Gilson, N. J. Holter y W.R. Glasscock. Unusual qrs alterations seen in continuous electrocardiogram recordings. *Ambulatory Heart Journal*, 1965. [10](#)
- [8] A. Antequera Guirado. *Temario oposiciones de enfermería: Servicio de Salud de la Comunidad de Madrid*. Fuden Formación, 2011. [29](#)
- [9] N. J. Holter. Radioelectrocardiography: a new technique for cardiovascular studies. *Ann New York Acad Sci*, 1957. [9](#)
- [10] N. J. Holter. New method for heart studies. *SCIENCE*, 134: 1214–1220, Octubre 1961. [9](#), [10](#)
- [11] N. J. Holter y J. A. Gengerelli. Remote recording of physiological data by radio. *Rocky Mountain Medical Journal*, 1949. [9](#)

- [12] H. L. Kennedy. Ambulatory electrocardiographic including holter recording technology. *Lea & Febiger*, 1981. [10](#)
- [13] M. Margolis. *Arduino Cookbook*. O'Reilly Media, Inc, 2011. [21](#)
- [14] Y. Mendelson. “Pulse Oximetry”, in *Wiley Encyclopedia of Biomedical Engineering*. 2006. [ix](#), [25](#)
- [15] A. Goicolea de Oro y C. Perezagua. *Monitorizacion ambulatoria del electrocardiograma en adultos jovenes sin cardiopatía aparente*. Tesis Doctoral, Universidad Complutense de Madrid, 2005. [9](#)
- [16] E. Pérez de la Plaza y A. M. Fernández Espinosa. *Auxiliar de Enfermería*. McGraw Hill, 2011. [18](#), [31](#)
- [17] J. I. Valle Racero. Electrocardiografía dinámica. 2010. [14](#)
- [18] R. Sabes-Figuera y I. Maghiros. European hospital survey: Benchmarking deployment of e-health services (2012-2013). Informe técnico, European commission, 2013. [1](#)
- [19] J. M. Pérez Santana, R. M. Morillo Cadierno y E. Puertas Calderón. *Enfermeras/os: Servicio de Salud de la Comunidad de Madrid*. MAD, 2012. [25](#), [33](#)
- [20] J. E. Sinex. Pulse oximetry: Principles and limitations. *The American Journal of Emergency Medicine*, 17: 59–66, 1999. [ix](#), [26](#)
- [21] J. G. Webster. Design of pulse oximeters, series in medical physics and biomedical engineering. *Boca Raton: CRC Press*, 1997. [24](#), [26](#)

