

HISTORIAL DIGITAL DE DATOS MÉDICOS

SALINAS PEDROZA HENRY JOSE

**UNIVERSIDAD DE SAN BUENAVENTURA
FACULTAD DE INGENIERIA
PROGRAMA DE ELECTRÓNICA
2009**

HISTORIAL DIGITAL DE DATOS MEDICOS

SALINAS PEDROZA HENRY JOSE

**Proyecto de grado como requisito
para optar por el título de ingeniero electrónico**

Asesor:

Ing. Amparo Contreras

**UNIVERSIDAD DE SAN BUENAVENTURA
FACULTAD DE INGENIERIA
PROGRAMA DE ELECTRÓNICA
2009**

TABLA DE CONTENIDO

INTRODUCCION	16
PLANTEAMIENTO DEL PROBLEMA	18
ANTECEDENTES.....	18
1.1.1 MONITOREO DE SIGNOS VITALES POR MEDIO DE COMPUTADORAS.....	18
1.1.2. LAS COMPUTADORAS AYUDAN A SUMINISTRAR TERAPIAS.	19
1.1.2.1. VENTILADORES ARTIFICIALES.	19
1.1.3. MEDNOTES.	19
1.1.4. BODYMETRIX.....	20
1.1.5. DATAMED.	21
1.2 DESCRIPCIÓN Y FORMULACIÓN DEL PROBLEMA.....	22
1.3 JUSTIFICACIÓN.....	23
1.4 OBJETIVOS DE LA INVESTIGACIÓN.....	244
1.4.1. OBJETIVO GENERAL.....	244
1.4.2. OBJETIVOS ESPECÍFICOS.....	244

1.5 ALCANCES Y LIMITACIONES DEL PROYECTO	24
1.5.1 ALCANCES.....	24
1.5.2. LIMITACIONES..	24
2. MARCO DE REFERENCIA	26
2.1 MARCO TEÓRICO CONCEPTUAL.....	26
2.1.1. UNIDADES DE INFORMACIÓN.	26
2.1.1.1. BIT (BINARY DIGIT O DÍGITO BINARIO).	26
2.1.1.2. CÓDIGO ASCII	26
2.1.2. SISTEMA BINARIO.	27
2.1.2.1. MEDIDAS DE ALMACENAMIENTO DE LA INFORMACIÓN.....	28
2.1.3. DISPOSITIVO DE ALMACENAMIENTO.....	28
2.1.4. SISTEMAS DE ARCHIVO.....	29
2.1.4.1. DISCO DURO.....	29
2.1.4.2. UNIDAD DE CD-ROM.	30
2.1.4.3. UNIDAD DE CD-RW (REGRABADORA).	31

2.1.4.4. UNIDAD DE DVD-ROM.	31
2.1.4.5. UNIDAD DE DVD-RW.	32
2.1.4.6. MEMORIA USB, PENDRIVE O USB FLASH DRIVE..	32
2.1.4.7. UNIDADES MAGNETO-ÓPTICAS.	33
2.1.4.8. OTROS DISPOSITIVOS DE ALMACENAMIENTO	33
2.1.4.9. EEPROM.	34
2.1.4.10. MEMORIAS SD..	35
2.1.4.11. ALGUNAS CARACTERÍSTICAS INTERESANTES.	36
2.1.4.12. OTRAS FUNCIONES.	37
2.1.5. NUEVAS TECNOLOGÍAS EN SISTEMAS DE ALMACENAMIENTO.....	38
2.1.5.1. SOLUCIONES A GRAN ESCALA.....	38
2.1.5.2. MICROCONTROLADOR.....	38
2.1.5.3. SENSOR.....	40
2.2 MARCO LEGAL O NORMATIVO.....	40
3. METODOLOGÍA.....	42

3.1 ENFOQUE DE LA INVESTIGACIÓN	42
3.2 LÍNEA DE INVESTIGACIÓN DE USB / SUB-LÍNEA DE FACULTAD / CAMPO TEMÁTICO DEL PROBLEMA.....	42
3.3 TÉCNICAS DE RECOLECCIÓN DE INFORMACIÓN	42
3.4 HIPÓTESIS	43
3.5 VARIABLES.....	43
3.5.1. VARIABLES INDEPENDIENTES.....	43
3.5.2. VARIABLES DEPENDIENTES.....	43
4. DESARROLLO INGENIERIL	44
4.1 ETAPA DE CONTROL.....	45
4.1.1 SENSOR DE TEMPERATURA.....	46
4.1.2 Sensor de frecuencia cardiaca.....	47
4.1.3 Memoria Interna.....	47
4.1.4 Visualización.....	47
4.1 DISEÑO DEL MONITOR DE FRECUENCIA CARDIACA.....	47
4.1.1. TRANSDUCTOR.....	48

4.1.2. ACOUPLE DE IMPEDANCIAS Y AMPLIFICADOR CON GANANCIA.	49
4.1.3. AMPLIFICADOR DEL INSTRUMENTO.	50
4.1.4. FORMA DE ONDA A LA SALIDA DEL AMPLIFICADOR DE INSTRUMENTACIÓN.	53
4.1.5. FILTROS PASA BAJO Y PASA ALTO.	53
4.1.6. FILTRO PASA BAJO.	54
4.1.7. FILTRO PASA ALTO.	55
4.1.8 ECUACIÓN FILTRO PASA ALTO.	56
4.3 PLANOS FINALES.	59
4.4 DIAGRAMA DE FLUJO ETAPA DIGITAL.	64
4.4.1 Mensaje de Presentación.	64
4.4.2 Enter.	64
4.4.3 SELECCIONE OPCION.	64
4.3 Up Down.	64
4.3 Comenzar Sesión.	65

5. CONCLUSIONES	108
6. RECOMENDACIONES.....	109
BIBLIOGRAFÍA.....	110
WEBLIOGRAFÍA.....	110
GLOSARIO.....	111

HISTORIAL DIGITAL DE DATOS MÉDICOS

NOTA ASIGNADA

FIRMA DE JURADO

FIRMA DE JURADO

21-04-2009 BOGOTA

INTRODUCCIÓN

La presente investigación se refiere a un dispositivo diseñado de forma digital, mediante un proceso de programación de un microcontrolador; este dispositivo será de gran ayuda en un centro de atención médica para disminuir el horario y agilizar las actividades.

La característica principal de este dispositivo es que brindará al usuario eficiencia en labores cotidianas, como la medición de algunos signos vitales.

Para analizar esta problemática es necesario mencionar sus causas. Una de ellas es reducir el índice de morbi- mortalidad, pues debido al gran volumen de pacientes que maneja cualquier servicio en un hospital y en algunas ocasiones la escasa cantidad de personal asistencial, con relación al número de pacientes (pacientes vs enfermeros), se ha logrado evidenciar un gran número de complicaciones médicas y quirúrgicas como consecuencia de la escasa vigilancia hemodinámica (frecuencia cardíaca y **temperatura**), necesaria para valorar la mejoría clínica del paciente, haciéndose más difícil para el personal de la salud, recordar hallazgos y posteriores pautas de manejo.

La investigación y desarrollo de este dispositivo se realiza por el interés de agilizar y evitar los conflictos que se presentan en la ciencia médica, debido a la interacción en diversas actividades del cuerpo médico.

En el marco de la metodología digital la investigación se realizó mediante procesos de programación de toma de datos y su respectiva decodificación o muestreo al usuario, teniendo en cuenta sugerencias tomadas en entrevistas a médicos en centros asistenciales como Saludcoop y la Clínica Juan N. Corpas, sugerencias analizadas para la solución al problema.

Las entrevistas se realizaron también al personal de enfermería, quien es el encargado de hacer la toma de datos en variables, trabajadas en la investigación (frecuencia cardíaca y temperatura), entre otras.

El objetivo principal es proponer y desarrollar una idea sencilla pero impactante que sea capaz de aportar algo útil a la sociedad, qué mejor cuando se habla de la vida misma, donde se desarrolla un dispositivo el cual permite al personal de la

salud tener un mejor control estadístico de los pacientes desde mecanismos digitales que ofrecen mayor confiabilidad de la información frente a mecanismos manuales de simple observación humana.

1. PLANTEAMIENTO DEL PROBLEMA

1.1 ANTECEDENTES

En el ámbito de la medicina existen numerosos dispositivos que procesan información y la archivan teniendo en cuenta diferentes características, como la historia clínica completa de cada paciente.

La práctica médica nunca ha sido fácil, pues en realidad se debe tener en la memoria una inmensa cantidad de información. Por cierto, más de alguna vez se tiene que acudir a alguna libreta, libro o colega para recordar algunos detalles.

1.1.1 Monitoreo de signos vitales por medio de computadoras. Las aplicaciones científicas que han nacido con las computadoras se basan en cálculos complejos. En la sala de urgencias se utilizan sistemas que analizan en fracción de segundos toda la información sobre los signos vitales del paciente.

- **Frecuencia cardíaca.** Actividad eléctrica del corazón en un tiempo determinado.
- **Pulso.** Presión sanguínea, transmitida a través de las arterias.
- **Saturación de oxígeno.** Nivel de oxígeno en sangre.
- **Frecuencia respiratoria.** Actividad de inspiración y expiración en un tiempo específico.
- **Presión arterial sistémica.** Presión con la que el corazón impulsa la sangre, para distribuirla, en contra de la resistencia de las arterias.

Si el paciente se encuentra grave, introduciendo un catéter¹ en el corazón, puede medirse su presión arterial pulmonar, lo cual permite contar con parámetros fisiológicos más exactos para el manejo de este tipo de pacientes. Esto es posible gracias a las computadoras, ya que el catéter es conectado a un transductor que convierte la fuerza hidráulica (presión) a un estímulo eléctrico que es enviado al chip contenido en el monitor que a su vez lo traduce en una curva (imagen) o en cifras.

1.1.2 Las computadoras ayudan a suministrar terapias.

1.1.2.1 Ventiladores artificiales. En las salas de urgencias existen también equipos diseñados específicamente para proporcionar ventilación a los enfermos que así lo requieren; es decir, son oxigenadores artificiales que apoyan a la respiración del paciente o la sustituyen en su totalidad. Estos cuentan con programas de software que facilitan los diversos procedimientos terapéuticos como son:

- **Control del volumen de aire administrado.** Modifica la cantidad de aire administrado en cada inspiración.
- **Frecuencia respiratoria.** Controla el número de ciclos inspiración y expiración en una unidad del tiempo.
- **Porcentaje de oxígeno suministrado.** El aire puede contener una cantidad variable de oxígeno, la cual puede ir del 0.21% al 100%, para ser inducido al paciente.
- **Aspersión de medicamentos.** Es un sistema para la administración de fármacos por medio del aire. Los aparatos controlan la presión aérea en un circuito cerrado de ventilación.

1.1.3 MedNotes. Ayuda a los médicos y administradores de muchas disciplinas médicas a alcanzar sus metas clínicas y administrativas gracias a la eficiencia compartida del software y del hardware. El usuario profesional sigue rutinas; cono-

¹ Catéter: es, en medicina, un dispositivo que puede ser introducido dentro de un tejido o vena. Los catéteres permiten la inyección de fármacos, el drenaje de líquidos o bien el acceso de otros instrumentos médicos.

ce en tiempo real y en detalle la historia clínica de los pacientes, sus alergias y reacciones peligrosas, sus diagnósticos, crea y almacena notas y registros médicos, se comunica con otros profesionales o expertos, tiene acceso a bibliografía especializada, etc. a través de un dispositivo móvil inalámbrico tipo tablet PC con conexión WIFI, dotado de una interfase amigable e intuitiva. Ver figura 1.

Figura 1. Agenda digital Mednotes.



Fuente: <http://www.mrandersonmd.com/2006/07/08/aplicaciones-médicas-para-tu-palm-i/> - 02-10-2008 6:20 PM

1.1.4 Bodymetrix. Contiene una base de datos para almacenar todos los datos personales de los pacientes médicos, incluyendo la fotografía de cada uno de ellos. Este dispositivo está diseñado para la carga fácil y rápida de las mediciones. Si lo desea, puede cargar las mediciones en una palm o pocket PC² o bien en cualquier otro PC a través de una planilla de cálculos que se entrega con el pro-

² pocket PC: es un ordenador de bolsillo, también llamado PDA (Personal Digital Assistant). Se trata de un pequeño ordenador, diseñado para ocupar el mínimo espacio y ser fácilmente transportable que ejecuta el sistema operativo Windows CE de Microsoft entre otros, el cual le proporciona capacidades similares a los PCs de escritorio.

grama, para que luego Bodymetrix incorpore automáticamente las mediciones y proceda con el cálculo correspondiente. Ver figura 2.

Figura 2. Bodymetrix.

	1	2	3
Tríceps	10	10	
Subescapular	15	15	
Bíceps	8	7	
Cresta iliaca	15.5	16.5	
Supraespinal	16	16.5	
Abdominal	17.2	17.8	
Muslo Frontal	15.2	16.1	
Pantorrilla	9.7	9.9	
Pecho	11	11.2	

Fuente: <http://www.mrandersonmd.com/2006/07/08/aplicaciones-medicas-para-tu-palm-i/>
02-10-2008- 6:30 pm

1.1.5 Datamed. Contiene una base de datos para almacenar todos los datos personales de los pacientes médicos, incluyendo la fotografía de cada uno de ellos. Ver figura 3.

Figura 3. Datamed



Fuente: <http://www.datamed.com/2006/07/08/aplicaciones-medicas-para-tu-palm-i/>
05-10-2008- 2:30 pm

1.2 DESCRIPCIÓN Y FORMULACIÓN DEL PROBLEMA.

El tiempo que se demora una enfermera o el cuerpo médico haciendo mediciones y anotando datos a cada uno de los pacientes en un centro médico manualmente, sumado al que demora hasta archivarlo; son la razón principal por la cual los médicos pierden tiempo en un centro asistencial de medicina. En consecuencia aumenta la extensión de horario.

La revisión de signos vitales se hacía, anteriormente, mediante mediciones manuales y requería mucho tiempo. La frecuencia cardíaca, por ejemplo, se medía tomando el pulso en la muñeca del paciente o escuchando al corazón por medio de un estetoscopio. Actualmente, para conocer rápidamente toda esta información, el personal médico sólo tiene que conectar al paciente a un sistema de monitoreo, con cables y dispositivos de aplicación externa que no causan ninguna molestia adicional al paciente.

La gran confiabilidad y exactitud de estos monitores, permite optimizar tiempo, que es tan valioso en la sala de urgencias y se traduce en la posibilidad de salvar vidas.

El estrés que maneja el cuerpo médico en un centro asistencial aumenta en la medida que crece el número de actividades, se esta hablando de: urgencias, consultas, etc. Dicho estrés hace que en ocasiones se descuiden funciones de suma importancia para la secuencia, el seguimiento y observación en un paciente como son la toma en los signos vitales, por ello, no se están supliendo las necesidades del consumidor.

En Colombia la carencia de instrumentos, los cuales ayudan a la medición y el diagnóstico de algunas variables del cuerpo, para cubrir necesidades del personal médico en los Hospitales públicos, de acuerdo con algunas entidades como la Asociación de Usuarios de Servicios de Salud Asudessa y la asociación de Usuarios del Instituto de Seguros Sociales de Antioquia, ya que en diversos hospitales públicos no se alcanza a suplir la revisión médica en todos los pacientes.

1.3 JUSTIFICACIÓN

La creación de estos dispositivos se hace necesaria, no sólo a nivel tecnológico, sino que es de gran ayuda para el ámbito médico y científico, permitiendo la recolección de datos exactos y ágiles, ofreciendo de una forma efectiva y segura la actividad informativa del estado de los pacientes.

Cabe anotar que este tema no sólo interesa a los ingenieros y médicos, sino también a las personas, en general, como los pacientes que serán los más beneficiados con esta investigación.

Este proyecto se desarrolla con la implementación de un dispositivo, que ofrece una reducción de actividades en un centro médico y minimiza el trabajo del personal médico, volviendo cómodo y agradable el horario de trabajo brindando además un aporte en la recuperación de los pacientes, puesto que se tendrá más en cuenta su estado, sin importar la gravedad, en favor a ello el dispositivo brindará la comodidad al usuario y la exactitud al médico

1.4 OBJETIVOS DE LA INVESTIGACIÓN

1.4.1 Objetivo general.

Diseñar un dispositivo que mediante un sistema sensórico tome datos de algunos signos vitales en un número determinado de horas, para luego llevar una historia médica mediante procesos digitales como la programación, digitalización de señales medidas y visualización de las mismas.

1.4.2 Objetivos específicos.

- Analizar la fisiología de las constantes vitales del cuerpo (frecuencia cardíaca) involucradas en la investigación.
- Crear un sistema digital mediante procesos de programación en el cual se pueda archivar el diagnóstico de cada paciente valorado.
- Programar el microcontrolador que se utilizará en el dispositivo para que este muestre los datos de las variables medidas en una pantalla LCD.

1.5 ALCANCES Y LIMITACIONES DEL PROYECTO

1.5.1 Alcances. El proyecto culmina con el desarrollo de un dispositivo que tome los signos vitales en pacientes, para lograr a la obtención de las variables de la **temperatura** y frecuencia cardíaca, con el fin de registrar en dicho dispositivo los datos medidos para disponer de una base de datos actualizada que hace posible llevar un control médico del paciente y un ahorro considerable de tiempo, ya que el ingreso de datos se efectúa en forma directa y evita los errores que se generan manualmente, mediante un software capaz de tomar señales de signos vitales en el cuerpo humano, procesarlas, archivarlas, y convertirlas en lectura digital y un hardware constituido por dos sensores que serán los instrumentos que interactúan en cada paciente y miden las señales deseadas.

1.5.2 Limitaciones. Los pacientes necesitan las mediciones de otros signos vitales, que son de gran importancia para la recuperación en cada uno de ellos, entre los que se tienen la presión sanguínea, el encefalograma, los cuales no serán te-

nidos en cuenta ya que eleva considerablemente los costos del proyecto. Por esta razón el estudio es enfocado a la medición de las variables anteriormente descritas.

2. MARCO DE REFERENCIA

2.1 MARCO TEÓRICO CONCEPTUAL

2.1.1 Unidades de información.

2.1.1.1 Bit (Binary Digit o dígito binario). Adquiere el valor 1 ó 0 en el sistema numérico binario. En el procesamiento y almacenamiento informático un bit es la unidad de información más pequeña manipulada por el ordenador y está representada físicamente por un elemento como un único pulso enviado a través de un circuito, o bien como un pequeño punto en un disco magnético capaz de almacenar un 0 o un 1. La representación de información se logra mediante la agrupación de bits para lograr un conjunto de valores mayor que permite manejar mayor información. Por ejemplo, la agrupación de ocho bits compone un byte que se utiliza para representar todo tipo de información, incluyendo las letras del alfabeto y los dígitos del 0 al 9.

