

UNIVERSIDAD DEL VALLE DE GUATEMALA
Facultad de Ingeniería



Excelencia que trasciende

DEL VALLE
GRUPO EDUCATIVO

Megaproyecto Konócete
Red Sensorial de Signos Vitales

Trabajo de Graduación presentado por Clara Lucía Cruz Del Valle para optar al grado académico de Licenciado en Ingeniería en Ciencias de la Computación, Luis Alejandro Juárez Payes y Edwing Isaac Mejía Orozco para optar al grado académico de Licenciado en Ingeniería Mecatrónica, y Juana Marlenne Rivera Marroquín para optar al grado académico de Licenciado en Ingeniería Electrónica.

Guatemala
2011

Konócete
Red Sensorial de Signos Vitales

UNIVERSIDAD DEL VALLE DE GUATEMALA
Facultad de Ingeniería



Excelencia que trasciende

DEL VALLE
GRUPO EDUCATIVO

Megaproyecto Konócete
Red Sensorial de Signos Vitales

Trabajo de Graduación presentado por Clara Lucía Cruz Del Valle para optar al grado académico de Licenciado en Ingeniería en Ciencias de la Computación, Luis Alejandro Juárez Payes y Edwing Isaac Mejía Orozco para optar al grado académico de Licenciado en Ingeniería Mecatrónica, y Juana Marlenne Rivera Marroquín para optar al grado académico de Licenciado en Ingeniería Electrónica.

Guatemala
2011

Vo. Bo.:

(f) _____

M. Sc. Carlos Esquit

Tribunal:

(f) _____

M. Sc. Carlos Esquit

(f) _____

Ing. Luis Reina

(f) _____

Ing. Roberto Delgado

Fecha de Aprobación:

24 de noviembre de 2011

PREFACIO

La idea nació originalmente de un trabajo en conjunto entre la Universidad del Valle de Guatemala y la Universidad Politécnica de Madrid. El proyecto original proponía diseñar e implementar un bastón asistencial con sensores de signos vitales para monitorear el estado de salud del paciente y asistirlo a caminar.

Este proyecto busca incursionar en Guatemala en el campo de la telemedicina, un país en el que las facilidades de acceso a lugares remotos están limitadas, dando como resultado un servicio de salud de baja calidad. Con este proyecto esperamos proporcionar una herramienta que brinde una mejor calidad de vida, facilitando la medición de signos vitales en comunidades de difícil acceso y pacientes que deben ser monitoreados constantemente.

Queremos agradecer a todas las personas que nos apoyaron durante el proceso de experimentación. También queremos agradecer a nuestros asesores Ing. Marie André Destarac, MSc. Carlos Esquit e Ing. Luis Fernando Reina por su tiempo y apoyo constante.

TABLA DE CONTENIDO

PREFACIO	vi
TABLA DE CONTENIDO	vii
LISTA DE GRÁFICOS	ix
LISTA DE CUADROS.....	xii
RESUMEN	xiii
I. INTRODUCCIÓN.....	1
II. OBJETIVOS	4
A. Objetivo general del proyecto	4
B. Objetivo específicos del proyecto	4
III. MARCO TEÓRICO	5
A. Tecnología Bluetooth.....	5
B. Parámetros de medición de signos vitales	7
C. Plataforma Android	11
IV. ANTECEDENTES	12
V. DELIMITACIÓN E IMPACTO DEL TEMA.....	13
VI. METODOLOGÍA.....	14
VII. RESULTADOS SISTEMA DE MEDICIÓN DE PESO	15
VIII. RESULTADOS SISTEMA DE MEDICIÓN DE FRECUENCIA RESPIRATORIA.....	16
IX. RESULTADOS SISTEMA DE MEDICIÓN DE GLUCOSA	17
X. RESULTADOS SISTEMA DE MEDICIÓN DE PRESIÓN ARTERIAL.....	18
XI. RESULTADOS SISTEMA DE MEDICIÓN DE TEMPERATURA.....	20
XII. RESULTADOS APLICACIÓN TELÉFONO CELULAR.....	21
XIII. RESULTADOS ADQUISICIÓN DE DATOS.....	23
XIV. DISCUSIÓN	24
XV. CONCLUSIONES Y RECOMENDACIONES	28
A. Conclusiones.....	28
B. Recomendaciones	28
XVI. BIBLIOGRAFÍA.....	30
XVII. APÉNDICE A: Módulo Central de Información	34
A. Metodología	34
B. Comunicación entre dos módulos Bluetooth (RBT-001).....	34
C. Uso del modo comando para el programa maestro.....	37
D. Comunicación entre el teléfono celular y dos módulos Bluetooth.....	43
E. Implementación final de la red sensorial.....	48