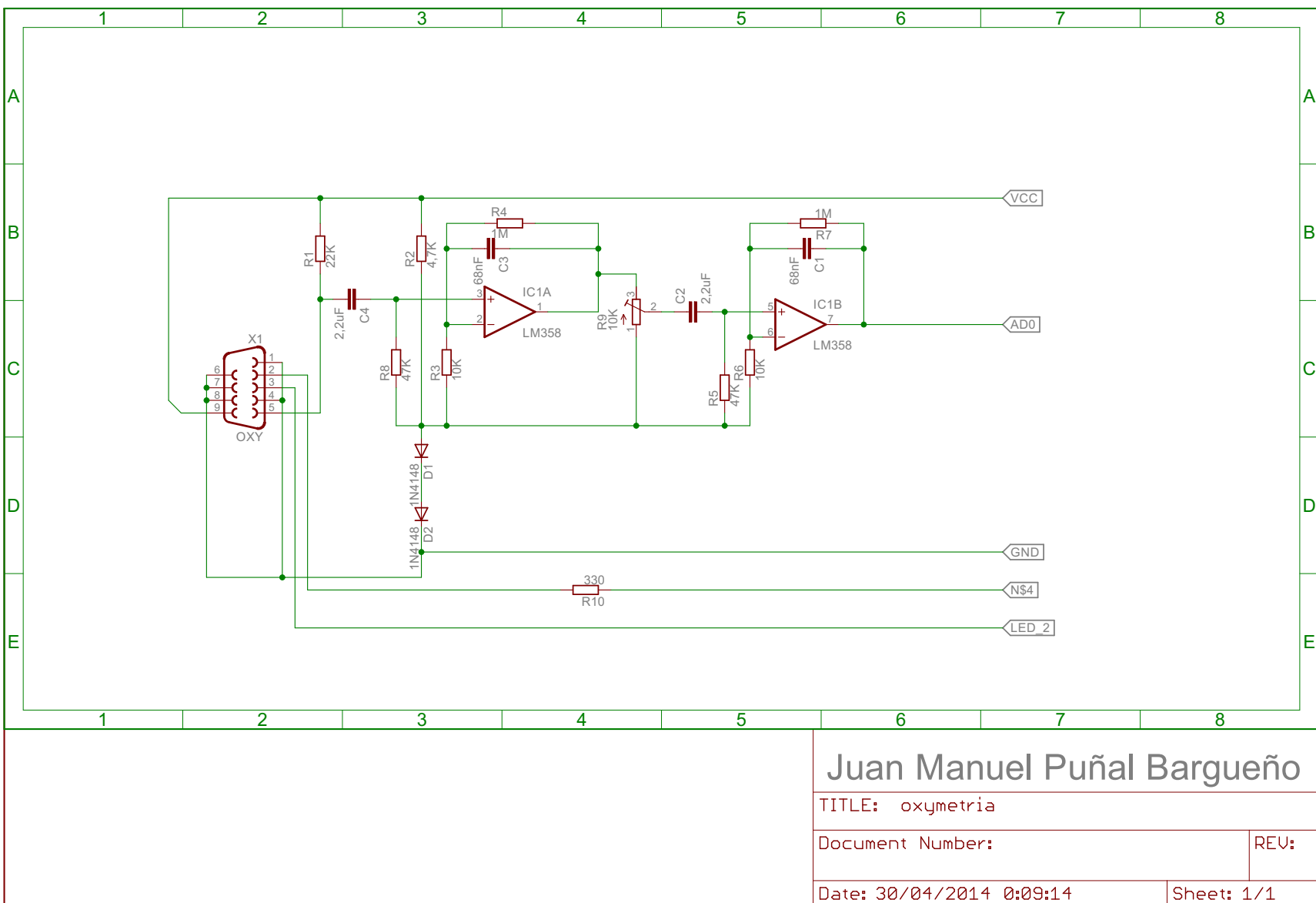
CONTENIDO DEL CD

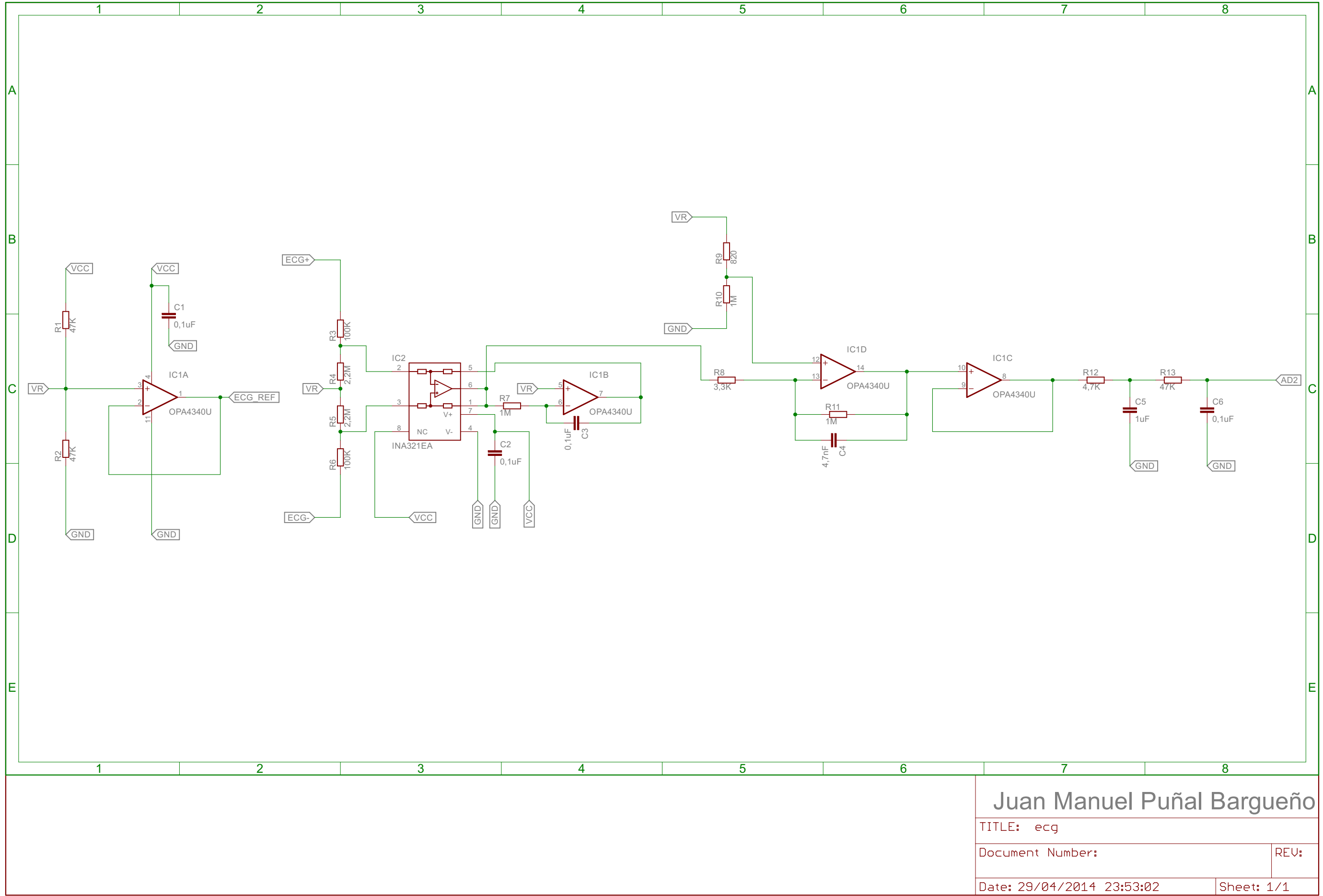
En el CD que acompaña a la memoria podemos encontrar los siguientes recursos:

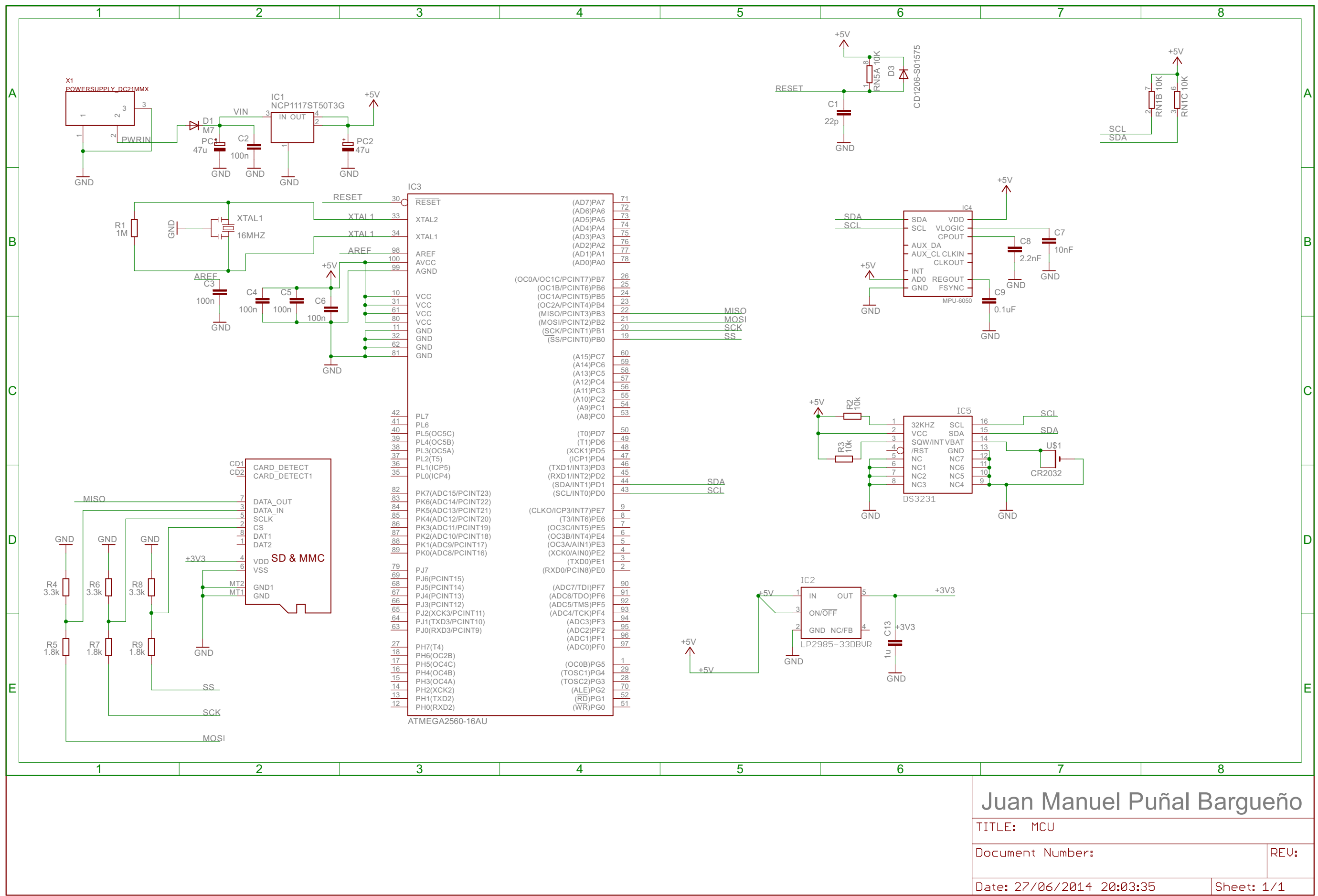
- Memoria del trabajo en formato PDF dentro del directorio Memoria.
- Código fuente del trabajo dentro del directorio Código fuente.
- Libros y artículos a los que se ha hecho referencia durante la memoria y que se han utilizado como bibliografía. Se pueden encontrar en el directorio Bibliografía.
- Todos los archivos generados durante las pruebas. Están en el directorio Pruebas.
- Todos los archivos generados con las herramientas para realizar las simulaciones. Localizados en el directorio Simulaciones.
- Los esquemas eléctricos en formato Eagle, situados en el directorio Esquemas.

Anexo A

ESQUEMAS ELÉCTRICOS







Juan Manuel Puñal Bargaño

TITLE: MCU

Document Number:

REV:

Date: 27/06/2014 20:03:35

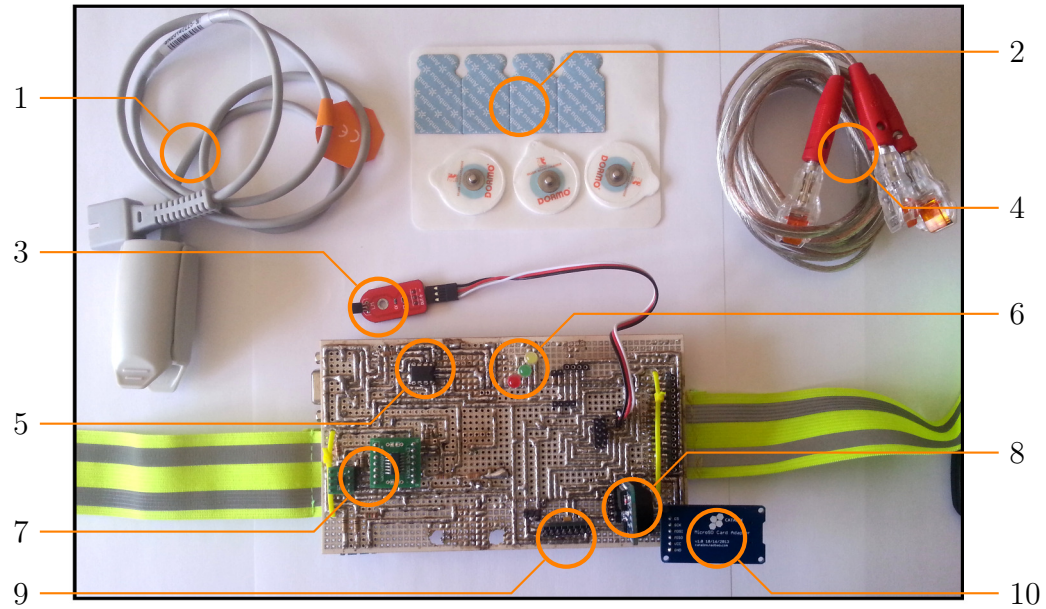
Sheet: 1/1

Anexo B

PARTES DEL PROTOTIPO

1. Sonda pulsioximétrica.
2. Electrodo ECG .
3. Sensor temperatura.
4. Cables ECG.
5. Amplificador operacional de la oximetría.
6. Diodos LED para depuración.
7. Amplificadores operacionales del ECG.
8. RTC.
9. Acelerómetro/Giroscopio.
10. Tarjeta MicroSD.
11. Conector del ECG.
12. Conectores para la comunicación entre la placa del microcontrolador y el prototipo.
13. Conector de la sonda pulsioximétrica.
14. Microcontrolador Atmega2650.
15. Conector de alimentación.

Parte delantera y accesorios



Parte trasera

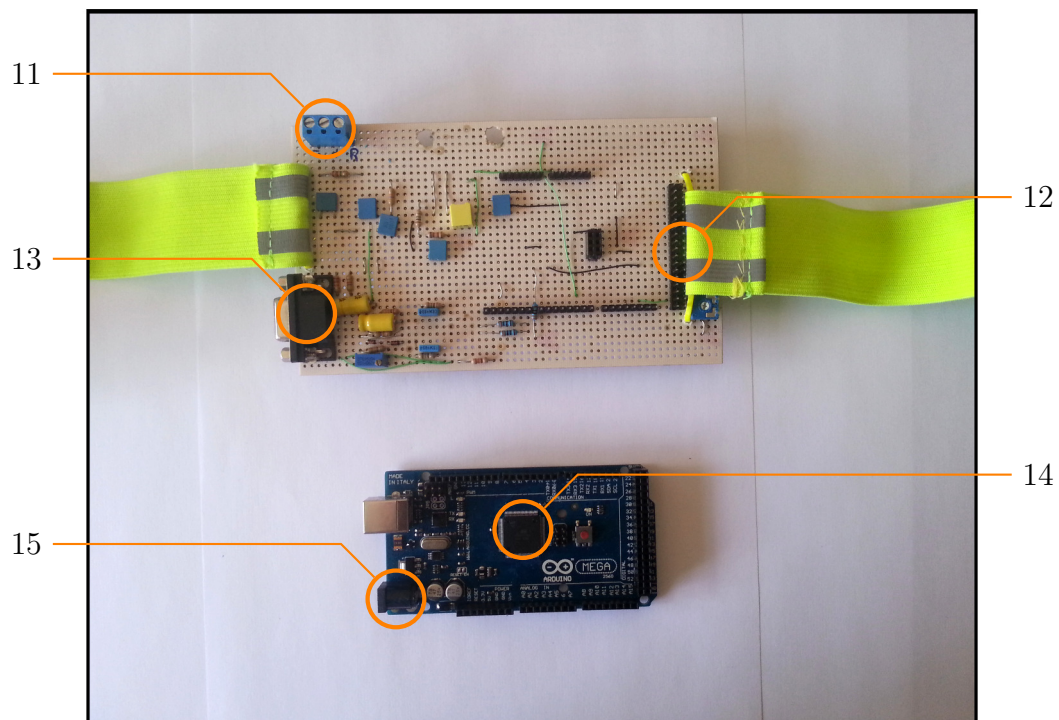


Figura B.1: Partes del prototipo.