2.1.1.2 Código ASCII (American Standard Code for Information Interchange o Código Estándar Americano para el Intercambio de Información). Esquema de codificación que asigna valores numéricos a las letras, números, signos de puntuación y algunos otros caracteres. Al normalizar los valores utilizados para dichos caracteres, ASCII permite que los ordenadores o computadoras y programas informáticos intercambien información.

Incluye 256 códigos divididos en dos conjuntos, estándar y extendido, de 128 cada uno. Estos conjuntos representan todas las combinaciones posibles de 7 u 8 bits, siendo esta última el número de bits en un byte. El conjunto ASCII básico, o estándar, utiliza 7 bits para cada código, lo que da como resultado 128 códigos de caracteres desde 0 hasta 127 (00H hasta 7FH hexadecimal). El conjunto ASCII extendido utiliza 8 bits para cada código, dando como resultado 128 códigos adicionales, numerados desde el 128 hasta el 255 (80H hasta FFH extendido).

En el conjunto de caracteres ASCII básico, los primeros 32 valores están asignados a los códigos de control de comunicaciones y de impresora, caracteres no imprimibles, como retroceso, y tabulación empleados para controlar la forma en que la información es transferida desde una computadora a otra o desde una computadora a una impresora. Los 96 códigos restantes se asignan a los signos de pun-

tuación corrientes, a los dígitos del 0 al 9 y a las letras mayúsculas y minúsculas del alfabeto latino.

Los códigos de ASCII extendido, del 128 al 255, se asignan a conjuntos de caracteres que varían según los fabricantes de computadoras y programadores de software. Estos códigos no son intercambiables entre los diferentes programas y computadoras como los caracteres ASCII estándar. Por ejemplo, IBM utiliza un grupo de caracteres ASCII extendido que suele denominarse conjunto de caracteres IBM extendido para sus computadoras personales. Apple Computer utiliza un grupo similar, aunque diferente, de caracteres ASCII extendido para su línea de computadoras Macintosh. Por ello, mientras que el conjunto de caracteres ASCII estándar es universal en el hardware y el software de los microordenadores, los caracteres ASCII extendido pueden interpretarse correctamente sólo si un programa, computadora o impresora han sido diseñados para ello.

2.1.2 Sistema binario.

El sistema binario desempeña un importante papel en la tecnología de los ordenadores. Los primeros 20 números en el sistema en base 2 son 1, 10, 11, 100, 101, 110, 111, 1000, 1001, 1010, 1011, 1100, 1101, 1110, 1111, 10000, 10001, 10010, 10011 y 10100. Cualquier número se puede representar en el sistema binario, como suma de varias potencias de dos.

Las operaciones aritméticas con números en base 2 son muy sencillas. Las reglas básicas son: $1 + 1 = 10$ y $1 \times 1 = 1$. El cero cumple las mismas propiedades que en el sistema decimal: $1 \times 0 = 0$ y $1 + 0 = 1$. La adición, sustracción y multiplicación se realizan de manera similar a las del sistema decimal:

Puesto que sólo se necesitan dos dígitos (o bits), el sistema binario se utiliza en los ordenadores o computadoras. Un número binario cualquiera se puede representar, por ejemplo, con las distintas posiciones de una serie de interruptores. La posición "encendido" corresponde al 1, y "apagado" al 0. Además de interruptores, también se pueden utilizar puntos imantados en una cinta magnética o disco: un punto imantado representa al dígito 1, y la ausencia de un punto imantado es el dígito 0. Los biestables son dispositivos electrónicos con sólo dos posibles valores de voltaje a la salida y que pueden saltar de un estado al otro mediante una señal externa, también se pueden utilizar para representar números binarios. Los circuitos lógicos realizan operaciones con números en base 2. La conversión de números decimales a binarios para hacer cálculos, y de números binarios a decimales para su presentación, se realizan electrónicamente.

2.1.2.1 Medidas de almacenamiento de la información.

Byte: unidad de información que consta de 8 bits; en procesamiento informático y almacenamiento, el equivalente a un único carácter, como puede ser una letra, un número o un signo de puntuación.

- Kilobyte (Kb): Equivale a 1.024 bytes.
- Megabyte (Mb): Un millón de bytes o 1.048.576 bytes.
- Gigabyte (Gb): Equivale a mil millones de bytes.

En informática, cada letra, número o signo de puntuación ocupa un byte (8 bits). Por ejemplo, cuando se dice que un archivo de texto ocupa 5.000 bytes estamos afirmando que éste equivale a 5.000 letras o caracteres. Ya que el byte es una unidad de información muy pequeña, se suelen utilizar sus múltiplos: kilobyte (Kb), megabyte (MB), gigabyte (GB). Como en informática se utilizan potencias de 2 en vez de potencias de 10, se da la circunstancia de que cada uno de estos múltiplos no es 1.000 veces mayor que el anterior, sino 1.024 ($2^{10} = 1.024$). Por lo que $1 \text{ GB} = 1.024 \text{ MB} = 1.048.576 \text{ Kb} = \text{más de } 1.073 \text{ millones de bytes}$.

2.1.3. Dispositivo de almacenamiento. Es todo aparato que se utilice para grabar los datos del computador de forma permanente o temporal. Una unidad de disco, junto con los discos que graba, es un dispositivo de almacenamiento. A veces se dice que un computador tiene dispositivos de almacenamiento primarios (o principales) y secundarios (o auxiliares). Cuando se hace esta distinción, el dispositivo de almacenamiento primario es la memoria de acceso aleatorio (RAM) del computador, un dispositivo de almacenamiento permanente, pero cuyo contenido es temporal. El almacenamiento secundario incluye los dispositivos de almacenamiento permanente, como unidades de disco y de cinta.

La velocidad de un dispositivo se mide por varios parámetros: la velocidad máxima que es capaz de soportar, que suele ser relativa, en un breve espacio de tiempo y en las mejores condiciones; la velocidad media, que puede mantener de forma constante en un cierto período de tiempo y por último, el tiempo medio de acceso que tarda el dispositivo en responder a una petición de información debido a que debe empezar a mover sus piezas, a girar y buscar el dato solicitado. Este tiempo

se mide en milisegundos (ms) y cuanto menor sea ésta cifra, más rápido será el acceso a los datos.

2.1.4 Sistemas de archivo. Todo dispositivo que almacene datos ha de ser formateado antes de poder utilizarlo; es decir, se ha dado la forma para que reconozca cómo ha de almacenar la información. Esta operación la realiza un programa como el FORMAT (formatear) que lo que hace es darle la forma de sectores y pistas para que un sistema operativo concreto reconozca ese espacio y a la vez destruye toda la información que contenga el citado dispositivo.

Cuando se guarda un archivo, se instala un programa, etc., el ordenador almacena la información en el disco duro en pequeñas áreas llamadas clústeres. Cuanto menor sea el tamaño del clúster que se utiliza más eficazmente se almacenará la información en el disco. El tamaño del clúster depende del tamaño de la partición (cada una de las divisiones lógicas de un disco, que se asemejan a discos duros separados) y el tamaño de la partición depende del sistema de archivos que utilice. Generalmente, la mayoría de los equipos utilizan una sola partición.

2.1.4.1 Disco duro. Es un dispositivo compuesto por una o varias láminas rígidas de forma circular, recubiertas de un material que posibilita la grabación magnética de datos. Un disco duro normal gira a una velocidad de 3.600 RPM³ y las cabezas de lectura y escritura se mueven en la superficie del disco sobre una burbuja de aire de una profundidad de 10 a 25 millonésimas de pulgada. El disco duro va sellado para evitar la interferencia de partículas en la mínima distancia que existe entre las cabezas y el disco. Los discos duros proporcionan un acceso más rápido a los datos que los discos flexibles y pueden almacenar mucha más información.

Al ser las láminas rígidas, pueden superponerse unas sobre otras, de modo que una unidad de disco duro puede tener acceso a más de una de ellas. La mayoría de los discos duros tienen de dos a ocho láminas. Actualmente, los tamaños son del orden de Gigabytes, su tiempo medio de acceso es muy bajo (algo menos de 20 milisegundos) y su velocidad de transferencia es tan alta que deben girar a más de 4.000 RPM. Ver figura 4.

³ RPM: Revoluciones por minuto (rpm, RPM o r/min) es una unidad de frecuencia, usada frecuentemente para medir la velocidad angular. En este contexto, una revolución es una vuelta de una rueda, un eje, un disco o cualquier cosa que gire.

Figura 4. Disco duro.



Fuente: <http://www.mrandersonmd.com/2006/07/08/06-10-2008-02:40-pm>

2.1.4.2 Unidad de CD-ROM. La unidad de CD-ROM permite utilizar discos ópticos de una mayor capacidad que los disquetes de 3,5 pulgadas hasta 700 MB. Ésta es su principal ventaja, pues los CD-ROM se han convertido en el estándar para distribuir sistemas operativos, aplicaciones, etc. Ver figura 5.

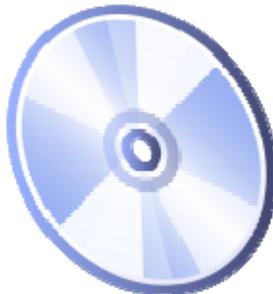
El uso de estas unidades está muy extendido, ya que también permiten leer los discos compactos de audio.

Para introducir un disco, en la mayoría de las unidades hay que pulsar un botón para que salga una especie de bandeja donde se deposita el CD-ROM. Pulsando nuevamente el botón, la bandeja se introduce.

En estas unidades, además, existe una toma para auriculares, y también pueden estar presentes los controles de navegación y de volumen típicos de los equipos de audio para saltar de una pista a otra.

Una característica básica de las unidades de CD-ROM es la velocidad de lectura que normalmente se expresa como un número seguido de una «x» (40x, 52x,...). Este número indica la velocidad de lectura en múltiplos de 128 KB/s. Así, una unidad de 52x lee información de $128 \text{ KB/s} \times 52 = 6,656 \text{ KB/s}$, es decir, a 6,5 MB/s.

Figura 5. CDROM.



Fuente: <http://www.cdrom.com/2006/07/08/06-10-2008-02:51pm>

2.1.4.3 Unidad de CD-RW (Regrabadora). Las unidades de CD-ROM son sólo de lectura. Es decir, pueden leer la información en un disco, pero no pueden escribir datos en él.

Una regrabadora (CD-RW) puede grabar y regrabar discos compactos. Las características básicas de estas unidades son la velocidad de lectura, de grabación y de regrabación. En discos regrabables es normalmente menor que en los discos grabables una sola vez. Las regrabadoras que trabajan a 8X, 16X, 20X, 24X, etc.

Permiten grabar los 650, 700 MB o más tamaño (hasta 900 MB) de un disco compacto en unos pocos minutos. Es habitual observar tres datos de velocidad, según la expresión ax bx cx (**a**: velocidad de lectura; **b**: velocidad de grabación; **c**: velocidad de regrabación).

2.1.4.4 Unidad de DVD-ROM. Las unidades de DVD-ROM son aparentemente iguales que las de CD-ROM, pueden leer tanto discos DVD-ROM como CD-ROM. Se diferencian de las unidades lectoras de CD-ROM en que el soporte empleado tiene hasta 17 GB de capacidad, y en la velocidad de lectura de los datos. La velocidad se expresa con otro número de la «x»: 12x, 16x... Pero ahora la x hace referencia a 1,32 MB/s. Así: 16x = 21,12 MB/s. Ver figura 6.

Figura 6. Unidad de DVD-ROM.



Fuente: <http://www.dvdrom.com/2006/07/08/06-10-2008-03:00-pm>.

Las conexiones de una unidad de DVD-ROM son similares a las de la unidad de CD-ROM: placa base, fuente de alimentación y tarjeta de sonido. La diferencia más destacable es que las unidades lectoras de discos DVD-ROM también pueden disponer de una salida de audio digital. Gracias a esta conexión es posible leer películas en formato DVD y escuchar seis canales de audio separados si se dispone de una buena tarjeta de sonido y un juego de altavoces apropiado (sub-woofer más cinco satélites).

2.1.4.5 Unidad de DVD-RW. Puede leer y grabar imágenes, sonido y datos en discos de varios gigabytes de capacidad, de una capacidad de 650 MB a 9 GB.

2.1.4.6 Memoria USB, pendrive o USB flash drive. Es un pequeño dispositivo de almacenamiento que utiliza memoria flash para guardar la información que puede requerir o no baterías (pilas), en los últimos modelos la batería no es requerida, la batería era utilizada por los primeros modelos. Estas memorias son resistentes a los rasguños externos y al polvo que han afectado a las formas previas de almacenamiento portátil, como los CDs y los disquetes. Ver figura 7.

Figura 7. Memoria USB.



Fuente: <http://www.usb.com/2006/07/08/06-10-2008-03:55> pm.

2.1.4.7. Unidades magneto-ópticas. Estas unidades magneto-ópticas son menos usadas en entornos domésticos que las unidades de CD-ROM, pero tienen varias ventajas:

- Por una parte; admiten discos de gran capacidad: 230 MB, 640 Mb o 1,3 GB.
- Además; son discos reescribibles, por lo que es interesante emplearlos, por ejemplo, para realizar copias de seguridad.

2.1.4.8 Otros dispositivos de almacenamiento. Otros dispositivos de almacenamiento son las memorias flash o los dispositivos de almacenamiento magnético de gran capacidad.

- **La memoria flash.** Es un tipo de memoria que se comercializa para el uso de aparatos portátiles, como cámaras digitales o agendas electrónicas. El aparato correspondiente o bien un lector de tarjetas, se conecta a la computadora a través del puerto USB o Firewire⁴. Ver Figura 8.

⁴ La tecnología Firewire originalmente fue desarrollada por Apple en la primera mitad de la década de los 90's. El puerto está conformado por 6 cables y maneja dos tipos de estándares en conexiones: la de 4 pines y la de 6. Sony utiliza únicamente 4 cables en sus conexiones para transferir

Figura 8. Memoria Flash.



Fuente: <http://www.memoria flash.com/2006/07/08/06-10-2008> – 04:57 pm.

- **Los discos duros o memorias portátiles.** Son memorias externas que se conectan directamente al puerto USB.
- **Discos y cintas magnéticas de gran capacidad.** Son unidades especiales que se utilizan para realizar copias de seguridad o respaldo en empresas y centros de investigación. Su capacidad de almacenamiento puede ser de cientos de gigabytes.

2.1.4.9 Eeprom. Son las siglas de Electrically-Erasable Programmable Read-Only Memory (ROM programable y borrable eléctricamente). En español se la suele denominar "E²PROM" y en inglés "E-Squared-PROM". Es un tipo de memoria ROM que puede ser programado, borrado y reprogramado eléctricamente, a diferencia de la EPROM que ha de borrarse mediante rayos ultravioleta.

Aunque una EEPROM puede ser leída un número ilimitado de veces, sólo puede ser borrada y reprogramada entre 100.000 y un millón de veces.

audio y video. Muchas computadoras de ahora ya tienen integrado el puerto firewire/I-link/IEEE 1394, ya sea en 4 pines o en 6.

Estos dispositivos suelen comunicarse mediante protocolos como I²C, SPI y Microwire. En otras ocasiones, se integra dentro de chips como microcontroladores y DSPs para lograr una mayor rapidez.

La memoria flash es una forma avanzada de EEPROM creada por el Dr. Fujio Masuoka mientras trabajaba para Toshiba en 1984 y fue presentada en la Reunión de Aparatos Electrónicos de la IEEE de 1984. Intel vio el potencial de la invención y en 1988 lanzó el primer chip comercial de tipo NOR. Ver figura 9.

Figura 9. Memoria EEPROM.



Fuente: <http://www.memoria eeprom.com/2006/07/08/>
06 –10-2008 – 05:58pm

2.1.4.10 Memorias SD. Es normal hoy que la cámara digital use memoria del tipo SD (usada por varios fabricantes, no sólo de máquinas fotográficas sino también de filmadoras, mp3 players, PDAs, GPS, etc). Por esto es bueno estar informado sobre las varias opciones que hay en memorias SD. La tarjeta de memoria SD (llamado de Secure Digital) tiene muy buenas referencias de todos los que la utilizan a diario, y por esto se ha vuelto un standard muy utilizado, juntamente con sus "hermanos" mini-sd y micro-sd. Ver figura 10.

Figura 10. Memoria SD.



Fuente: <http://www.memoria SD.com/2006/07/08/05-10-2008-02:58Pm>.

2.1.4.11 Algunas características interesantes.

FORMATO: El formato es asimétrico, por lo tanto, no hay riesgo de ser inserta por error en su equipo, además de tener un tamaño bien pequeño:

Memoria SD tradicional: 32 mm x 24 mm x 2.1 mm.

Memoria Mini SD: 20 mm x 21,5 mm x 1,4 mm.

Memoria Micro SD o Transflash: 11 mm x 15 mm x 1 mm.

CAPACIDAD: Se encuentran con diferentes capacidades: 8, 16, 32, 64, 128, 256, 512 MB y 1, 2, 4, 6, 8 GB. Es importante recordar que no todos los equipos poseen soporte para tarjetas con 2GB o más, debido a la diferencia en el formateo (algunas veces el problema es resuelto por actualización de firmware de dicho equipo).

El hecho de que las máquinas fotográficas ofrezcan cada vez más megapíxeles sin aumentar el precio, genera la necesidad de más memoria para que quepan las fotografías que, a más alta definición ocupan mayor espacio.

COMPATIBILIDAD: Utilizado por incontables fabricantes de maquinas fotogrficas digitales como Casio, Canon, Nikon, Pentax, Kodak, Panasonic, Konica/Minolta.

Es compatible con tarjetas micro SD, tambin llamado de Transflash (utilizado en algunos mviles, mp3 players, memory keys, etc.) y mini SD (utilizado en algunos PDAs, maquinas digitales, etc). En realidad existe un adaptador que transforma estas otras tarjetas en tarjetas SD.

PROTECCIN: Posee un cdigo que permite bloquear la sd para. Sin embargo, a diferencia de los diskettes, este bloqueo es hecho por el firmware del aparato donde la llave est activada.

Los contactos de la tarjeta son colocados en una depresin de la tarjeta que evita el contacto con la grasa del dedo, y otros materiales.

VELOCIDAD: Puede poseer diferentes velocidades de transferencia (lo que vara tambin en el precio), conforme a la versin utilizada. La velocidad es medida en mltiplos de 150K (igual al CD-ROM). Lo ms usual es encontrar tarjetas/equipamientos que utilizan hasta velocidad de 16x (2,4 Mbyte/seg), y algunos ms rpidos de 66x (10 Mbyte/seg), aunque tambin pueden encontrarse tarjetas ultra-rpidas de 133x o hasta 150x. En realidad, muchas veces slo se aprovecharn las tarjetas de velocidades mayores al llevar a cabo la transferencia de datos de la tarjeta a la computadora, utilizando lectores de tarjeta con USB 2.0 por ejemplo, ya que difcilmente cmaras fotogrficas digitales de uso comn tendrn este soporte (funcionarn con la tarjeta, pero ms lento), y muchas ni especifican este tipo de informacin. Sin embargo, si usted est pensando en utilizar equipamientos ms avanzados (cmaras profesionales, semi-profesionales, SLR, etc.), tendr con certeza una ganancia grande en la utilizacin de tarjetas de memoria ms rpidas.