F.	Discusión.....	57
XVIII.	APÉNDICE B: Módulo de Monitoreo y comunicación.....	61
A.	Metodología	61
B.	Aplicación J2ME	63
	1. Diseño	63
	2. Funcionalidad Bluetooth.....	67
	3. Discusión..	69
C.	Aplicación Android.....	70
	1. Diseño	70
	2. Resultados.....	75
	3. Almacenamiento de datos tarjeta micro SD.....	77
	4. Comparación de mediciones de signos vitales.	77
D.	Discusión.....	78
XIX.	APÉNDICE C. Sistema de Medición de Temperatura, Glucosa	
	y Presión Arterial.....	81
A.	Metodología general	81
B.	Esfigmomanómetro.....	82
	1. Diseño experimental.....	82
	2. Resultados.....	83
C.	Glucómetro.....	91
	1. Diseño experimental.....	91
	2. Resultados.....	91
D.	Termómetro.....	98
	1. Diseño experimental.....	98
	2. Resultados.....	99
E.	Discusión.....	103
XX.	APÉNDICE D. Sistema de Medición de Frecuencia Respiratoria	
	y Peso.....	108
A.	Metodología general	108
B.	Medición de peso.....	109
	1. Diseño experimental.....	109
	2. Resultados.....	113
	3. Discusión..	115
C.	Medición de frecuencia respiratoria.....	118
	1. Diseño experimental.....	118
	2. Resultados.....	123
	3. Discusión.	129
XXI.	GLOSARIO.....	132

LISTA DE GRÁFICOS

Figura 1 Módulos que conforman el proyecto Konócete.....	2
Figura2 Rangos de presión sanguínea para adultos(Department Of Health and Human Services, 2004)	8
Figura3 Rangos Normales de temperatura corporal	9
Figura 4 Categorías de peso con base a BMI(Prevention, 2011).....	10
Figura 5 Topología de red implementada para la comunicación con glucómetro	17
Figura 6 Adquisición de la medición realizada por el glucómetro	17
Figura 7Interfáz aplicación Android	21
Figura 8: Diagrama de flujo del programa maestro.	36
Figura 9: Diagrama de flujo del programa esclavo.	37
Figura 10: Detalles del comando SPP_SEND_DATA.....	39
Figura 11: Especificaciones del comando SPP_INCOMING_DATA	39
Figura 12: Descripción de las especificaciones del comando SET_EVENT_FILTER.	40
Figura 13 Extracto que muestra las características del maestro Piconet y	41
Figura 14: Diagrama de flujo del programa Maestro.....	42
Figura 15: Diagrama de flujo del programa esclavo.....	42
Figura 16: Detalle de la instrucción SET_PORTS_TO_OPEN.....	43
Figura 17: Extracto de la tabla de localidades NVS.....	44
Figura 18 Resultados de la lectura de la memoria NVS del módulo maestro.	44
Figura 19 Conexión, envío y recepción de datos entre el módulo maestro, el esclavo y el teléfono celular.....	46
Figura 20: Dispositivos encontrados por la aplicación Java del teléfono celular.	47
Figura 21: Datos recibidos por la aplicación Java del teléfono celular vía Bluetooth.	47
Figura 22: Dato recibido por el módulo esclavo vía Bluetooth del módulo maestro.	48
Figura 23: Topología de la red de comunicación.....	49
Figura 24: Algoritmo del programa del microcontrolador maestro.....	50
Figura 25: Circuito del Módulo de Monitoreo.....	51
Figura 26: Mensaje desplegado al ejecutar la instrucción SET_EVENT_FILTER.....	52
Figura 27: Mensaje mostrado cuando el maestro espera la conexión del teléfono celular.....	52
Figura 28: Mensaje mostrado al iniciar la conexión utilizando la instrucción SPP_ESTABLISH_LINK.	52
Figura 29 Mensajes y códigos de error que pueden ocurrir	53
Figura 30 Extracto del funcionamiento del RBT-001 en modo automático.	55
Figura 31 Extracto del funcionamiento del RBT-001 en modo No-Automático.	56