Anexo C

MANUAL DE USUARIO

En este anexo se ofrece un pequeño manual de usuario para la puesta en marcha del prototipo realizado en este trabajo. El manual está dividido en secciones para una rápida y sencilla consulta.

C.1. Preparación del prototipo

1. Conectar los terminales del electrocardiograma con ayuda de un destornillador para que queden bien sujetos, tal como se indica en la Figura C.1.

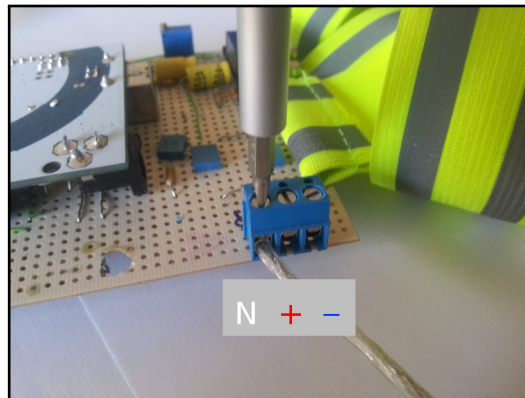


Figura C.1: Conexión del ECG.

2. Conectar la sonda pulsioximétrica, ver Figura C.2.

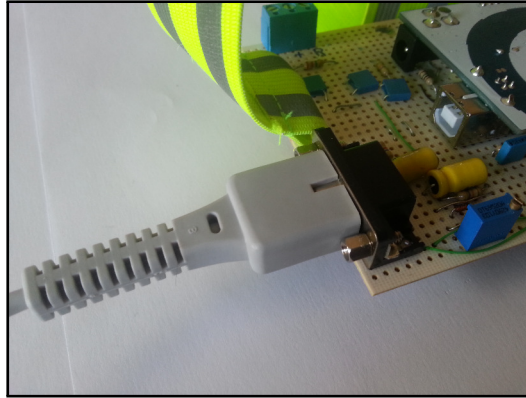


Figura C.2: Conector de la sonda pulsioximétrica.

3. Colocar los electrodos al paciente, como en la Figura C.3 , rasurando el vello en el caso necesario y limpiando la piel mediante alcohol antiséptico.

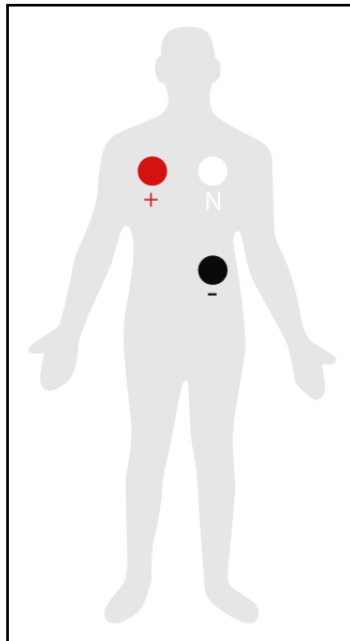


Figura C.3: Posición de los parches de la ECG.

4. Colocar la sonda pulsioximétrica en el paciente, ver Figura C.4.



Figura C.4: Colocación de la sonda pulsioximétrica en el paciente.