2.1.4.12 OTRAS FUNCIONES. Debido a su gran popularizacin, la SD (card Association), est definiendo nuevas funciones, nuevas aplicaciones, etc y algunas de ellas ya pueden ser encontradas en el mercado, como: SD Card con Copyright (est siendo utilizado para "distribucin" de pelculas, o sea, si compra un SD Card, que tiene una pelcula en un formato DVD, para poder visualizarla en equipos como PS3, dispositivos porttiles, etc). Esto ya es una realidad si bien este formato vino para quedarse, existen diferencias significativas de caractersticas, de capacidad y precio. Por eso es necesario, antes de adquirirlo, conocer un poco ms para poder seleccionar mejor el modelo que traer el resultado esperado en estos equipos.

2.1.5 Nuevas tecnologías en sistemas de almacenamiento.

2.1.5.1. Soluciones a gran escala. Los sistemas de almacenamiento se han convertido en un aspecto singular y complejo de la informática que puede enfocarse desde distintos puntos de vista. Se podría definir como el hardware donde se guardan datos.

Se pone un ejemplo un pequeño servidor en una oficina de diez o menos usuarios. El sistema de almacenamiento serían los discos duros integrados en el servidor, donde se guarda la información. En entornos de gran empresa, el sistema de almacenamiento puede ser un gran armario SAN lleno de discos duros donde el espacio ha sido muy fino para proporcionar rendimiento y redundancia.

2.1.5.2 Microcontrolador. Es un circuito integrado o chip que incluye en su interior las tres unidades funcionales de un computador: CPU, Memoria y Unidades de E/S, es decir, se trata de un computador completo en un solo circuito integrado.

Aunque sus prestaciones son limitadas, además de dicha integración, su característica principal es su alto nivel de especialización. Aunque los hay del tamaño de un sello de correos, lo normal es que sean incluso más pequeños, ya que, lógicamente, forman parte del dispositivo que controlan.

Es un microprocesador optimizado para ser utilizado para controlar equipos electrónicos. Los microcontroladores representan la inmensa mayoría de los chips de computadoras vendidos, sobre un 50% son controladores simples y el restante corresponde a dispositivos más especializados. Mientras se puede tener uno o dos microprocesadores de propósito general en casa, se tiene probablemente distribuido entre los electrodomésticos del hogar una o dos docenas de microcontroladores. Pueden encontrarse en casi cualquier dispositivo eléctrico como automóviles, lavadoras, hornos microondas, teléfonos, etc.

Un microcontrolador difiere de una CPU normal, debido a que es más fácil convertirla en un computador en funcionamiento, con un mínimo de chips externos de apoyo. La idea es que el chip se coloque en el dispositivo, enganchado a la fuente de energía y de información que necesite. Un microprocesador tradicional no le permitirá hacer esto, ya que espera que todas estas tareas sean manejadas por otros chips, Por ejemplo, un microcontrolador típico tendrá un generador de reloj

integrado y una pequeña cantidad de memoria RAM⁵ y ROM/EPROM/EEPROM, significando que para hacerlo funcionar, todo lo que se necesita son unos pocos programas de control y un cristal de sincronización. Los microcontroladores disponen generalmente también de una gran variedad de dispositivos de entrada/salida, como convertidores de analógico a digital, temporizadores y buses de interfaz serie especializados, como I²C y CAN⁶. Frecuentemente, estos dispositivos integrados pueden ser controlados por instrucciones de procesadores especializados. Los modernos microcontroladores frecuentemente incluyen un lenguaje de programación integrado, como el BASIC que se utiliza bastante con este propósito.

Los microcontroladores negocian la velocidad y la flexibilidad para facilitar su uso.

Debido a que se utiliza bastante espacio en el chip para incluir funcionalidad, como los dispositivos de entrada/salida la memoria que incluye el microcontrolador, se ha de prescindir de cualquier otra circuitería. Ver figura 11.

Figura 11. Microcontrolador.



Fuente: <http://www.microcontrolador.com/2006/07/08/05-10-2008-03:05PM>.

⁵ RAM: es una memoria de acceso aleatorio, o memoria de acceso directo (en inglés: Random Access Memory, cuyo acrónimo es RAM), o más conocida como memoria RAM, se compone de uno o más chips y se utiliza como memoria de trabajo para programas y datos. Es un tipo de memoria temporal que pierde sus datos cuando se queda sin energía (por ejemplo, al apagar la computadora), por lo cual es una memoria volátil.

CAN: Controller Area Network (CAN) es un Standard de bus serie Para permitir la adaptación con redes de conexionado multiplexado desarrollado conjuntamente por Bosch e Intel para el cableado de dispositivos en automóviles. hoy en día sus aplicaciones van desde las redes de alta velocidad a cableado multiplexado de bajo costo.

2.1.5.3 Sensor. Es un dispositivo que detecta o sensa manifestaciones de cualidades o fenómenos físicos, como la energía, velocidad, aceleración, tamaño, cantidad, etc. Se puede decir también que es un dispositivo que aprovecha una de sus propiedades con el fin de adaptar la señal que mide para que la pueda interpretar otro elemento. Como por ejemplo; el termómetro de mercurio, que aprovecha la propiedad que posee el mercurio de dilatarse o contraerse por la acción de la temperatura.

Muchos de los sensores son eléctricos o electrónicos, aunque existen otros tipos. Un sensor es un tipo de transductor que transforma la magnitud que se quiere medir, en otra, que facilita su medida. Pueden ser de indicación directa ej: un termómetro de mercurio o pueden estar conectados a un indicador (posiblemente a través de un convertidor analógico a digital, un computador) de modo que los valores sensados puedan ser leídos por un humano.

2.2 MARCO LEGAL O NORMATIVO

En Colombia en cuanto a la aplicación de normas referentes a los instrumentos de medicina son muy pocas, se encuentra el código de ética médica LEY 23 de 1981, en su capítulo II, en cuanto al procedimiento médico paciente, el médico podrá dar aplicación a todos los aportes científicos que proporcionen una mejor calidad de vida al paciente.

Teniendo en cuenta las leyes internacionales se puede citar: ley # 41 de 1985, Gaceta Oficial de la República de Cuba, ley de la salud pública, Art.106 que mediante el Ministerio de salud pública garantiza a través de la red de servicios técnicos de electromedicina el montaje, mantenimiento y reparación de los equipos médicos, estomatológicos y de laboratorio instalados en sus unidades.

Ley 21/2000, del Parlamento de Cataluña, sobre los derechos de información concernientes a la salud y la autonomía del paciente y la documentación clínica.

El derecho de dar o no dar su consentimiento antes de cada actuación sanitaria, deriva del derecho básico a que se respete la autonomía personal en la situación de enfermedad. Por ello, a lo largo de la segunda mitad del siglo XX se ha ido configurando como obligación legal en la mayoría de países. En España se recogió con rango normativo en la Ley 14/1986, de 25 de abril, General de Sanidad (LGS). A partir de ella y lentamente, se va tomando conciencia de este derecho con el

desarrollo de lo que se llama consentimiento informado en bioética y en la práctica clínica.

Ley 21/2000, del Parlamento de Cataluña, sobre los derechos de información concernientes a la salud y la autonomía del paciente y la documentación clínica.

Art. 2 Formulación y alcance del derecho a la información.

1. Las personas tienen derecho a conocer toda la información obtenida con respecto a su salud. No obstante, tiene que respetarse la voluntad de una persona de no ser informada.

La información tiene que formar parte de todas las actuaciones asistenciales, tiene que darse de forma comprensible, ser verídica y adecuada a las necesidades y los requerimientos del paciente, con el fin de ayudarlo a tomar decisiones de forma autónoma.

2. Corresponde al médico responsable del paciente garantizar el cumplimiento del derecho a la información. También tienen que asumir responsabilidades en el proceso de información aquellos profesionales asistenciales que le atiendan o le apliquen una técnica o un procedimiento concreto.

Art. 3 Del titular del derecho a la información.

1. El titular del derecho a la información es el paciente. Se debe informar a las personas vinculadas al paciente en la medida en que éste lo permita expresamente o tácitamente.

1. En caso de incapacidad del paciente, la información tiene que darse también quien ostente su representación.

4. Cuando el paciente, a criterio del médico responsable de la asistencia, no sea competente para entender la información como consecuencia de encontrarse en un estado físico o psíquico que no le permita hacerse cargo de su situación, se tendrá que informar también a los familiares o personas vinculadas.

3. METODOLOGÍA

3.1 ENFOQUE DE LA INVESTIGACIÓN

El estudio que se realiza tendrá un enfoque empírico-analítico; ya que durante el desarrollo del mismo se implementará un nuevo dispositivo orientado a la interpretación y transformación del mundo material, mediante procesos matemáticos y simulaciones computacionales.

Al desarrollar este dispositivo se obtendrá información más confiable en relación con algunos signos vitales en el hombre.

3.2 LÍNEA DE INVESTIGACIÓN DE USB / SUB-LÍNEA DE FACULTAD / CAMPO TEMÁTICO DEL PROBLEMA.

El campo de investigación que involucra este proyecto es tecnologías actuales y sociedad, que como se mencionó anteriormente, lo que se pretende es tomar datos mediante algunas etapas y luego archivarlos digitalmente.

Por otra parte la sublínea de investigación de electrónica conveniente para este proyecto es instrumentación y control de procesos, cuándo se determinaran y manipularan sensores, para hallar las variables deseadas.

Y en el campo de investigación se tiene la microelectrónica se trabajará con microcontroladores.

3.3 TÉCNICAS DE RECOLECCIÓN DE INFORMACIÓN

Se obtuvo información para el desarrollo del dispositivo a partir de entrevistas al personal médico de la clínica Juan N Corpas y Saludcoop; se hablo de un punto clave en la entrevista y se vio la necesidad de diseñar y construir un dispositivo que interactué en actividades de enfermería en un centro clínico u hospitalario.

Se hará una observación importante: en un centro medico se tienen 20 pacientes a los cuales se les debe que hacer la toma de sus signos vitales cada hora, por lo

que el dispositivo ejecutara esta rutina de una forma precisa y rápida, dando espacio al personal médico para cumplir otras actividades de mayor importancia.

3.4 HIPÓTESIS

Basado en que el personal de un hospital se esfuerza demasiado con rutinas que a veces se tienen que suplir por otras de mayor importancia (dicho en las entrevistas realizadas a enfermeros y médicos en centros hospitalarios (clínica Juan N. Corpas) se propone el diseño, construcción y análisis de un dispositivo portátil que mide variables vitales en pacientes, construido con dispositivos electrónicos análogos y digitales, a fin de obtener un artefacto de bajo costo y fácil manejo, que pronostique y archive de forma rápida ahorrando tiempo en actividad médica.

3.5 VARIABLES

3.5.1. Variables independientes. Se tendrán variables de temperatura y frecuencia cardíaca para con ello mantener una historia clínica archivada sin importar el transcurso de tiempo.

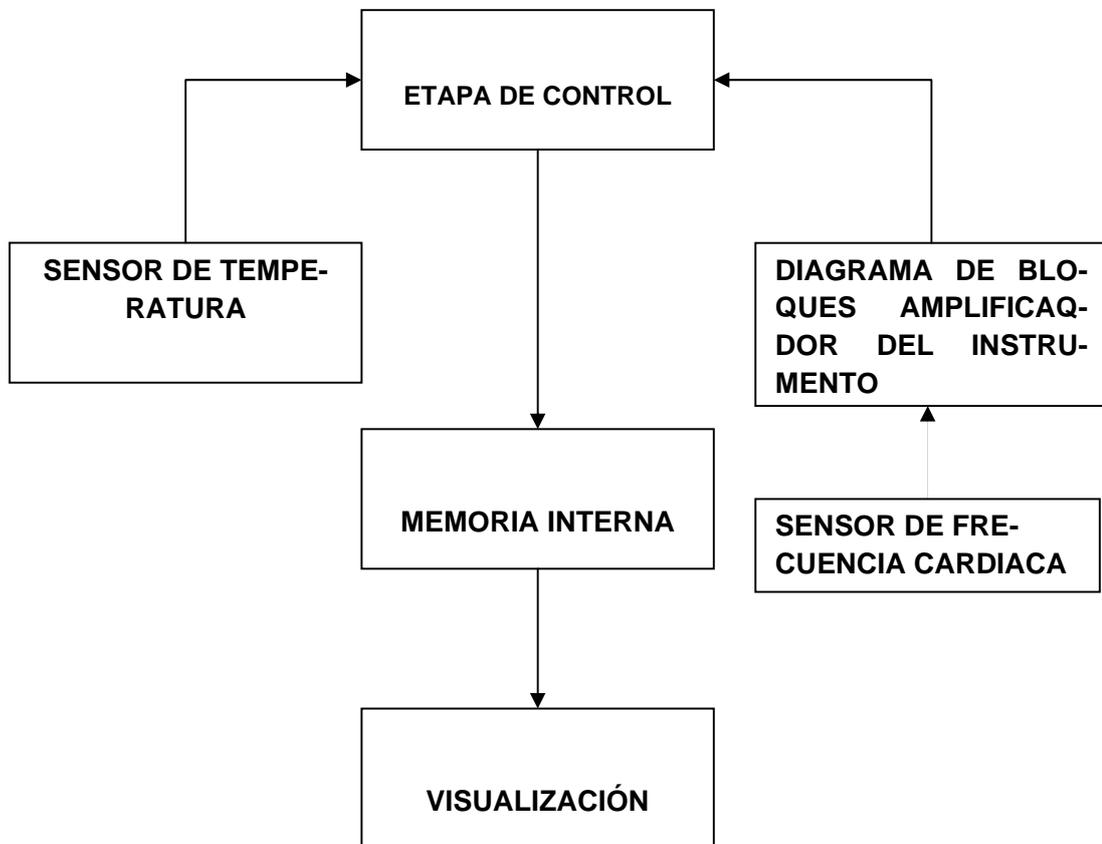
Se tendrán en cuenta variables que interactúan en la obtención de los datos esperados como es el sistema cardiaco. Estas variables se obtendrán mediante sistemas sensóricos.

3.5.2. Variables dependientes. Se tiene como variables dependientes: el sensor de frecuencia cardiaca y temperatura, el microcontrolador a programar en este caso un PIC 18f452, un buzzer como alarma y la pantalla LCD de 4x20 que interactúa con la decodificación de los datos obtenidos.

4. DESARROLLO INGENIERIL

En el diagrama descriptivo de bloques funcionales se resumirá las características de cada etapa que interactúa en el dispositivo, también se explicaran las características de los dispositivos que se usaron para el diseño y su respectivo funcionamiento.

Figura 12. Diagrama descriptivo de bloques funcionales del dispositivo.



4.1 ETAPA DE CONTROL

Se tomaron tres microcontroladores los cuales son algunos de los más económicos y conseguidos en el mercado, además se estudio cada uno de ellos y se concluyo el uso de uno de estos a partir de las características más adecuadas para el diseño del dispositivo historial digital de datos médicos.

Figura 13. Características de algunos microcontroladores.

MICROCONTROLADOR	RAM	ROM/FLASH	I/O	ADC líneas de entrada	EEprom/bytes	SET DE INSTRUCCIONES	#PINES
PIC 16877A	256 bytes	8kbytes	33	8 de 10 bits	256	35	40
PIC 18F452	256 bytes	32kbytes	33	8 de 10 bits	256	35	40
MOTOROLA MC68HC908GP32	256 bytes	32kbytes	33	10 de 10 bits	512		40

Se trabajo con el PIC18f452 por ser un microcontrolador de alta gama que combina un extenso conjunto de periféricos con un potente núcleo microcontrolador.

A diferencia del 16f877 el 18f452 cuenta con 4 veces más capacidad en memoria flash, esto facilita una mayor cobertura de opciones en el dispositivo, como el archivo de datos medidos; entre mas de capacidad memoria flash, mayor cantidad de datos a archivar.

A pesar de las ventajas del microcontrolador Motorola MC68HC908GP32 se prefirió trabajar con el PIC 18f452 puesto que este maneja un set de instrucciones más factibles para el diseñador, entorno a la experiencia que el maneja en el lenguaje de programación assembler.

4.1.1 Sensor de temperatura. El sensor de temperatura esta compuesto de un termopar con características y desempeño apropiados para la lectura de tempera-

tura corporal; su función es traducir los niveles de temperatura corporal en niveles de voltaje interpretables por un conversor análogo digital.

No se utilizó el LM 35 ya que a pesar de su eficiencia en usos electrónicos en este caso su aplicación es muy complicada debido a que por su estructura física no alcanza a leer la variable corporal, para poder leerla se debe poner un conductor de señal directamente al sensor o lamina que permite al LM35 sensor.

El sensor que se utilizó es una resistencia variable con la temperatura, un NTC. Para medidas de temperatura del cuerpo. Para este dispositivo se utilizó sensores NTC de 10kΩ a temperatura ambiente. Según su fabricación, la resistencia irá disminuyendo al aumentar la temperatura. Se debe trabajar en la zona de temperaturas a medir para poder diseñar apropiadamente los acondicionadores. Este bloque se diseñó para configurar el conversor análogo digital y su proceso con la señal de temperatura. Ver sensores en anexos.

Figura 14. Características de algunos sensores.

SENSOR DE TEMPERATURA	CARACTERISTICAS DEL SENSOR
LM 35	<p>El LM35 es un sensor de temperatura con una precisión calibrada de 1°C y un rango que abarca desde -55° a + 150°C. La salida es lineal y equivale a 10 mV/°C.</p> <ul style="list-style-type: none"> • Precisión de : ~1,5°C (peor caso) • No linealidad : ~0,5°C (peor caso) <p>Su tensión de salida es linealmente proporcional con la temperatura en la escala Celsius (centígrada). Posee una precisión aceptable para la aplicación requerida, no necesita calibración externa y es de bajo costo</p>
TERMISTOR	<p>Los termistores, o resistores térmicos, son dispositivos semiconductores que se comportan como resistencias con un coeficiente de temperatura de resistencia alto y, generalmente negativo. En algunos casos, la resistencia de un termistor a temperatura ambiente puede disminuir hasta un 6% por cada 1°C que se eleve la temperatura. Dada esta alta sensibilidad al cambio de temperatura hacen al termistor muy conveniente para mediciones, control y compensar con precisión la temperatura.</p>

4.1.2 Sensor de frecuencia cardiaca. Esta diseñado para procesar la señal de la frecuencia cardíaca través del conversor análogo digital del microcontrolador por medio de los transductores en este caso son los electrodos, La señal del amplificador instrumental es la señal que se Toma como referencia en el monitoreo Cardíaco. Los puntos a medir en el torso del Paciente o entre los brazos y piernas y la derivación de referencia del cuerpo del paciente, la parte del programa en el diseño de digitalización de los pulsos cardiacos se encuentra en la parte de digitalización de la frecuencia cardiaca anexos. Ver figura12. Ver sensores en anexos.