Figura 32: Especificaciones del comando WRITE_OPERATION_MODE	56
Figura 33 Diagrama de bloques Konócte módulo monitoreo y comunicación	62
Figura 34 Diagrama de flujo CommandAction en J2ME	65
Figura 35 Diagrama de flujo Bluetooth en J2ME	65
Figura 36 Diagrama de flujo funcionalidad SMS en J2ME.....	66
Figura 37 Interfaz de usuario de la aplicación ejecutándose en el simulador J2ME	67
Figura 38 Conexión Bluetooth aplicación J2ME.....	67
Figura 39 Envío de datos a módulo central de monitoreo aplicación J2ME	68
Figura 41 Diagrama de flujo configuración Bluetooth Android	70
Figura 42 Diagrama de flujo búsqueda de dispositivos Bluetooth Android	71
Figura 43 Diagrama de flujo conexión Bluetooth con Módulo Central de Información	71
Figura 44 Diagrama de flujo Envío de mensajes.....	72
Figura 45 Diagrama de flujo almacenamiento de mediciones en tarjeta micro SD	73
Figura 46 Diagrama de flujo recepción de mediciones en Módulo Central de Monitoreo.....	74
Figura 47 Interfaz de usuario de aplicación Android ejecutada en emulador	75
Figura 48 Aplicación Android Bluetooth funcionando en teléfono celular	76
Figura 49 Envío de mensajes de texto funcionando en simulador Android	77
Figura 50 Pruebas aplicación Android en diferentes modelos	80
Figura 51: Intervención de memoria HT24LC16 del esfigmomanómetro.	83
Figura 52: Diagrama de memoria HT24LC16.....	83
Figura 53: Secuencia para acceso aleatorio de la memoria HT24LC16.....	84
Figura 54: Señal en pin de alimentación en memoria HT24LC16 en modo de espera del esfigmomanómetro.	85
Figura 55: Señal en pin WP de la memoria HT24LC16 en estado activo del esfigmomanómetro.....	85
Figura 56: Señal en pin WP en memoria HT24LC16 durante almacenamiento de lectura.	86
Figura 57 Datos adquiridos de memoria HT24LC16	87
Figura 58: Circuito implementado para detección de condición de almacenamiento en memoria del esfigmomanómetro y lectura de última medición realizada.	88
Figura 59: Presión sistólica, diastólica y pulso cardíaco adquiridos.	89
Figura 60: Medición almacenada en memoria de esfigmomanómetro ADC 6015	89
Figura 61: Medición de exactitud en medición de presión sistólica del esfigmomanómetro invadido.	89
Figura 62 Medición de exactitud en medición de presión diastólica del esfigmomanómetro invadido.	90
Figura 63 Medición de exactitud en la medición de pulso cardíaco del esfigmomanómetro invadido	90
Figura 64: Circuito principal glucómetro Accu-Check Aviva.....	92
Figura 65: Analizador de paquetes implementado para registro de protocolo infrarrojo del glucómetro Accu-Check Aviva.	93
Figura 66: Comparación entre ancho de pulso UART e IrDA.....	94
Figura 67: Topología para la captura del protocolo entre glucómetro y PDA.	94

Figura 68: Datos adquiridos con el analizador de paquetes infrarrojos implementado.....	95
Figura 69: Representación ASCII de datos capturados del protocolo infrarrojo implementado por glucómetro.....	95
Figura 70: Topología de circuito implementado para la comunicación infrarroja con glucómetro.....	97
Figura 71: Medición adquirida mediante comunicación infrarroja con glucómetro. ...	97
Figura 72: Última medición almacenada en glucómetro.....	98
Figura 73: Recubrimiento de termómetro digital DS18B20 en termómetro oral implementado.....	99
Figura 74: Termómetro oral implementado.	100
Figura 75: Algoritmo implementado en termómetro oral diseñado.	101
Figura 76: Temperatura adquirida con termómetro diseñado.....	101
Figura 77: Comparación entre temperaturas registradas por termómetro MT3001 y termómetro diseñado.	103
Figura 78: Efectividad de mediciones de temperatura.....	103
Figura 79: División del Sistema de Medición de Frecuencia Respiratoria y Peso.	109
Figura 80: Tarjeta interna y sensor de carga de la balanza.....	110
Figura 81: Conexión de celdas de carga dentro de la balanza.....	110
Figura 82: Circuito de implementación de los sensores de la balanza.	111
Figura 83 Algoritmo para medición de peso.....	112
Figura 84 Resultado obtenido con base en el peso de referencia en libras.....	113
Figura 85: Resultados de la caracterización de la balanza.....	114
Figura 86: Errores cuadrados de la caracterización de la balanza.....	114
Figura 87: Modelo de cinturón en inhalación y exhalación.....	119
Figura 88: Cinturón para medición de frecuencia respiratoria.....	119
Figura 89: Segunda versión del cinturón para medición de frecuencia respiratoria.....	119
Figura 90: Usuario utilizando la segunda versión del cinturón.....	120
Figura 91: Circuito implementado para la medición de frecuencia respiratoria.....	120
Figura 92: Algoritmo para medir frecuencia respiratoria sección A.....	121
Figura 93 Algoritmo para medición de frecuencia respiratoria sección B.....	122
Figura 94 Algoritmo para medición de Frecuencia Respiratoria sección C.....	123
Figura 95: Mediciones de la primera versión del cinturón tomadas cada 400ms sin aplicar filtros.....	124
Figura96 Mediciones con alteraciones en la primera versión del cinturón.....	124
Figura 97: Promedio y desviación para estimar la barrera de medición.....	125
Figura 98 Detección de cambios de inspiración y espiración en un usuario.....	125
Figura 99 Resultados de medición de frecuencia respiratoria en distintos usuarios.....	126
Figura 100: Porcentajes de mediciones respecto a la desviación de 1 o 2 rpm.....	127
Figura 101: Mediciones realizadas con el cinturón completo.....	128