5. Insertar una tarjeta micro SD con un sistema de archivos FAT32, como se muestra en la Figura C.5.

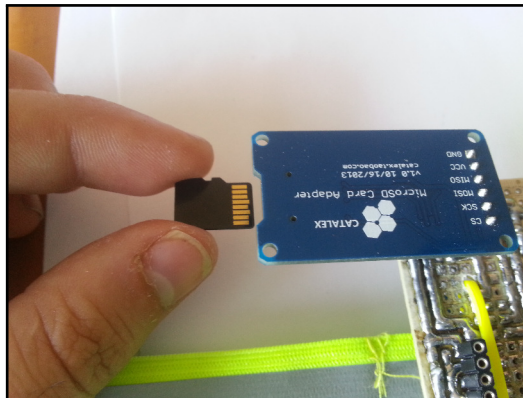


Figura C.5: Colocación de la tarjeta SD.

6. Colocar el prototipo en el paciente tal como indica la Figura C.6.

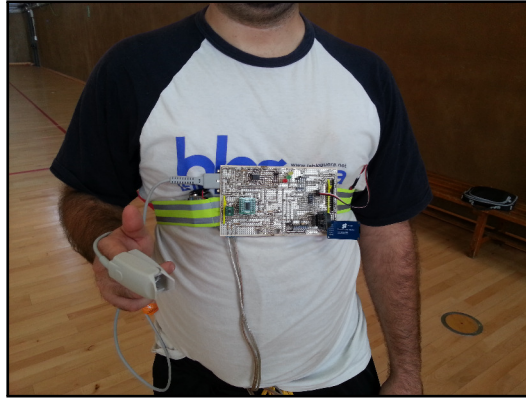


Figura C.6: Colocación del prototipo en el paciente.

C.2. Puesta en marcha del prototipo

7. Para iniciar el prototipo conectar la alimentación como indica la Figura C.7.

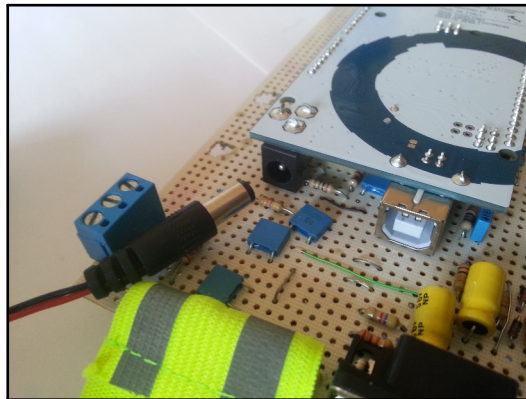


Figura C.7: Conector de alimentación de prototipo.

8. A los pocos segundos de alimentar el prototipo, los diodos LED de depuración, representados en la Figura C.8, comenzarán a parpadear. El LED amarillo significa que se ha realizado una conversión en el conversor analógico/digital, el LED verde indica que el acelerómetro ha realizado un cálculo en su procesador avisando de esto al microcontrolador, este suceso graba los datos enviados del acelerómetro en la tarjeta SD y por último el diodo LED rojo indica que se ha grabado un dato médico en la tarjeta SD.

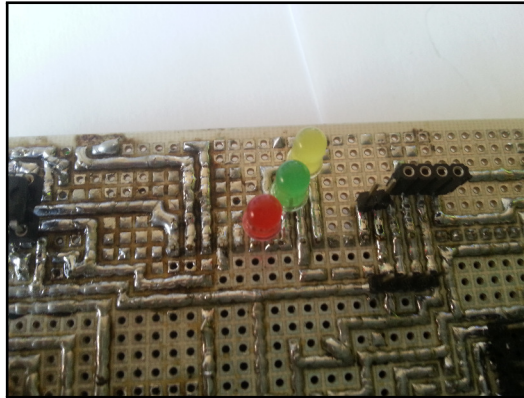


Figura C.8: Diodos LED de depuración.

9. Para desconectar el prototipo solo debemos desenchufar la alimentación, ningún dato se perderá ya que todas las grabaciones se realizan directamente sin usar ninguna memoria intermedia.

C.3. Extracción de datos

10. Para extraer los datos de la tarjeta SD es necesario un lector de tarjetas, como por ejemplo el mostrado en la Figura C.9. Este tipo de lectores nos permite leer los ficheros contenidos en la tarjeta SD en cualquier computador que disponga de puerto USB.



Figura C.9: Lector de tarjetas.