El potencial registrado por el electrocardiógrafo tiene una amplitud aproximada de 1mV y se obtiene aplicando electrodos de registro de biopotenciales para las derivaciones precordiales se utilizan electrodos adhesivos y electrodos de succión. El espectro en frecuencias de la señal electrocardiográfica normalmente no tiene componentes arriba de los 60Hz en pacientes normales, por lo que se considera adecuado un ancho de banda de trabajo entre 0.05 y 150Hz para electrocardiógrafos.

4.1.3 Memoria Interna. Luego de que las señales medidas (señal de frecuencia cardiaca, señal de temperatura corporal) han sido digitalizadas en el microcontrolador, este las procesa dentro de la memoria interna (memoria EEPROM de 256 bytes de almacenamiento de información) para así tener acceso a mediciones tomadas anteriormente.

4.1.4 Visualización. En este bloque se observa las mediciones guardadas en el dispositivo e incluso se observa la medición en proceso, En este bloque se describe el proceso de diseño digital para obtener la visualización por la pantalla de cristal líquido LCD de 4x20

4.2 DISEÑO DEL MONITOR DE FRECUENCIA CARDIACA

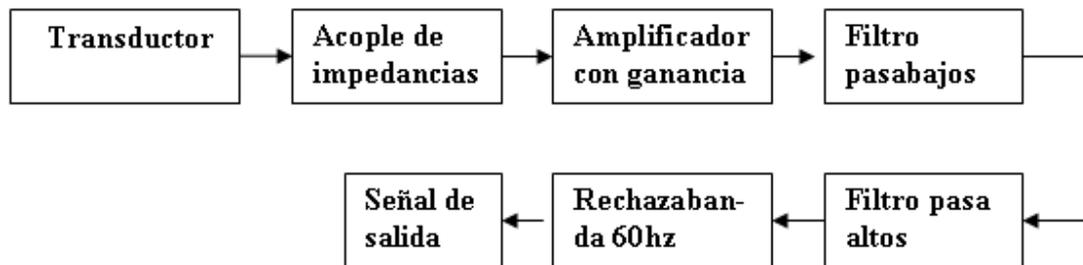
Los requerimientos de implementación a partir de un diseño, son los siguientes:

- La señal de ECG tiene componentes relevantes solo entre 0.05 Hz y 150 Hz
- Los valores de la señal en la piel oscilan en pocos milivoltios (entre unos 0.5 y 10 mv como máximo).
- Ganancia aumentada 1000 veces (30dB).

Otras consideraciones importantes: CMRR lo más alto posible.

Resistencia de entrada de aproximadamente $2\text{ M}\Omega$ o superior, para obtener un acople de impedancias y no atenuar la señal.

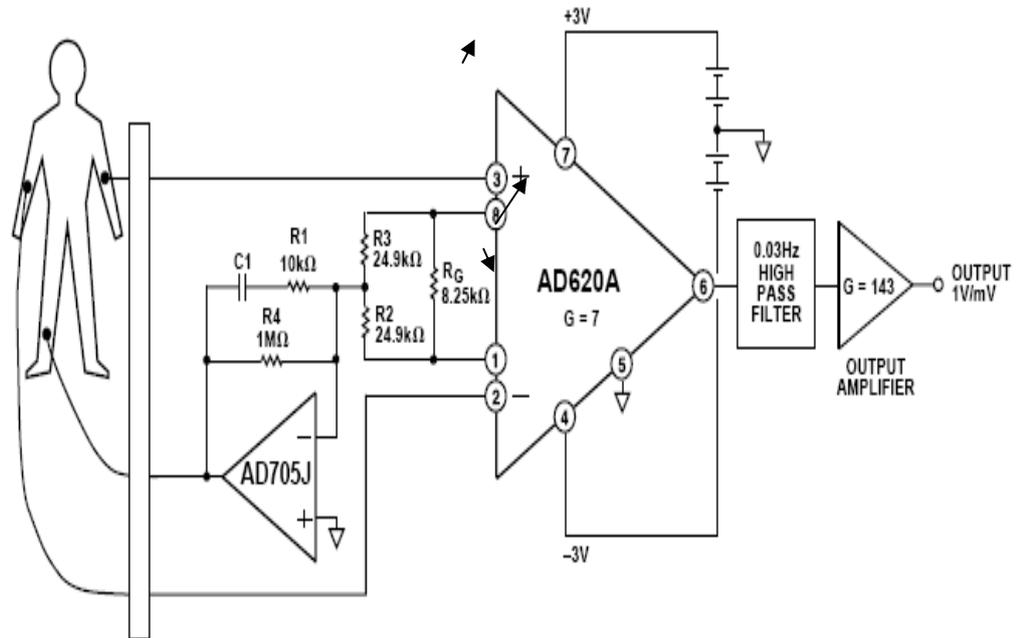
Figura 15. Diagrama de bloques del amplificador del instrumento



4.2.1 Transductor. Para la adquisición de datos, se ha procedido de la siguiente manera: se obtiene la señal electrocardiográfica del usuario a través de los electrodos y estos a su vez se encuentran conectados al circuito a través de medición de un cable apantallado que permite la eliminación de ruidos.

Los electrodos que se utilizarán serán de tipo superficial por su facilidad de manejo y economía. La derivación que se utilizará será: un electrodo que ira a la altura del corazón (encima), otro electrodo va en la parte derecha a la altura intercostal y un último electrodo que servirá como referencia y va a la altura de la cintura en la parte izquierda. Ver figura 16.

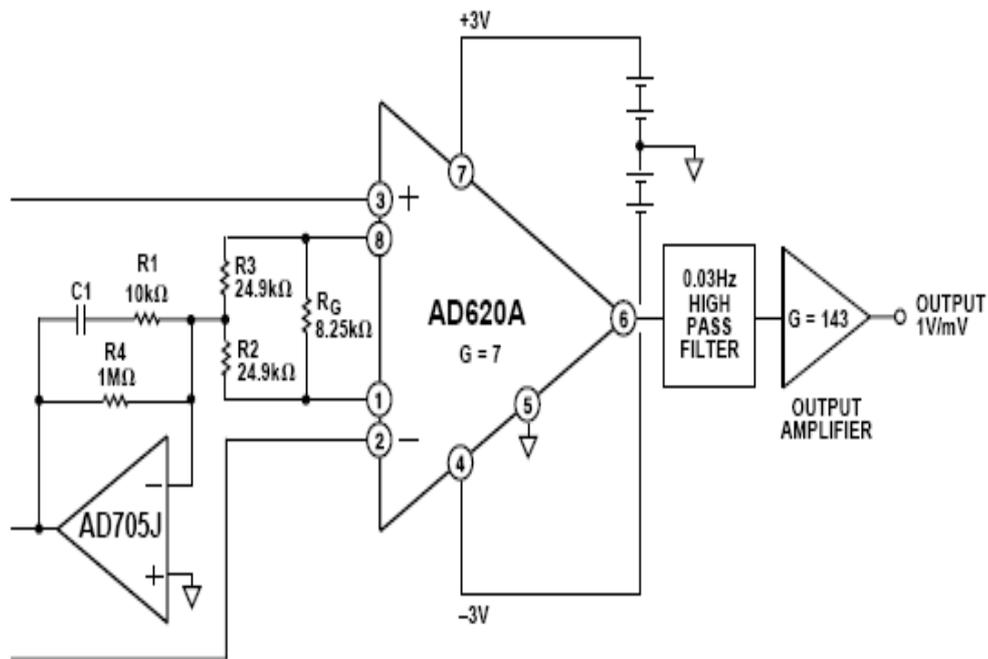
Figura 16. Diagrama de ubicación transductor



Fuente: www.transductor.imagenes.com - 15-07-08 3:30 PM

4.2.2 Acople de impedancias y amplificador con ganancia. Esta parte se hará con un amplificador de instrumentación, ya que brinda una impedancia de entrada infinita, produciéndose el efecto de acople de impedancias y un amplificador diferencial, el cual amplifica la diferencia de la señal proveniente de los electrodos 1 y 2. El amplificador de instrumentación se compone de tres amplificadores operacionales y tiene la siguiente estructura. Ver figura 17.

Figura 17. Amplificador del instrumento



Fuente: proteus 7 Professional - 01-03-08 4:00 pm

La figura anterior es el esquema del amplificador del instrumento.

Como las amplitudes de la señal eléctrica del corazón van desde 1 hasta 5 milivoltios, la ganancia del amplificador debe ser aumentada 1000 veces (30dB) como se había especificado en los requerimientos.

4.2.3 Amplificador del instrumento.

La parte matemática del diseño, esta consignada en el siguiente proceso:

$$V_0 = -V_a \left(\frac{R_3}{R_2} \right) + V_b \left(\frac{R_2 + R_3}{R_2' + R_3'} \right) \frac{R_3'}{R_2} = \frac{R_3'}{R_2'} (V_b - V_a)$$

$$V_a = V_1 \left(1 + \frac{R_1}{R_G} \right) - V_2 \left(\frac{R_1}{R_G} \right)$$

$$V_b = V_1 \left(1 + \frac{R_1'}{R_G} \right) - V_2 \left(\frac{R_1'}{R_G} \right)$$

$$G_d = \frac{R_3}{R_2} \left(1 + 2 \frac{R_1}{R_G} \right) = G_{d1} * G_{d2}$$

V_0 = salida del amplificador.

V_a y V_b son las entradas al diferenciador.

G_d = es la ganancia del amplificador.

Se necesita que G_d sea de $10 \log 1000 = 30$ dB para conseguir esto se asumirá:

$R_g = 8.25 \text{ k}\Omega$.

$R_3 = 24.9 \text{ k}\Omega$.

$R_2 = 24.9 \text{ k}\Omega$.

Falta determinar el valor de R_1 .

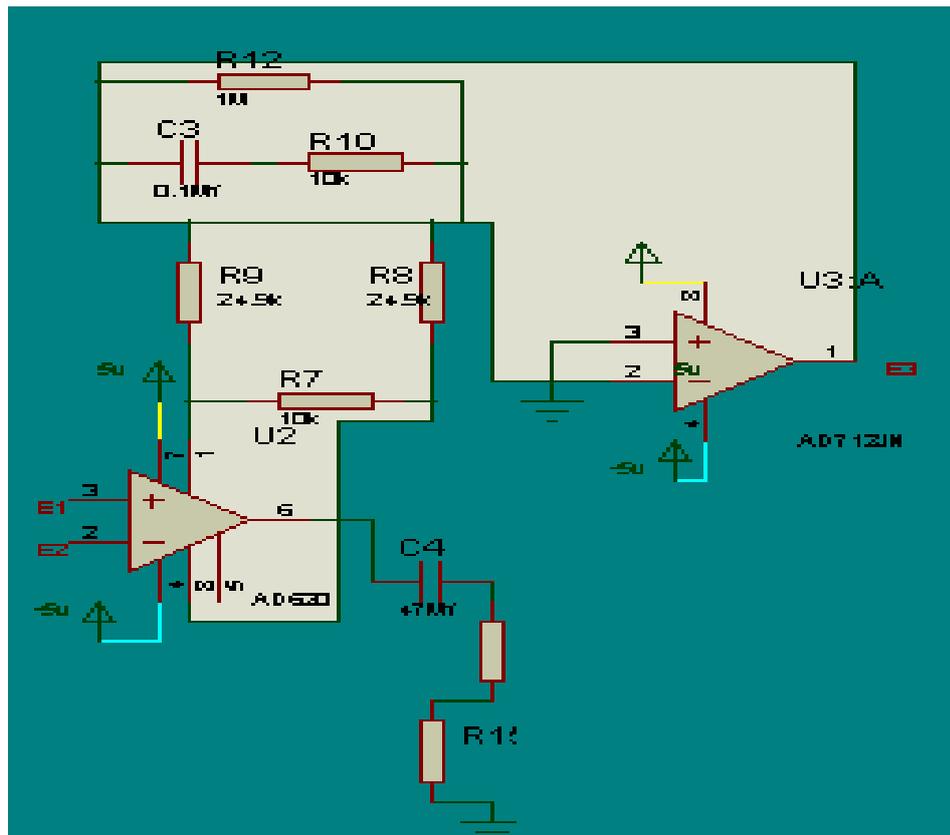
Entonces:

$$R1 = \left[\left(Gd * \frac{R2}{R3} - 1 \right) * RG \right] \div 2$$

R1 = 10KΩ.

Ahora se proseguirá para verificar si hubo o no errores en el cálculo de la ganancia, mediante el proceso de simulación. Ver figura 18.

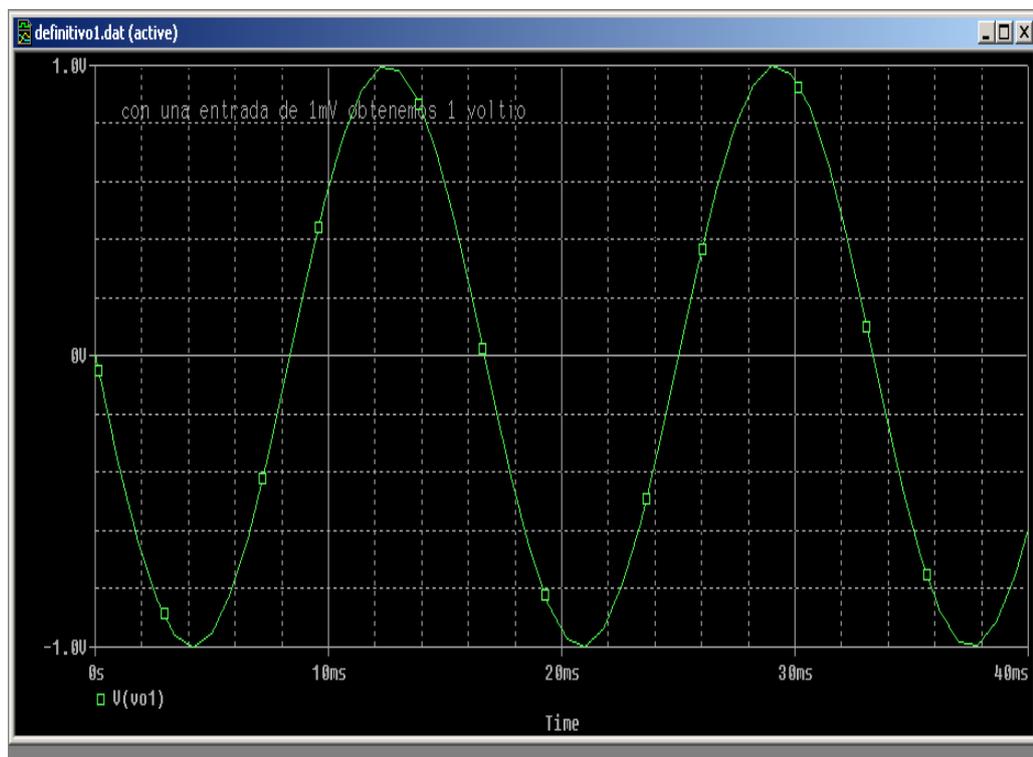
Figura 18. Esquema del amplificador del instrumento.



Fuente: proteus 7 Professional - 02-03-08 2:20 P.m.

4.2.4 Forma de onda a la salida del amplificador de instrumentación. Como se esperaba el amplificador mostró en la simulación que tiene una ganancia de $10 \log 1000 = 30 \text{ dB}$, se aplicó un milivoltio a la entrada y se obtuvo a la salida 1 voltio. Algo que faltó hablar es la alimentación del circuito que fue de 12 y -12 voltios de esto se hablara más adelante cuando se habla del filtro rechaza banda de 60 Hz. Ver figura 19.

Figura 19. Forma de onda a la salida del amplificador de instrumentación

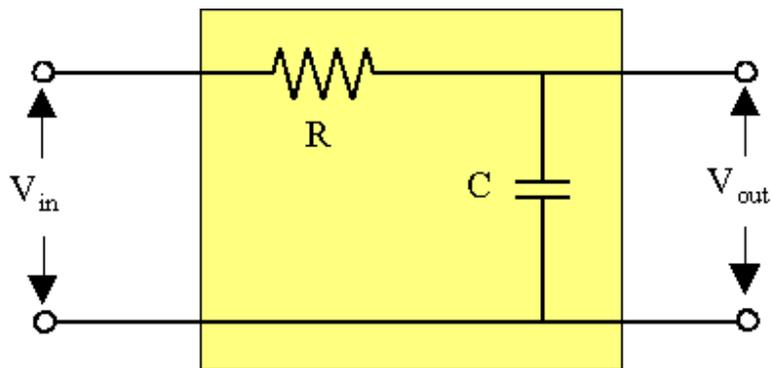


Fuente: proteus 7 Professional - 02-03-08 3:20 P.m.

4.2.5 Filtros pasa bajo y pasa alto. Una de las partes más importantes de un acondicionador de señales es el filtrado, el cual determinara el ancho de banda del circuito. Como se había mencionado anteriormente la señal de ECG tiene componentes relevantes solo entre 0.05 Hz y 150 Hz, por tanto, el circuito solo debe dejar pasar las señales que se encuentren en este rango.

Se utilizará un filtro pasa bajo sencillo el cual consta de una resistencia y un condensador y tiene la siguiente configuración. Ver figura 20.

Figura 20. Gráfica del filtro pasa bajo.