LISTA DE CUADROS

Cuadro 1 Medición de peso corporal con la balanza caracterizada	15
Cuadro 2 Mediciones de frecuencia respiratoria con el Cinturón	16
Cuadro 3 Medición de exactitud en la medición de presión sistólica del esfigmomanómetro invadido	18
Cuadro 4 Medición de exactitud en la medición de presión diastólica del esfigmomanómetro invadido	18
Cuadro 5 Medición de exactitud en la medición de pulso cardíaco del esfigmomanómetro invadido	19
Cuadro 6 Medición de exactitud del termómetro diseñado	20
Cuadro 7 Pruebas aplicación teléfono celular	22
Cuadro 8 Tiempo de adquisición de medición por sensor	23
Cuadro 9 Formato de datos que maneja cada sensor	23

RESUMEN

Este proyecto corresponde a la implementación de un sistema remoto de monitoreo de signos vitales en pacientes mayores a 18 años, en la cual, el personal encargado del paciente, puede recibir a través de un mensaje de texto telefónico la información concerniente a su salud. De esta forma, los signos vitales que la red de monitoreo Konócte es capaz de registrar son: Presión arterial sistólica y diastólica, pulso cardíaco, frecuencia respiratoria, concentración de glucosa en la sangre, temperatura y peso corporal.

El proyecto se dividió en cuatro módulos, que son: dos módulos encargados de implementar los sensores de signos vitales, un módulo de procesamiento, y un módulo encargado de la aplicación del celular y el envío de mensajes de texto.

Encontrando que se logró implementar una red de sensores y una plataforma de comunicación la cual que permite evaluar el estado general de salud del paciente, tomando todos los signos vitales incluidos en el equipo en un tiempo menor a 5 minutos.

I. INTRODUCCIÓN

Este proyecto busca incursionar en Guatemala en el campo de la telemedicina, un concepto que implica el intercambio de información de manera electrónica, de tal manera que se mejore el control de la salud del paciente. Dentro de los servicios de la telemedicina se incluyen las consultas de pacientes de manera virtual, monitoreo vital de manera remota y educación médica en áreas remotas.