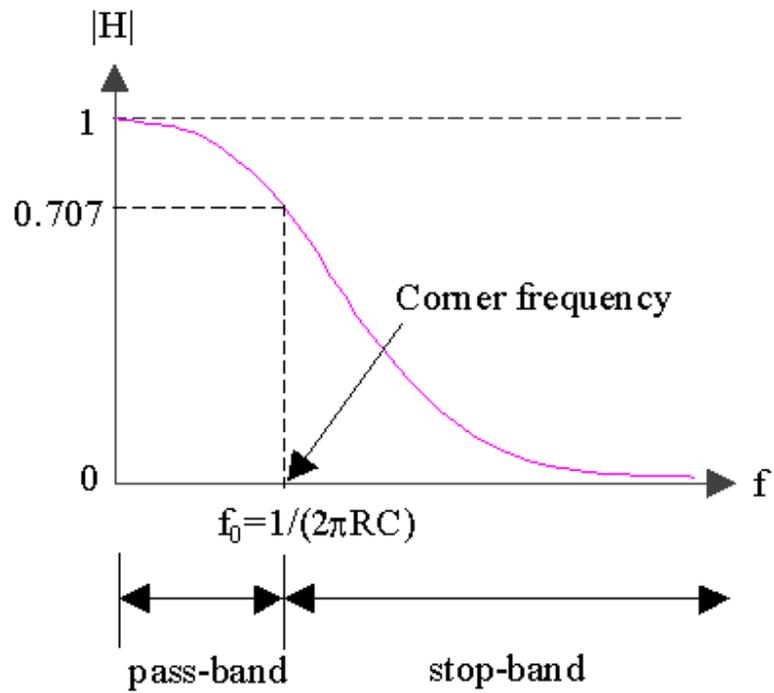
Fuente: electrónica workbench

4.2.6 Filtro pasa bajo. La frecuencia de corte es 150 Hz, se asume un condensador de 1 μ f y de la ecuación se despeja R.

$$R = \frac{1}{150 * 2 * \pi * 1\mu f} = 1061,03\Omega$$

$$F0 = \frac{1}{2\pi RC}$$

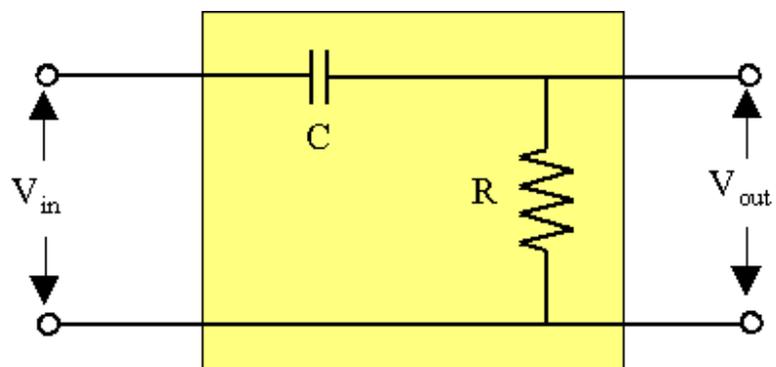
Figura 21. Punto de operación del filtro pasa bajo.



Fuente: electrónica workbench - 09-03-08 4:40 P.m.

4.2.7 Filtro pasa alto. Se utilizará un filtro pasa alto sencillo el cual consta de una resistencia y un condensador y tiene la siguiente configuración. Ver figura 22.

Figura 22. Gráfica del filtro pasa alto.



Fuente: proteus 7 Professional - 09-03-08 6:40 PM.

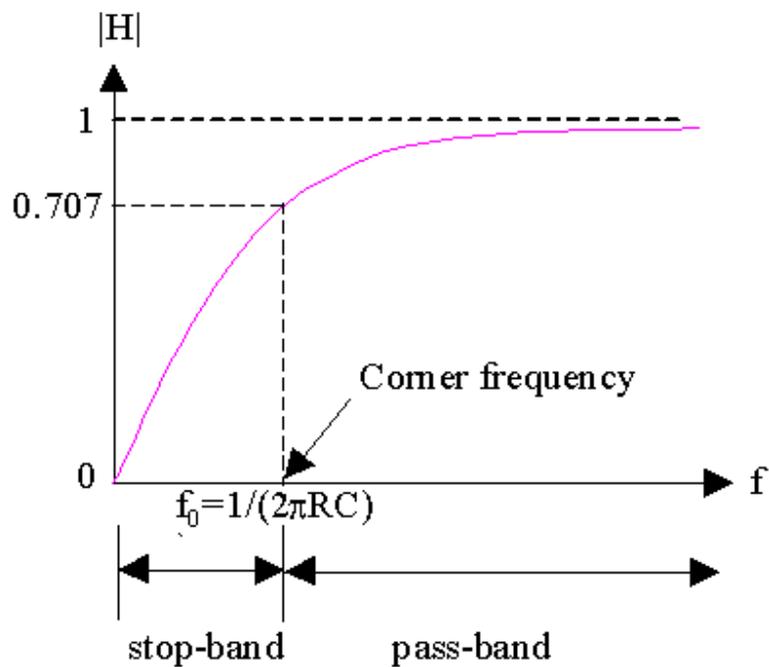
4.2.7 Ecuación Filtro pasa alto.

$$F0 = \frac{1}{2\pi RC}$$

La frecuencia de corte es 0.05 Hz, se asume un condensador de 1 μ f y de la fórmula se despeja R. ver figura 23.

$$R = \frac{1}{0.05 * 2 * \pi * 1\mu f} = 3.18\Omega$$

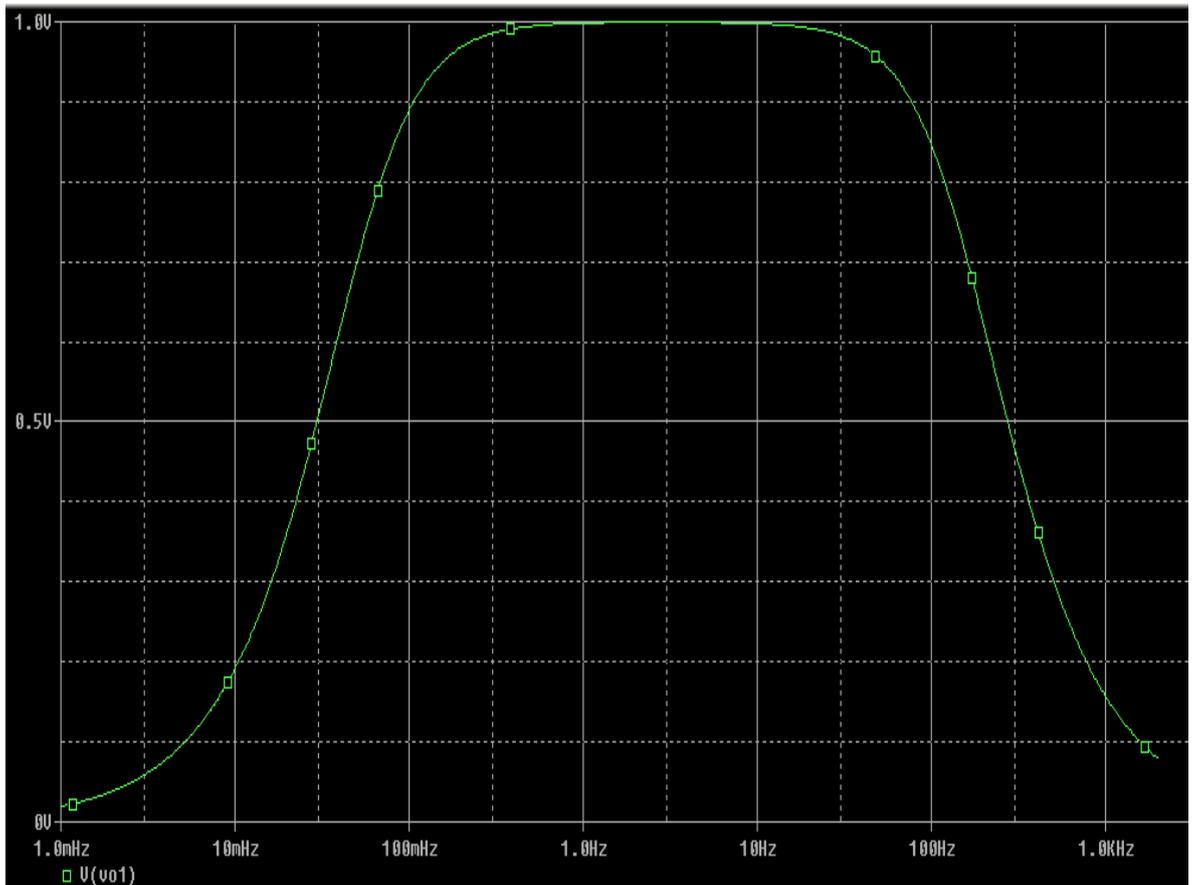
Figura 23. Filtro pasa alto.



Fuente: proteus 7 Professional - 10-03-08 6:40 PM.

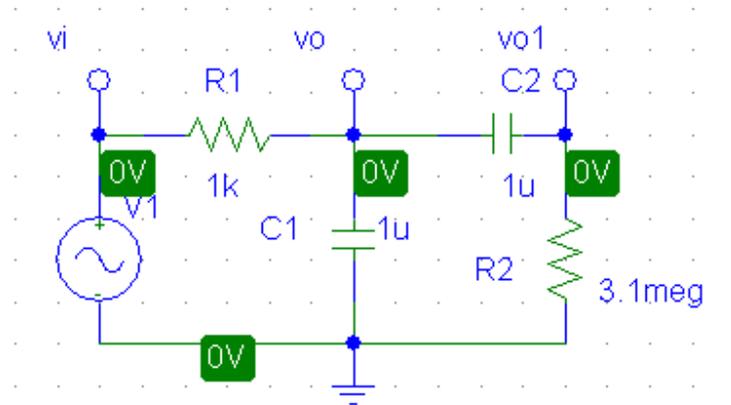
Ahora se procederá a la simulación final, para verificar que los cálculos estén bien hechos. Ver figura 24.

Figura 24. Filtros en función de la frecuencia.



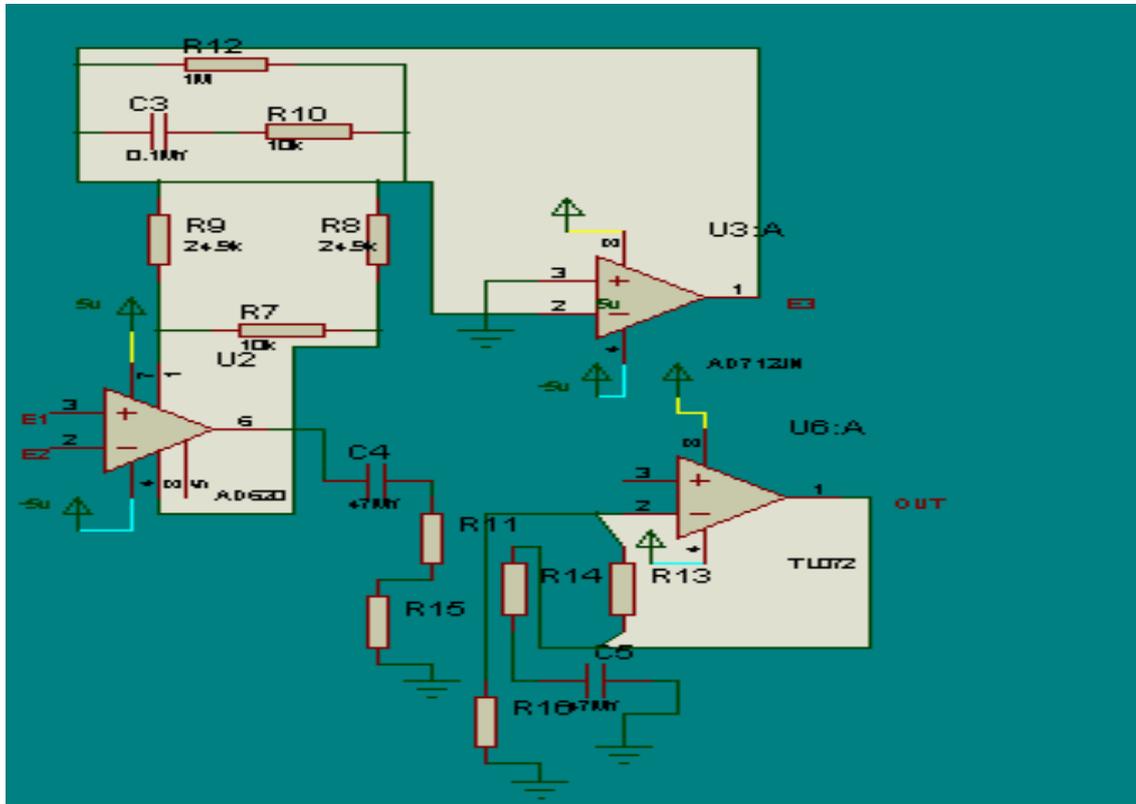
Fuente: proteus 7 Professional

Figura 25. Gráfica de los filtros pasa bajo y pasa alto.



Fuente: proteus 7 Professional - 12-03-08 5:40 P.m.

Figura 26. Amplificador instrumental



Fuente: proteus 7 Professional - 02-03-08 2:20 P.m.

4.3 PLANOS FINALES

Se observa en el diseño del esquemático y el PCB la conectividad de cada uno de los dispositivos que interactúan en el funcionamiento del historial digital de datos médicos, entre los que se encuentra el microcontrolador el cual es el cerebro del dispositivo y su respectiva configuración de funcionamiento, 4 pulsadores conectados a los pines 5, 6, 7 y 16 del microcontrolador, 2 reguladores de 5 (7805) y -5 (7905) voltios los cuales convierten el voltaje de entrada del dispositivo y alimentan los componentes de la tecnología CMOS, 1 cristal de cuarzo conectado a los pines 13 y 14 del microcontrolador, 1 transistor conectado al pin 18 del microcontrolador el cual protege el microcontrolador de las corrientes que pide el buzzer o sonador, 1 pantalla LCD, mediante la cual se decodifican los datos del dispositivo,

conectada desde el pin 33 al 40 del microcontrolador y por último la parte del electrocardiógrafo constituido por varios amplificadores operacionales que tienen como función amplificar las señales eléctricas de corazón para que mediante un proceso de digitalización puedan ser observadas por el ojo humano, este dispositivo va conectado al pin 23 del microcontrolador y el termómetro digital el cual está conectado al pin 3 del microcontrolador que es un conversor análogo digital.

FIGURA 27. Plano esquemático del dispositivo

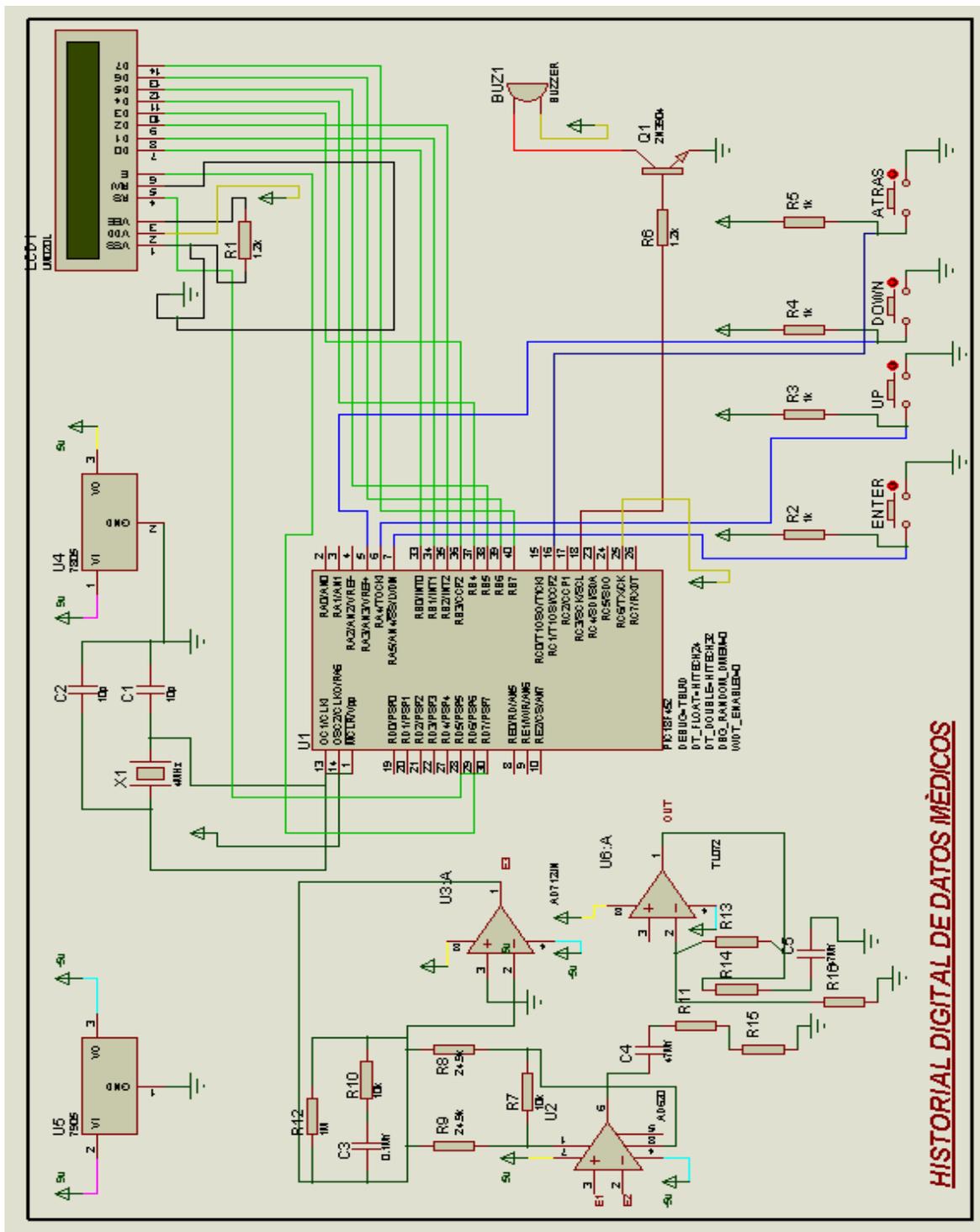


FIGURA 28. Esquema final del impreso

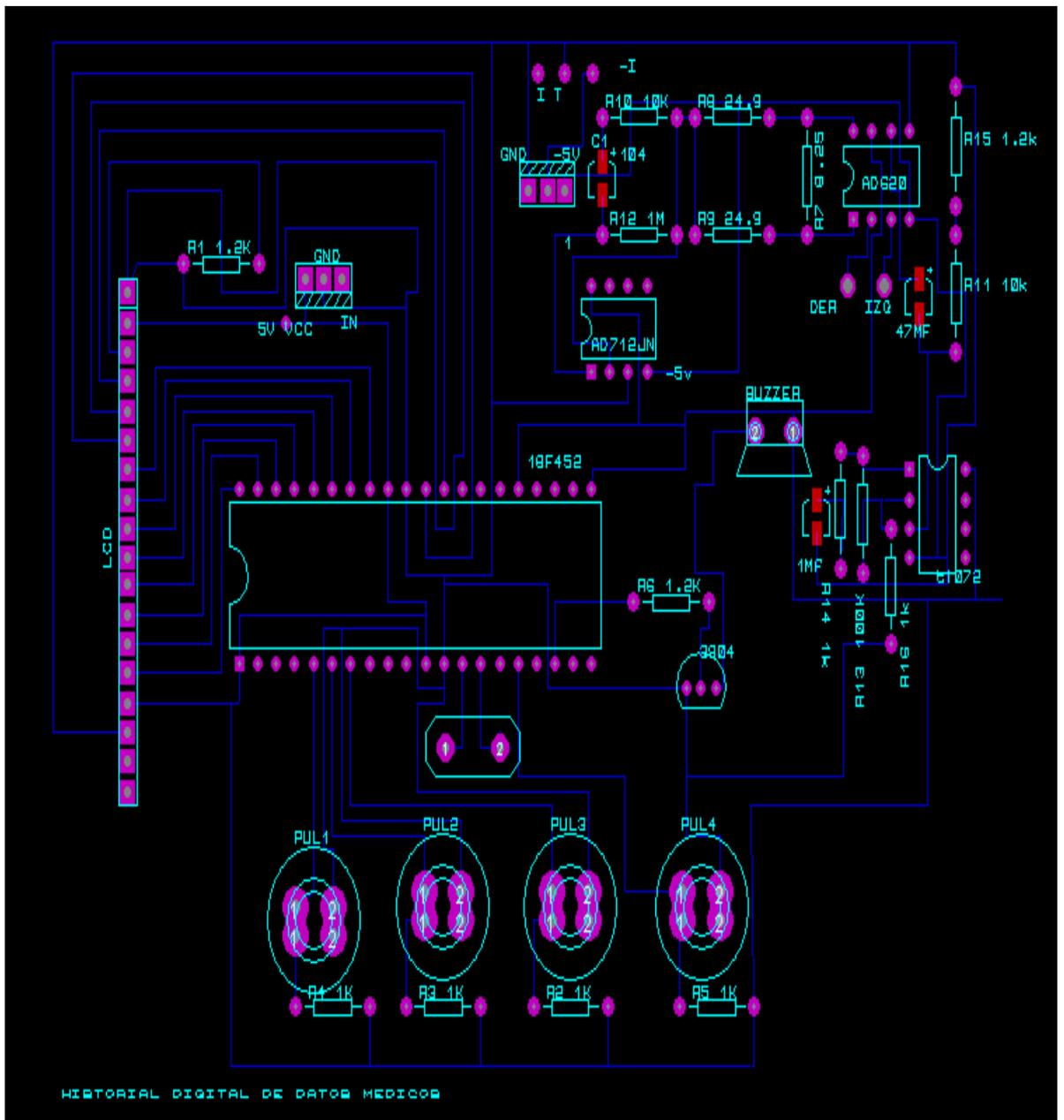
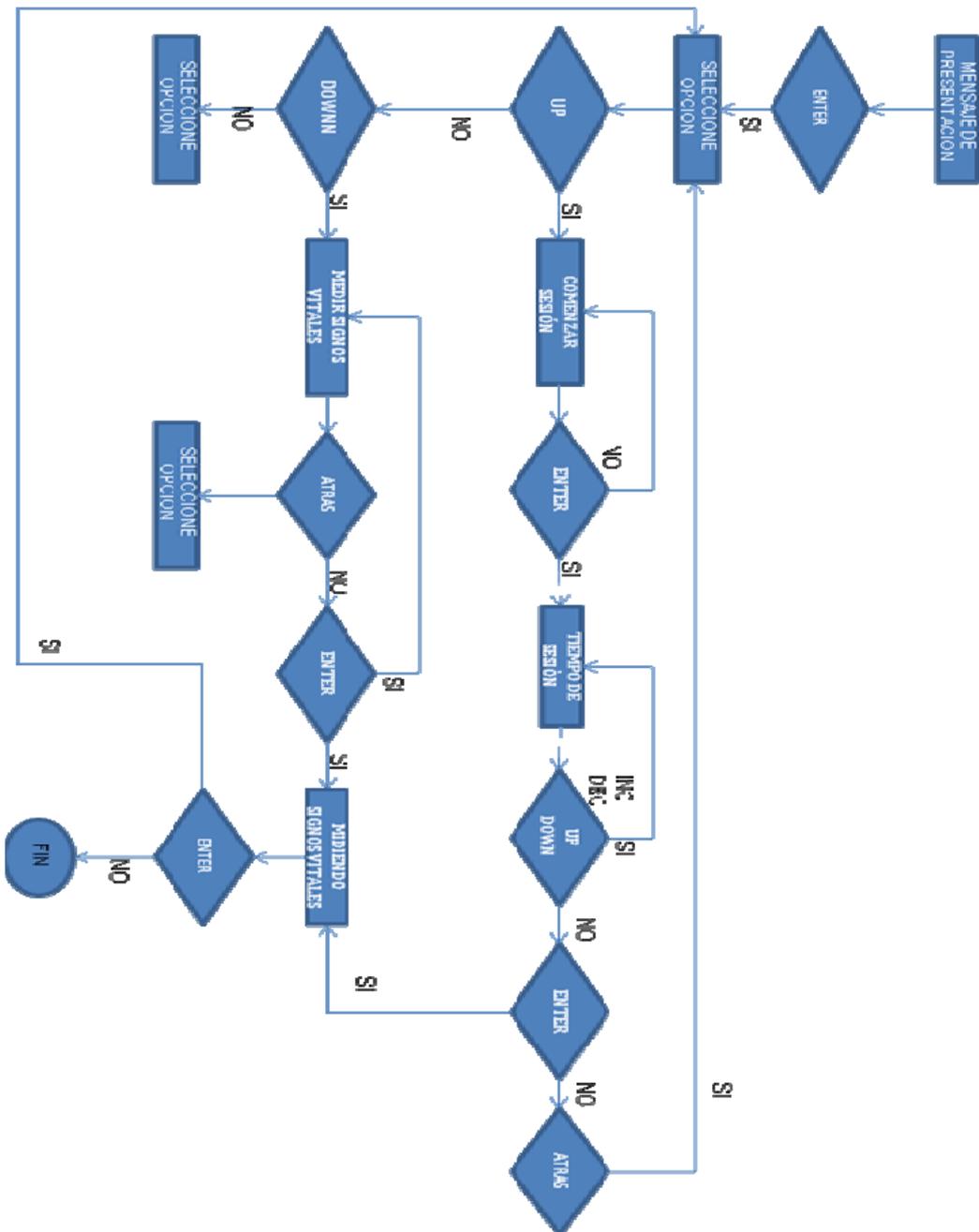


FIGURA 29. DIAGRAMA DE FLUJO ETAPA DIGITAL



4.4 DIAGRAMA DE FLUJO ETAPA DIGITAL

4.4.1 Mensaje de presentación.

En este bloque se visualiza el mensaje de bienvenida al usuario, de parte del diseñador del dispositivo para el usuario, haciendo énfasis a la institución a la que pertenece (universidad de san buenaventura).

4.4.2 Enter.

Se refiere al botón o pulsador enter, con el cual se comienza el subprograma del dispositivo.

4.4.3 Seleccione opción.

El dispositivo cuenta con un número de subprogramas el cual da comodidad al usuario según su necesidad, ya sea para varias tomas de signos vitales en una sesión con un tiempo determinado, una sola toma de signos vitales de una forma instantánea, ver los datos de la última sesión o toma de signos vitales, o borrar los datos de la última sesión de signos vitales con el fin de comenzar una nueva sesión.

4.4.4 Up-Down

Con estos dos pulsadores se escoge la opción deseada según la actividad a realizar con el paciente médico, luego de estar seguro de cual subprograma se utilizará aparecerán varias opciones según el subprograma escogido. Ej: si escoge iniciar sesión aparecerá la opción de escoja tiempo de sesión y luego de estar seguro de el tiempo de sesión según el estado de el paciente se oprime enter para comenzar sesión, si pulsa el botón **atrás** y lo mantiene oprimido luego de 5 segundos el programa se ira a la parte del menú de inicio.

Aparecen otras opciones como medir signos vitales oprimiendo **UP – DOWN** al dar enter en esta opción el dispositivo medirá y archivara en su memoria interna los signos vitales de una forma instantánea.

Cuando se tiene en la pantalla la opción de ver **datos medidos** después de haber oprimido ya sea **UP** o **DOWN**, cuando se oprime el pulsador enter el programa mostrara la ultima medida de signos vitales ya sea en una sesión o en una toma de signos vitales instantánea.

Luego de oprimir **UP-DOWN** llegara un momento en el cual aparecerá en la pantalla LCD una opción de borrar datos, se oprime **ENTER** en esta opción cuando se quiera borrar los datos de la memoria interna del microcontrolador (**EEPROM**), con el fin de hacer una nueva sesión ya sea con un paciente diferente.

4.4.5 Comenzar sesión.

En esta parte del diagrama de bloques se hace el inicio de sesión de la toma de signos vitales, única y exclusivamente para sesiones en las cuales interactúen varias medidas de signos vitales durante un tiempo determinado el cual escoge el especialista encargado, al tener oprimido el pulsador

ATRÁS Durante 5 segundos cuando este midiendo los signos vitales el dispositivo guarda los datos que tenga hasta el momento de oprimir dicho pulsador y a su vez visualiza los datos medidos hasta ese momento.

ANEXOS

CODIGO DE LINEA

SENSORES: DIGITALIZACIÓN DE LA FRECUENCIA CARDÍACA Y TEMPERATURA

```

                                Clrf   trisa
                                ; -----frecuencia cardiaca-----
-----
cero1 call   tic
            call   tic
            movlw .05
            movwf  tempo3
reviso1    movlw .10
            Movwf   tempo5
revisa1    movlw .100
            Call   demora
ciclo2 clrf adcon1
            Movlw   0x84 ; configurar canales eecom 1 canales de análo-
gos digitales
            movwf  adcon0
            movlw 0x49 ;frec traba
            movwf  adcon0
            bsf    adcon0,2 ;inicia la conversi?n a/d
            btfsc adcon0,2 ;espera que termine de convertir
            goto  $ -1
            movf  adresh, 0 ;resultado de conversi?n -> w
            movwf buffer ;guarda el dato obtenido de adresl en el buffer
temporal
            bcf    status,2 ;limpia el carry
            btfsc adresh, 0 ;mira el bit menos significativo de adresh (bit 8)
            bsf    status,2 ;si est? en 1 pone en uno el carry
            movf  buffer, 0 ;dato digitalizado -> w
            call  decimal ;obtiene uni, dec y cen con el agregado de
30h para la tabla ascii
            movlw 0x9a3 ;coloca el cursor en la posición 0dh de la panta-
lla.
            call  control
```

```

movf  cen, 0      ;coloca en el LCD las centenas
call  dato
movf  dec, 0      ;coloca las decenas
call  dato
movf  uni, 0      ;coloca las unidades
call  dato
decfsz      tempo5,1
goto  revisa1
decfsztempo3,1
goto  reviso1
movlw .100
call  demora
movff buffer, dato1

clrf  eeadr      ;direcciona el primer byte de la eeprom
movf  dato1,0
movwf  eedata      ;temp. activación -> eeprom
call  eesave

btfsc portc,1
goto  visualizar

```

```

;-----temperatura-----uno
ceroa call  tic
          movlw .05
          movwf tempo5
revisoa  movlw .10
          movwf tempo3
revisaa  movlw .100
          call  demora
          clrf  adcon1
          movlw 0x84
          movwf adcon0
          movlw 0x41
          movwf adcon0
          bsf   adcon0,2      ;inicia la conversi?n a/d
          btfsc adcon0,2      ;espera que termine de convertir
          goto  $ -1
          movf  adresh, 0      ;resultado de conversi?n -> w
          movwf buffer7      ;guarda el dato obtenido de adresl
en el buffer temporal
          bcf   status,2      ;limpia el carry

```

```

btfsc adresh, 0 ;mira el bit menos significativo de adresh (bit 8)
      bsf      status,2 ;si est? en 1 pone en uno el carry
      movf    buffer7, 0 ;dato digitalizado -> w
      call    decimal ;obtiene uni, dec y cen con el agre-
gado de 30h para la tabla ascii
      movlw   0xde3 ;coloca el cursor en la posición 0dh
de la pantalla.
      call    control
      movf    dec, 0 ;coloca las decenas
      call    dato
      movf    uni, 0 ;coloca las unidades
      call    dato
      decfsz tiempo3,1
      goto   revisaa
      decfsz tiempo5,1
      goto   revisoa
      movlw  .100
      call   demora
      movff  buffer7,dato7

      movld  '1'
      movwf  eeadr ;direcciona el primer byte de la ee-
prom
      movf  dato7,0
      movwf eedata ;temp. activation -> eeprom
      call  eesave

      call  cuadra2
      btfsc portc,1
      goto  visualizar

;-----dos-----
      call  tic
      call  tic
      movlw .100
      call  demora
cero2 call  tic
      movlw .05
      movwf tiempo3
reviso2  movlw .10
      movwf tiempo5
revisa2  movlw .100
      call  demora
      clrf  adcon1

```

```

movlw 0x84
movwf adcon0
movlw 0x49
movwf adcon0
bsf      adcon0,2      ;inicia la conversi?n a/d
btfsc   adcon0,2      ;espera que termine de convertir
goto    $-1
movf    adresh, 0      ;resultado de conversi?n -> w
movwf   buffer2        ;guarda el dato obtenido de adresl
en el buffer temporal
bcf      status,2      ;limpia el carry
btfsc   adresh, 0      ;mira el bit menos significativo de adresh
(bit 8)
bsf      status,2      ;si est? en 1 pone en uno el carry
movf    buffer2, 0      ;dato digitalizado -> w
call    decimal        ;obtiene uni, dec y cen con el agregado de
30h para la tabla ascii
movlw   0x9a3          ;coloca el cursor en la posici?n 0dh de la
pantalla.
call    control
movf    cen, 0          ;coloca en el LCD las centenas
call    dato
movf    dec, 0          ;coloca las decenas
call    dato
movf    uni, 0          ;coloca las unidades
call    dato
decfsz  tempo5,1
goto    revisa2
decfsz  tempo3,1
goto    reviso2
movlw   .100
call    demora
movff   buffer2,dato2

movlw   '2'
movwf   eeadr          ;direcciona el primer byte de la eeprom
movf    dato2,0
movwf   eedata        ;temp. activaci?n -> eeprom
call    eesave

btfsc   portc,1
goto    visualizar

```

```

;-----temperatura-----dos
cerob call  tic

                movlw .05
                movwf tempo3
revisob        movlw .10
                movwf tempo5
revisab        movlw .100
                call  demora
                call  demora
                clrf  adcon1
                movlw 0x84
                movwf adcon0
                movlw 0x41
                movwf adcon0
                bsf   adcon0,2           ;inicia la conversi?n a/d
                btfsc adcon0,2         ;espera que termine de convertir
                goto  $ -1
                movf  adresh, 0        ;resultado de conversi?n -> w
                movwf buffer8         ;guarda el dato obtenido de adresl
en el buffer temporal
                bcf   status,2         ;limpia el carry
                btfsc adresh, 0        ;mira el bit menos significativo de
adresh (bit 8)
                bsf   status,2         ;si est? en 1 pone en uno el carry
                movf  buffer8, 0        ;dato digitalizado -> w
                call  decimal          ;obtiene uni, dec y cen con el agre-
gado de 30h para la tabla ascii
                movlw 0xde3           ;coloca el cursor en la posici?n 0dh
de la pantalla.
                call  control

                movf  dec, 0           ;coloca las decenas
                call  dato
                movf  uni, 0          ;coloca las unidades
                call  dato

                decfsz tempo5, 1
                goto  revisab
                decfsz tempo3, 1
                goto  revisob

```

```

movlw .100
call demora
movlw .100
call demora
call tic
movff buffer8,dato8

prom      movlwd'3'
          movwfeadr           ;direcciona el primer byte de la ee-
          movf dato8,0
          movwf eedata       ;temp. activación -> eeprom
          call eesave
          call cuadra3
          btfsc portc,1
          goto visualizar
;-----tres-----
          call tic
          call tic

cero3 call movlw .100
          call demora
          tic
          movlw .05
reviso3   movwf tempo3
          movlw .10
          movwf tempo5
revisa3   movlw .100
          call demora
          clrf adcon1
          movlw 0x84
          movwf adcon0
          movlw 0x49
          movwf adcon0
          bsf adcon0,2       ;inicia la conversi?n a/d
          btfsc adcon0,2    ;espera que termine de convertir
          goto $-1
          movf adresh, 0    ;resultado de conversi?n -> w
          movwf buffer3    ;guarda el dato obtenido de adresl
en el buffer temporal
          bcf status,2      ;limpia el carry
          btfsc adresh, 0   ;mira el bit menos significativo de
adresh (bit 8)

```

```

        bsf          status,2           ;si est? en 1 pone en uno el carry
        movf  buffer3, 0                ;dato digitalizado -> w
        call  decimal                   ;obtiene uni, dec y cen con el agre-
gado de 30h para la tabla ascii
        movlw 0x9a3                     ;coloca el cursor en la posición 0dh
de la pantalla.
        call  control
        movf  cen, 0                    ;coloca en el LCD las centenas
        call  dato
        movf  dec, 0                    ;coloca las decenas
        call  dato
        movf  uni, 0                    ;coloca las unidades
        call  dato
        decfsztempo5,1
        goto  revisa3
        decfsztempo3,1
        goto  reviso3
        movlw .100
        call  demora
        movff buffer3,dato3

        movld'4'
        movwfeadr                       ;direcciona el primer byte de la ee-
prom
        movf  dato3,0
        movwf  eedata                    ;temp. activación -> eeprom
        call  eesave
        btfsc portc,1
        goto  visualizar

;-----temperatura-----tres
        call  tic

        movlw .100
        call  demora
ceroc call tic
        movlw .05
        movwf tempo3
revisoc  movlw .10
        movwf tempo5
revisac  movlw .100
        call  demora
        call  demora
        clrf  adcon1
        movlw 0x84
        movwf adcon0

```

```

movlw 0x41
movw fadcon0
bsf      adcon0,2      ;inicia la conversi?n a/d
btfsc   adcon0,2      ;espera que termine de convertir
goto    $-1
movf    adresh, 0     ;resultado de conversi?n -> w
movw    buffer9      ;guarda el dato obtenido de adresl
en el buffer temporal
bcf      status,2     ;limpia el carry
btfsc   adresh, 0     ;mira el bit menos significativo de
adresh (bit 8)
bsf      status,2     ;si est? en 1 pone en uno el carry
movf    buffer9, 0    ;dato digitalizado -> w
call    decimal      ;obtiene uni, dec y cen con el agre-
gado de 30h para la tabla ascii
movlw   0xde3        ;coloca el cursor en la posici?n 0dh
de la pantalla.
call    control

movf    dec, 0        ;coloca las decenas
call    dato
movf    uni, 0        ;coloca las unidades
call    dato
decfsz  tiempo5,1
goto    revisac
decfsz  tiempo3,1
goto    revisoc
movlw   .100
call    demora
movlw   .100
call    demora
movff   buffer9,dato9

movlw   d'5'
movw    ffeedr      ;direcciona el primer byte de la ee-
prom
movf    dato9,0
movw    feedata
call    eesave

call    cuadra4
btfsc   portc,1
goto    visualizar

```

```

;-----frecuencia cardiaca-----
-----
call tic
call tic

movlw .100
call demora
cero4 call tic
movlw .05
movwftempo3
reviso4 movlw .10
movwftempo5
revisa4 movlw .100
call demora
clrf adcon1
movlw 0x84
movwfadcon0
movlw 0x49
movwfadcon0
bsf adcon0,2
btfsc adcon0,2 ;espera que termine de convertir
goto $ -1
movf adresh, 0 ;resultado de conversi?n -> w
movwfbuffer4 ;guarda el dato obtenido de adresl en el
buffer temporal
bcf status,2 ;limpia el carry
btfsc adresh, 0 ;mira el bit menos significativo de adresh
(bit 8)
bsf status,2
movf buffer4, 0 ;dato digitalizado -> w
call decimal ;obtiene uni, dec y cen con el agregado de
30h para la tabla ascii ; pantalla.
movlw 0x9a3
call control
movf cen, 0 ;coloca en el lcd las centenas
call dato
movf dec, 0 ;coloca las decenas
call dato
movf uni, 0 ;coloca las unidades
call dato
decfsztempo5,1
goto revisa4
decfsztempo3,1
goto reviso4
movff buffer4,dato4

```

```

movlw d'6'
movwfeadr           ;direcciona el primer byte de la eeprom
movf dato4,0
movwf eedata
call eesave

btfsc portc,1
goto visualizar

;-----temperatura-----cuatro
call tic

cerod call
movlw .100
call demora
tic
movlw .05
movwftempo3
revisod
movlw .10
movwftempo5
revisad
movlw .100
call demora
call demora
clrf adcon1
movlw 0x84
movwfadcon0
movlw 0x41
movwfadcon0
bsf          adcon0,2
btfsc adcon0,2           ;espera que termine de convertir
goto $ -1
movf adresh, 0
movwfbuffer10           ;guarda el dato obtenido de adresl en el
buffer temporal
bcf          status,2           ;limpia el carry
btfsc adresh, 0           ;mira el bit menos significativo de adresh
(bit 8)
bsf          status,2
movf buffer10, 0           ;dato digitalizado -> w
call decimal           ;obtiene uni, dec y cen con el agregado de
30h para la tabla ascii
movlw 0xde3           ;coloca el cursor en la posición 0dh de la
pantalla.
call control

```

```

movf dec, 0           ;coloca las decenas
call dato
movf uni, 0          ;coloca las unidades
call dato
decfsztempo5,1
goto revisad
decfsztempo3,1
goto revisod
movff buffer10,dato10

movlwd'7'
movwfeadr           ;direcciona el primer byte de la eeprom
movf dato10,0
movwf eedata
call eesave

call cuadra5
btfsc portc,1
goto visualizar

```

```

; ----- frecuencia cardiaca -----
call tic
call tic

movlw .100
call demora
cero5 call tic
movlw .05
reviso5 movwf tempo3
movlw .10
revisa5 movwf tempo5
movlw .100
call demora
clrf adcon1
movlw 0x84
movwf adcon0
movlw 0x49
movwf adcon0
bsf adcon0,2 ;
btfsc adcon0,2 ;espera que termine de convertir
goto $-1
movf adresh, 0 movwf buffer5
buffer temporal ;guarda el dato obtenido de adresh en el

```

```

                bcf          status,2           ;limpia el carry
                btfsc   adresh, 0             ;mira el bit menos significativo de adresh
(bit 8)
                bsf          status,2           ;si est? en 1 pone en uno el carry
                movf   buffer5, 0             ;dato digitalizado -> w
                call    decimal              ;obtiene uni, dec y cen con el agregado de
30h para la tabla ascii
                movlw
                0x9a3
                call    control
                movf   cen, 0                 ;coloca en el lcd las centenas
                call    dato
                movf   dec, 0                 ;coloca las decenas
                call    dato
                movf   uni, 0                 ;coloca las unidades
                call    dato
                decfsztempo5,1
                goto   revisa5
                decfsztempo3,1
                goto   reviso5
                movff  buffer5,dato5

                movlwd'8'
                movwfeeadr                    ;direcciona el primer byte de la eeprom
                movf   dato5,0
                movwf   eedata
                call    eesave

                btfsc   portc,1
                goto   visualizar

; -----temperatura-----quinto
                call    tic

                movlw .100
                call    demora
ceroe call    tic
                movlw .05
revisoe      movwftempo3
                movlw .10
revisae      movwftempo5
                movlw .100
                call    demora
                clrf   adcon1
                movlw 0x84
                movwf   adcon0

```

```

movlw 0x41
movw fadcon0
bsf      adcon0,2
btfsc   adcon0,2
goto    $-1
movf    adresh, 0
movwf   buffer11
bcf     status,2
btfsc   adresh, 0
bsf     status,2

```

```

movf    buffer11, 0
call    decimal
movlw   0xde3
call    control
movf    dec, 0
call    dato
movf    uni, 0
call    dato
decfsz  tempo5,1
goto    revisae
decfsz  tempo3,1
goto    revisoe
movff   buffer11,dato11

```

```

movlw   d'9'
movwf   eeadr           ;direcciona el primer byte de la eeprom
movf    dato11,0
movwf   eedata
call    eesave

```

```

call    cuadra6
btfsc   portc,1
goto    visualizar

```

-----**sextofrecuencia cardiaca**-----

```

call    tic
call    tic

movlw   .100
call    demora
cero6   call    tic
movlw   .05
movwf   tempo3

```

```

reviso6      movlw .10
              movwftempo5
revisa6      movlw .100
              call  demora
              clrf  adcon1
              movlw 0x84
              movwfadcon0
              movlw 0x49
              movwfadcon0
              bsf   adcon0, 2           ; inicia la conversión a/d
              btfsc adcon0,2         ; espera que termine de convertir
              goto  $ -1
              movf  adresh, 0         ; resultado de conversión -> w
              movwfbuffer6           ; guarda el dato obtenido de adresh en el
buffer temporal
              bcf   status,2         ; limpia el carry
              btfsc adresh, 0       ; mira el bit menos significativo de adresh
(bit 8)
              bsf   status,2         ; si está en 1 pone en uno el carry
              movf  buffer6, 0       ; dato digitalizado -> w
              call  decimal          ; obtiene uni, dec y cen con el agregado de
30h para la tabla ascii
              movlw 0x9a3           ; coloca el cursor en la posición 0dh de la
pantalla.
              call  control
              movf  cen, 0           ; coloca en el LCD las centenas
              call  dato
              movf  dec, 0          ; coloca las decenas
              call  dato
              movf  uni, 0          ; coloca las unidades
              call  dato
              decfsztempo5,1
              goto  revisa6
              decfsztempo3,1
              goto  reviso6
              movff buffer6,dato6

              movld'10'
              movwf eaddr           ; direcciona el primer byte de la eeprom
              movf  dato6,0
              movwf eedata         ; temp. activation -> eeprom
              call  eesave

              btfsc portc,1

```

```

                goto    visualizar

;-----temperatura-----sexto
                call   tic
                movlw .100
                call   demora
cerof call      tic
                movlw .05
                movwftempo3
revisofmovlw   .10
                movwftempo5
revisafmovlw  .100
                call   demora
                call   demora
                clrf   adcon1
                movlw 0x84
                movwftadcon0
                movlw 0x41
                movwftadcon0
                bsf    adcon0,2           ;inicia la conversi?n a/d
                btfsc adcon0,2           ;espera que termine de convertir
                goto  $ -1
                movf  adresh, 0           ;resultado de conversi?n -> w
                movwfbuffer12            ;guarda el dato obtenido de adresl
en el buffer temporal
                bcf    status,2           ;limpia el carry
                btfsc adresh, 0           ;mira el bit menos significativo de
adresh (bit 8)
                bsf    status,2           ;si est? en 1 pone en uno el carry
                movf  buffer12, 0         ;dato digitalizado -> w
                call   decimal           ;obtiene uni, dec y cen con el agre-
gado de 30h para la tabla ascii
                movlw 0xde3              ;coloca el cursor en la posici?n 0dh
de la pantalla.
                call   control

                movf  dec, 0              ;coloca las decenas
                call   dato
                movf  uni, 0              ;coloca las unidades
                call   dato

                decfsztempo5,1
                goto  revisar
                decfsztempo3,1
                goto  reviso,f

```

```

        call  cuadra
        movff buffer12,dato12

prom   movlw d'11'
        movwf eeadr           ;direcciona el primer byte de la ee-

        movf  dato12,0
        movwf eedata         ;temp. activación -> eeprom
        call  eesave

cep1   btfsc portc,1
        goto  visualizar

        movlw .255
        call  demora

cep    btfss portc,1
        goto  visualizar
        goto  visualizar

```

VISUALIZACION.

```

apagado   bcf  portd,5
explorar  btfss portc,6
          goto  apagado
          bsf  portd,5
          calltic
          call  retardo

          call  retardo           ;retardo trae implicito el off

men       bcf  portc,1
          call  limpiar
          clrf time
          movlw 0x80           ;posiciona el cursor en la 1a. linea
          call  lcd

          movlw" "
          call  dato

```

```

movlw" "
call dato
movlw" "
call dato
movlw" "
call dato
movlw"u" ;mensaje: historial digital de
call dato
movlw"n"
call dato
movlw"i"
call dato
movlw"v"
call dato
movlw"e"
call dato
movlw"r"
call dato
movlw"s"
call dato
movlw"i"
call dato
movlw"d"
call dato
movlw"a"
call dato
movlw"d"
call dato
movlw" "
call dato

```

```

movlw0xc0 ;posiciona el cursor en la 2da. linea
call lcd

movlw"d"
call dato

```

```

movlw"e"
call dato
movlw" "
call dato
movlw"s"
call dato
movlw"a"
call dato
movlw"n"
call dato
movlw" "
call dato
movlw"b"
call dato
movlw"u"
call dato
movlw"e"
call dato
movlw"n"
call dato
movlw"a"
call dato
movlw"v"
call dato
movlw"e"
call dato
movlw"n"
call dato
movlw"t"
call dato
movlw"u"
call dato
movlw"r"
call dato
movlw"a"
call dato
movlw" "
call dato
;mensaje: historial digital de

movlw0x94
call lcd
;posiciona el cursor en la 2da. linea

movlw"h"
call dato
movlw"i"

```

```
call dato
movlw"s"
call dato
movlw"t"
call dato
movlw"o"
call dato
movlw"r"
call dato
movlw"i"
call dato
movlw"a"
call dato
movlw"l"
call dato
movlw" "
call dato
movlw"d"
call dato
movlw"i"
call dato
movlw"g"
call dato
movlw"i"
call dato
movlw"t"
call dato
movlw"a"
call dato
movlw"l"
call dato
movlw" "
call dato
movlw"d"
call dato
movlw"e"
call dato
```

```
movlw0xd4
call lcd
```

;posiciona el cursor en la 2da. linea

```
movlw" "
call dato
movlw" "
call dato
```

```

movlw" "
call dato
movlw"d"
call dato
movlw"a"
call dato
movlw"t"
call dato
movlw"o"
call dato
movlw"s"
call dato
movlw" "
call dato
movlw"m"
call dato
movlw"e"
call dato
movlw"d"
call dato
movlw"i"
call dato
movlw"c"
call dato
movlw"o"
call dato
movlw"s"
call dato
movlw" "

```

```
call cencillo2
```

```

movlw0x80
call lcd

movlw" "

```

```
;posiciona el cursor en la 1a. linea
```

```
call dato
movlw" "
call dato
movlw" "
call dato
movlw"p"
call dato
movlw"r"
call dato
movlw"e"
call dato
movlw"s"
call dato
movlw"i"
call dato
movlw"o"
call dato
movlw"n"
call dato
movlw"e"
call dato
movlw" "
call dato
movlw"e"
call dato
movlw"n"
call dato
movlw"t"
call dato
movlw"e"
call dato
movlw"r"
call dato
movlw" "
call dato
movlw" "
call dato
movlw" "
call dato
```

;mensaje: historial digital de

```
movlw0xc0
call lcd
```

;posiciona el cursor en la 2da. linea

```
movlw" "
```



```

enter  btfss  porta,5
        goto  enter

        movlw .255
        call  demora

        call  retardo
        btfsc portc,1      ;atras
        goto  do
        call  tic

do      call  tic
        clrf  trise
        clrf  porta
        clrf  portb
        clrf  portc
        clrf  portd
        clrf  porte
        call  limpiar
        movlw 0x83      ;posiciona el cursor en la 1a. linea
        call  lcd

        movlw "p"
        call  dato
        movlw "r"
        call  dato
        movlw "e"
        call  dato
        movlw "s"
        call  dato
        movlw "i"
        call  dato
        movlw "o"
        call  dato
        movlw "n"
        call  dato
        movlw "e"
        call  dato
        movlw " "
        call  dato
        movlw "e"
        call  dato
        movlw "n"
        call  dato

```

```
movlw"t"  
call dato  
movlw"e"  
call dato  
movlw"r"  
call dato
```

```
movlw0xc4  
call lcd  
movlw"p"  
call dato  
movlw"r"  
call dato  
movlw"e"  
call dato  
movlw"s"  
call dato  
movlw"i"  
call dato  
movlw"o"  
call dato  
movlw"n"  
call dato  
movlw"e"  
call dato  
movlw" "  
call dato  
movlw"u"  
call dato  
movlw"p"  
call dato
```

;posiciona el cursor en la 2da. linea

```
movlw0xc2  
call lcd  
movlw"p"  
call dato  
movlw"r"  
call dato  
movlw"e"  
call dato  
movlw"s"  
call dato  
movlw"i"
```

;posiciona el cursor en la 2da. linea

```
call dato
movlw"o"
call dato
movlw"n"
call dato
movlw"e"
call dato
movlw" "
call dato
movlw"u"
call dato
movlw"p"
call dato
movlw" "
call dato
movlw"d"
call dato
movlw"o"
call dato
movlw"w"
call dato
movlw"n"
call dato
```

```
movlw0x96
call lcd
```

;posiciona el cursor en la 1a. linea

```
movlw"s"
call dato
movlw"e"
call dato
movlw"l"
call dato
movlw"e"
call dato
movlw"c"
call dato
movlw"c"
call dato
movlw"i"
call dato
movlw"o"
call dato
movlw"n"
```

```
call dato
movlw"e"
call dato
movlw" "
call dato
movlw"o"
call dato
movlw"p"
call dato
movlw"c"
call dato
movlw"i"
call dato
movlw"o"
call dato
movlw"n"
call dato
```

ta

```
movlw0xd6
call lcd
```

;posiciona el cursor en la 2da. linea

```
movlw"c"
call dato
movlw"o"
call dato
movlw"m"
call dato
movlw"e"
call dato
movlw"n"
call dato
movlw"z"
call dato
movlw"a"
call dato
movlw"r"
call dato
movlw" "
call dato
movlw"s"
call dato
movlw"e"
call dato
movlw"s"
call dato
```

```

movlw "i"
call dato
movlw "o"
call dato
movlw "n"
call dato
movlw "?"
call dato

                bcf          porta,3
                bcf          porta,5
uno            btfsc   porta,4          ;up
                call    re
                btfsc   porta,5          ;enter
                goto    fa
                btfsc   porta,3          ;down
                goto    bora
                btfsc   porta,5          ;enter
                goto    fa
                goto    uno

re            call    tic
                movlw 0x96          ;posiciona el cursor en la 1a. linea
                call    lcd

                movlw "s"
                call    dato
                movlw "e"
                call    dato
                movlw "l"
                call    dato
                movlw "e"
                call    dato
                movlw "c"
                call    dato
                movlw "c"
                call    dato
                movlw "i"
                call    dato
                movlw "o"
                call    dato
                movlw "n"
                call    dato

```

```
movlw"e"  
call dato  
movlw" "  
call dato  
movlw"o"  
call dato  
movlw"p"  
call dato  
movlw"c"  
call dato  
movlw"i"  
call dato  
movlw"o"  
call dato  
movlw"n"  
call dato
```

```
movlw0xd4  
call lcd  
movlw"m"  
call dato  
movlw"e"  
call dato  
movlw"d"  
call dato  
movlw"i"  
call dato  
movlw"r"  
call dato  
movlw" "  
call dato  
movlw"s"  
call dato  
movlw"i"  
call dato  
movlw"g"  
call dato  
movlw"n"  
call dato  
movlw"o"  
call dato  
movlw"s"  
call dato  
movlw" "  
call dato
```

;posiciona el cursor en la 2da. linea

```

movlw "v"
call dato
movlw "i"
call dato
movlw "t"
call dato
movlw "a"
call dato
movlw "l"
call dato
movlw "e"
call dato
movlw "s"
call dato
clrf time

devu btfsc porta,4          ;up
      call med
      btfsc porta,5          ;enter
      goto mi
      btfsc porta,3          ;down
      goto do
      btfsc porta,5          ;enter
      goto mi
      goto devu

```

```

; BASE DE DATOS

```

```

;dato1
      clrf eeadr
      bcf eecon1,7
      bcf eecon1,6
      bsf eecon1,0
      movf eedata, 0
      movwfdato1

```

```

;dato2
      incf eeadr
      bcf eecon1,7
      bcf eecon1,6
      bsf eecon1,0
      movf eedata, 0
      movwfdato7

```

```

;dato3

```

```

incf eeadr
bcf eecon1,7
bcf eecon1,6
bsf eecon1,0
movf eedata, 0
movwfdato2
;dato4
incf eeadr
bcf eecon1,7
bcf eecon1,6
bsf eecon1,0
movf eedata, 0
movwfdato8
;dato5
incf eeadr
bcf eecon1,7 ;direcciona la primera posici?n de la ee-
prom
bcf eecon1,6 ;selecciona el banco de eeprom de datos
bsf eecon1,0
movf eedata, 0 ;dato le?do de la eeprom -> w
movwfdato3
;dato6
incf eeadr
bcf eecon1,7 ;direcciona la primera posici?n de la ee-
prom
bcf eecon1,6 ;selecciona el banco de eeprom de datos
bsf eecon1,0
movf eedata, 0 ;dato le?do de la eeprom -> w
movwfdato9
;dato7
incf eeadr
bcf eecon1,7 ;direcciona la primera posici?n de la ee-
prom
bcf eecon1,6 ;selecciona el banco de eeprom de datos
bsf eecon1,0
movf eedata, 0 ;dato le?do de la eeprom -> w
movwfdato4
;dato8
incf eeadr
bcf eecon1,7 ;direcciona la primera posici?n de la ee-
prom

```

```

bcf          eecon1,6    ;selecciona el banco de eeprom de datos
bsf          eecon1,0
movf  eedata, 0        ;dato le?do de la eeprom -> w
movwfdato10

;dato9
incf  eeadr
bcf          eecon1,7    ;direcciona la primera posici?n de la ee-
prom
bcf          eecon1,6    ;selecciona el banco de eeprom de datos
bsf          eecon1,0
movf  eedata, 0        ;dato le?do de la eeprom -> w
movwfdato5

;dato10
incf  eeadr
bcf          eecon1,7    ;direcciona la primera posici?n de la ee-
prom
bcf          eecon1,6    ;selecciona el banco de eeprom de datos
bsf          eecon1,0
movf  eedata, 0        ;dato le?do de la eeprom -> w
movwfdato11

;dato11
incf  eeadr
bcf          eecon1,7    ;direcciona la primera posici?n de la ee-
prom
bcf          eecon1,6    ;selecciona el banco de eeprom de datos
bsf          eecon1,0
movf  eedata, 0        ;dato le?do de la eeprom -> w
movwfdato6

;dato12
incf  eeadr
bcf          eecon1,7    ;direcciona la primera posici?n de la ee-
prom
bcf          eecon1,6    ;selecciona el banco de eeprom de datos
bsf          eecon1,0
movf  eedata, 0        ;dato le?do de la eeprom -> w
movwfdato12

clrf  eeadr            ;direcciona el primer byte de la eeprom
movf  dato1,0
movwfeedata            ;temp. activaci?n -> eeprom
call  eesave

```

```

movlwd'1'
movwfeadr      ;direcciona el primer byte de la eeprom
movf dato7,0
movwfeedata    ;temp. activaci?n -> eeprom
call eesave

```

```

movlwd'2'
movwfeadr      ;direcciona el primer byte de la eeprom
movf dato2,0
movwfeedata    ;temp. activaci?n -> eeprom
call eesave

```

```

movlwd'3'
movwfeadr      ;direcciona el primer byte de la eeprom
movf dato8,0
movwfeedata    ;temp. activaci?n -> eeprom
call eesave

```

```

movlwd'4'
movwfeadr      ;direcciona el primer byte de la eeprom
movf dato3,0
movwfeedata    ;temp. activaci?n -> eeprom
call eesave

```

```

movlwd'5'
movwfeadr      ;direcciona el primer byte de la eeprom
movf dato9,0
movwfeedata    ;temp. activaci?n -> eeprom
call eesave

```

```

movlwd'6'
movwfeadr      ;direcciona el primer byte de la eeprom
movf dato4,0
movwfeedata    ;temp. activaci?n -> eeprom
call eesave

```

```

movlwd'7'
movwfeadr      ;direcciona el primer byte de la eeprom
movf dato10,0
movwfeedata    ;temp. activaci?n -> eeprom
call eesave

```

```

movlwd'8'
movwfeadr      ;direcciona el primer byte de la eeprom
movf dato5,0
movwfeedata    ;temp. activaci?n -> eeprom
call eesave
movlwd'9'
movwfeadr      ;direcciona el primer byte de la eeprom
movf dato11,0
movwfeedata    ;temp. activaci?n -> eeprom
call eesave

movlwd'10'
movwfeadr      ;direcciona el primer byte de la eeprom
movf dato6,0
movwfeedata    ;temp. activaci?n -> eeprom
call eesave

movlwd'11'
movwfeadr      ;direcciona el primer byte de la eeprom
movf dato12,0
movwfeedata    ;temp. activaci?n -> eeprom
call eesave

;dato1
clrf eeadr
bcf eecon1,7
bcf eecon1,6
bsf eecon1,0
movf eedata, 0
movwfdato1
;dato2
incf eeadr
bcf eecon1,7
bcf eecon1,6
bsf eecon1,0
movf eedata, 0
movwfdato7

;dato3
incf eeadr
bcf eecon1,7
bcf eecon1,6
bsf eecon1,0
movf eedata, 0

```

```

movwfdato2
;dato4
incf eeadr
bcf eecon1,7
bcf eecon1,6
bsf eecon1,0
movf eedata, 0
movwfdato8

;dato5
incf eeadr
bcf eecon1,7 ;direcciona la primera posici?n de la ee-
prom
bcf eecon1,6 ;selecciona el banco de eeprom de datos
bsf eecon1,0
movf eedata, 0 ;dato le?do de la eeprom -> w
movwfdato3

;dato6
incf eeadr
bcf eecon1,7 ;direcciona la primera posici?n de la ee-
prom
bcf eecon1,6 ;selecciona el banco de eeprom de datos
bsf eecon1,0
movf eedata, 0 ;dato le?do de la eeprom -> w
movwfdato9

;dato7
incf eeadr
bcf eecon1,7 ;direcciona la primera posici?n de la ee-
prom
bcf eecon1,6 ;selecciona el banco de eeprom de datos
bsf eecon1,0
movf eedata, 0 ;dato le?do de la eeprom -> w
movwfdato4

;dato8
incf eeadr
bcf eecon1,7 ;direcciona la primera posici?n de la ee-
prom
bcf eecon1,6 ;selecciona el banco de eeprom de datos
bsf eecon1,0
movf eedata, 0 ;dato le?do de la eeprom -> w
movwfdato10

```

```

;dato9
incf eeadr
bcf eecon1,7 ;direcciona la primera posici?n de la ee-
prom
bcf eecon1,6 ;selecciona el banco de eeprom de datos
bsf eecon1,0
movf eedata, 0 ;dato le?do de la eeprom -> w
movwfdato5

```

```

;dato10
incf eeadr
bcf eecon1,7 ;direcciona la primera posici?n de la ee-
prom
bcf eecon1,6 ;selecciona el banco de eeprom de datos
bsf eecon1,0
movf eedata, 0 ;dato le?do de la eeprom -> w
movwfdato11

```

```

;dato11
incf eeadr
bcf eecon1,7 ;direcciona la primera posici?n de la ee-
prom
bcf eecon1,6 ;selecciona el banco de eeprom de datos
bsf eecon1,0
movf eedata, 0 ;dato le?do de la eeprom -> w
movwfdato6

```

```

;dato12
incf eeadr
bcf eecon1,7 ;direcciona la primera posici?n de la ee-
prom
bcf eecon1,6 ;selecciona el banco de eeprom de datos
bsf eecon1,0
movf eedata, 0 ;dato le?do de la eeprom -> w
movwfdato12

```

SUBRUTINAS DE LOS PULSADORES

```

bcf porta,3
bcf porta,5
uno btfsc porta,4 ;up
call re

```

```

                btfsc porta,5          ;enter
                goto fa
                btfsc porta,3          ;down
                goto bora
                btfsc porta,5          ;enter
                goto fa
goto            uno

devu btfsc porta,4          ;up
      call med
      btfsc porta,5          ;enter
      goto mi
      btfsc porta,3          ;down
      goto do
      btfsc porta,5          ;enter
      goto mi
      goto devu

revisar15 btfsc porta,4          ;up
          call bora
          btfsc porta,5          ;enter
          goto visualizar
          btfsc porta,3          ;down
          goto re
          btfsc porta,5          ;enter
          goto visualizar
          goto revisar15

```

SUBROUTINAS DE LOS PULSADORES

```

control    bcf          portd,6
           goto enviar
dato bsf          portd,6
enviar movwfpportb
           movlw .1
           call demora
           bsf          portd,7
           movlw .1
           call demora

```

```

        bcf          portd,7
        movlw .1
        call  demora
        return

tic          bsf          portc,3
            movlwd'100'
            call  demora
            bcf          portc,3
            return

limpiar     movlw .2
            call  demora
            movlw0x01
            call  control
            movlw .2
            call  demora
            return

demora      movwftiempo2          ;demora tantos milisegundos como
valor en w
top2        movlwd'110'
            movwftiempo1

top1        nop
            nop
            nop
            nop
            nop
            decfsztiempo1, 1
            goto  top1
            decfsztiempo2, 1
            goto  top2
            return

demora1     movwftiempo2          ;demora tantos milisegundos como
valor en w
top3        movlwd'1110'
            movwftiempo1

top4        nop
            nop
            nop
            nop
            nop

```

```

                                decfsz tiempo1, 1
                                goto top4
                                decfsz tiempo2, 1
                                goto top3
                                return

retardo    movlw .15
           movwf conta1

dev        movlw .70
           movwf conta2

dos        movlw .20
           movwf conta3

tres       decfsz conta3, 1
           goto tres
           decfsz conta2, 1
           goto dos
           decfsz conta1, 1
           goto dev
           return

lcd        call control
           movlw .1
           call demora
           return

decimal    movwf uni           ;convierte el dato presente en w en uni, dec y cen
           clrf dec
           clrf cen
           movlw d'100'       ;determina la cant. de centenas
centena    subwf uni, 1
           btfss status, 0
           goto cien
           incf cen, 1
           goto centena
cien       addwf uni, 1
           movlw d'10'        ;determina la cant. de decenas
decena     subwf uni, 1
           btfss status, 0
           goto diez
           incf dec, 1
           goto decena
diez       addwf uni, 1        ;uni queda con la cant. de unidades (sin decenas ni cen-
tenas)

```

```

        movlw 0x30          ;le suma 30h a los valores de uni, dec y cen para que
queden en ascii
        addwf uni, 1
        addwf dec, 1
        addwf cen, 1
        return

```

```

sol1    btfsc porta,5      ;enter
        goto sol1
        return

```

```

ver     movf time, 0
        call decimal
        movlw 0x9c
        call control
        movlw .1
        call demora
        movf dec, 0
        call dato
        movf uni, 0
        call dato
        return

```

TIEMPO DE SESIÓN

Para cuadrar el tiempo

```

cuadra2    clrf  buffer2
           clrf  buffer8
cuadra3    clrf  buffer3
           clrf  buffer9
cuadra4    clrf  buffer4
           clrf  buffer10
cuadra5    clrf  buffer5
           clrf  buffer11
cuadra6    clrf  buffer6
           clrf  buffer12

cuadra     movf  time, 0
           xorlw .0
           btfsc status,2

```

```

goto tri
movf time, 0
xorlw .1
btfsc status,2
goto cencillo
movf time,0
xorlw .2
btfsc status,2
goto doble
movf time,0
xorlw .3
btfsc status,2
goto triple
movf time,0
xorlw .4
btfsc status,2
goto cuarto
movf time,0
xorlw .5
btfsc status,2
goto quinto
xorlw .6
btfsc status,2
goto sexto

btfsc portc,1
goto visualizar

cencillo2    movlw .02
              movwf tempo20
reviso20     movlw .02
              movwf tempo21
revisa20     movlw .100
              call demora
              decfsz tempo21,1
              goto revisa20
              decfsz tempo20,1
              goto reviso20
              return

cencillo     movlw .01
              movwf tempo4
reviso12     movlw .01
              movwf tempo6
revisa12     movlw .100

```

```

        call demora
        decfsztempo6,1
        goto revisa12
        decfsztempo4,1
        goto reviso12
        return

doble   movlw .20
        movwftempo1
reviso13 movlw .10
        movwftempo2
revisa13 movlw .100
        call demora
        decfsztempo2,1
        goto revisa13
        decfsztempo1,1
        goto reviso13
        return

triple  movlw .15
        movwftempo7
reviso14 movlw .20
        movwftempo8
revisa14 movlw .100
        call demora
        decfsztempo8,1
        goto revisa14
        decfsztempo7,1
        goto reviso14
        return

cuarto  movlw .40
        movwftempo9
reviso15 movlw .10
        movwftempo10
revisa15 movlw .100
        call demora
        decfsztempo10,1
        goto revisa15
        decfsztempo9,1
        goto reviso15
        return

```

```

quinto      movlw .50
            movwf tempo11
reviso16    movlw .10
            movwf tempo12
revisa16    movlw .100
            call demora
            decfsz tempo12,1
            goto revisa16
            decfsz tempo11,1
            goto reviso16
            return

```

```

sexto       movlw .30
            movwf tempo13
reviso17    movlw .20
            movwf tempo14
revisa17    movlw .100
            call demora
            decfsz tempo14,1
            goto revisa17
            decfsz tempo13,1
            goto reviso17
            return

```

```

septimo     movlw .60
            movwf tempo4
reviso18    movlw .10
            movwf tempo6
revisa18    movlw .100
            call demora
            decfsz tempo6,1
            goto revisa18
            decfsz tempo4,1
            goto reviso18
            return

```

5. CONCLUSIONES

- Se estudió el comportamiento de los signos vitales mencionados y con ello se obtuvieron las señales deseadas para su respectivo proceso digital.
- En la obtención de la frecuencia de los pulsos del corazón se usó un electrocardiógrafo el cual dio la señal en forma análoga y la frecuencia de la medida depende de la persona a la cual se le ponga el electrocardiógrafo, de dicha frecuencia se proceso para tener el valor que el dispositivo ahora nos muestra de una forma digital, la señal análoga que fue mostrada en el osciloscopio, paso por una etapa a la cual llamamos frecuenciador, para que el microcontrolador pudiese leer una señal tan veloz.
- Mediante el uso ensamblador del assembler (programa utilizado para diseñar y programar el microcontrolador) se hizo el código de línea para el proceso completo del dispositivo el cual está fundamentado en: etapa sensorica, amplificación, proceso de archivo y decodificación de datos medidos. Una vez programado el microcontrolador se configuró el circuito externo y se armó la parte externa del dispositivo (tapa).

6. RECOMENDACIONES

Al dispositivo se le puede aplicar tecnología inalámbrica con el fin de tener un mejor acceso al archivo de datos de cada paciente, desde cualquier sitio médico donde se maneje el dispositivo, esto con el fin de ahorrar tiempo y brindar una mejor atención a los pacientes.

También se le puede implementar formato de audio el cual prestará el servicio de almacenamiento por voz. El médico tendrá la posibilidad de llevar la historia clínica de cada paciente, para evitar que pasen desapercibidos datos importantes que son necesarios para la pronta recuperación del mismo.

Se le puede adaptar un sistema de memoria extraíble por medio de puerto USB o de memoria SD, con el fin de llevar los datos obtenidos en el dispositivo al medio donde por último se almacenen.

BIBLIOGRAFÍA

JUAN CARLOS ÁLVAREZ ANTÓN. Instrumentación electrónica, da edición, año 2006.

PALACIOS ENRIQUE RAMIRO. Fernando, Lucas J, López, Microcontrolador PIC 16 f84, 2 da edición; alfa omega S.A, 2006.

PALLÁS ARENY RAMÓN. Adquisición y distribución de señales, ediciones técnicas)

Biomedical Transducers And Instruments
tatsuo togawa (1997)
pake oberg

WEBLIOGRAFÍA

http://www.cibernetia.com/tesis_es.com,

<http://www.datasheet.ad620.com/>.

<http://www.med-ia.ch/medcalc/>.

<http://www.todalaley.com/mostrarLey188p1tn.htm>

http://www.unodc.org/unodc/es/legal_library/cu/legal_library_1988-04-14_1985-91.html

GLOSARIO

Modelo: conjunto de ecuaciones matemáticas que caracteriza el comportamiento de un sistema.

Sensor: Es un dispositivo que detecta manifestaciones de cualidades o fenómenos físicos, como la energía, velocidad, aceleración, tamaño, cantidad, etc. Se puede decir también que es un dispositivo que aprovecha una de sus propiedades con el fin de adaptar la señal que mide para que la pueda interpretar otro elemento. Como por ejemplo el termómetro de mercurio que aprovecha la propiedad que posee el mercurio de dilatarse o contraerse por la acción de la temperatura.

Programación: Comúnmente a la Programación de Computadoras se le suele llamar simplemente Programación y consiste en la creación de un programa de computadora (Código máquina compilado e interpretado directamente desde el núcleo del sistema) o desde un script (Código fuente interpretado), un conjunto concreto de instrucciones que una computadora puede ejecutar. El programa se escribe en un lenguaje de programación, aunque también se pueda escribir directamente en lenguaje de máquina, con cierta dificultad. Un programa se puede dividir en diversas partes, que pueden estar escritas en lenguajes distintos.

Instrumentación electrónica: es parte de la electrónica, principalmente analógica, que se encarga del diseño y manejo de los aparatos electrónicos, sobre todo para su uso en mediciones. La instrumentación electrónica se aplica en el sensado y procesamiento de la información proveniente de variables físicas y químicas, a partir de las cuales realiza el monitoreo y control de procesos, empleando dispositivos y tecnologías electrónicas.

microcontrolador: es un circuito integrado o chip que incluye en su interior las tres unidades funcionales de una computadora: CPU, Memoria y Unidades de E/S, es decir, se trata de un computador completo en un solo circuito integrado.

EEPROM: son las siglas de electrically-erasable programmable read-only memory (ROM programable y borrable eléctricamente), en español o castellano se suele referir al hablar como E²PROM y en inglés "E-Squared-PROM". Es un tipo de memoria ROM que puede ser programado, borrado y reprogramado eléctricamente, a diferencia de la EPROM que ha de borrarse mediante rayos ultravioletas. Aunque

una EEPROM puede ser leída un número ilimitado de veces, sólo puede ser borrada y reprogramada entre 100.000 y 1.000.000 de veces.

LCD: (acrónimo del inglés Liquid crystal display) es una pantalla delgada y plana formada por un número de píxeles en color o monocromos colocados delante de una fuente de luz o reflectora. A menudo se utiliza en pilas, dispositivos electrónicos, ya que utiliza cantidades muy pequeñas de energía eléctrica.

Sistema binario: El sistema binario desempeña un importante papel en la tecnología de los ordenadores. Los primeros 20 números en el sistema en base 2 son 1, 10, 11, 100, 101, 110, 111, 1000, 1001, 1010, 1011, 1100, 1101, 1110, 1111, 10000, 10001, 10010, 10011 y 10100. Cualquier número se puede representar en el sistema binario, como suma de varias potencias de dos.

Las operaciones aritméticas con números en base 2 son muy sencillas. Las reglas básicas son: $1 + 1 = 10$ y $1 \times 1 = 1$. El cero cumple las mismas propiedades que en el sistema decimal: $1 \times 0 = 0$ y $1 + 0 = 1$. La adición, sustracción y multiplicación se realizan de manera similar a las del sistema decimal:

Puesto que sólo se necesitan dos dígitos (o bits), el sistema binario se utiliza en los ordenadores o computadoras. Un número binario cualquiera se puede representar, por ejemplo, con las distintas posiciones de una serie de interruptores. La posición "encendido" corresponde al 1, y "apagado" al 0. Además de interruptores, también se pueden utilizar puntos imantados en una cinta magnética o disco: un punto imantado representa al dígito 1, y la ausencia de un punto imantado es el dígito 0. Los visitables —dispositivos electrónicos con sólo dos posibles valores de voltaje a la salida y que pueden saltar de un estado al otro mediante una señal externa, también se pueden utilizar para representar números binarios. Los circuitos lógicos realizan operaciones con números en base 2. La conversión de números decimales a binarios para hacer cálculos, y de números binarios a decimales para su presentación, se realizan electrónicamente.