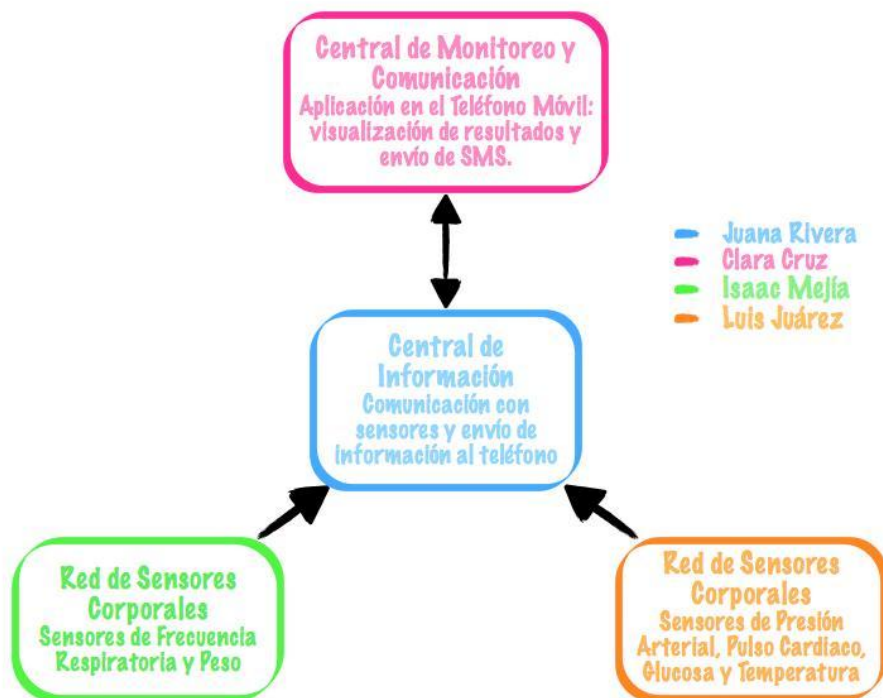
La telemetría ha formado una rama completa de investigación alrededor del mundo. Varios países están trabajando en nuevos proyectos a manera de obtener avances útiles para la comunidad. En Guatemala, esta rama de la medicina e ingeniería es una posible solución para facilitar la atención médica a lugares de difícil acceso, así como a pacientes que se desea monitorear constantemente.

El proyecto Konócete tiene como objetivo principal la implementación de una red remota de monitoreo de signos vitales en pacientes mayores a los 18 años, de tal manera que la información adquirida correspondiente a la salud del paciente sea enviada por medio de mensaje de texto al personal encargado del paciente. Dentro de los signos vitales que esta red es capaz de monitorear se encuentra la presión arterial sistólica y diastólica, pulso cardíaco, temperatura corporal, frecuencia respiratoria, concentración de glucosa en la sangre y peso corporal. El proyecto se dividió en cuatro módulos (ver Figura 1) que son:

- Módulo Central de Información: basado en la tecnología Bluetooth y sus ventajas para la creación de redes locales pequeñas.
- Módulo de Monitoreo y comunicación: Basado en la investigación de programación Bluetooth y envío de mensajes SMS en plataforma J2ME y Android así como almacenamiento de información en tarjeta SD en plataforma Android.
- Sistema de medición de glucosa, presión arterial, pulso y temperatura corporal: basado en protocolos digitales de comunicación y sensores digitales.

- Sistema de medición de frecuencia respiratoria y peso corporal: basado en circuitos analógicos utilizando galgas extensiométricas y sensores de carga así como el procesamiento de estas señales analógicas.

Figura 1 Módulos que conforman el proyecto Konócete



Para la implementación de este proyecto, se garantizó el funcionamiento de cada módulo individualmente para luego incorporar la comunicación inalámbrica a cada uno de los sensores, de tal manera que se estableció comunicación con la Central de Información. Finalmente, la Central de Información se conectó con la unidad central de monitoreo y comunicación, logrando que los signos vitales recolectados fueran enviados por mensaje de texto al dispositivo móvil del responsable de estado de salud del paciente.

Las mediciones son válidas debido a que se cuenta con una desviación respecto a los sensores comerciales no mayor de 4.9lb para el peso, 3 respiraciones por minuto para la frecuencia respiratoria, 4 mmHg para la presión arterial, 6 latidos por minuto para el pulso cardiaco y 0.4°C para la temperatura. Este valor es suficientemente pequeño como para validar el funcionamiento de los sensores utilizados en Konócete.

Con los resultados de las pruebas de la aplicación del teléfono celular se llegó a la conclusión que esta solo se puede ejecutar en dispositivos Sony Xperia. Esto se debe a la funcionalidad Bluetooth la cual tiene especificaciones diferentes para cada marca.

Al final se logró presentar un sistema que mide presión sistólica, presión diastólica, pulso cardiaco, concentración de glucosa en la sangre, temperatura, frecuencia respiratoria y peso en un paciente en menos de 5 minutos.

II. OBJETIVOS

A. **Objetivo general del proyecto**

Diseñar una red de sensores y una plataforma de comunicación que permitan evaluar el estado general de salud del paciente y transmitan información específica del paciente a familiares, cuidadores y/o médicos por medio de mensajes de texto a teléfonos celulares.

B. **Objetivo específicos del proyecto**

1. Diseñar e implementar los circuitos capaces de obtener las mediciones realizadas por los sensores utilizados para peso corporal y frecuencia respiratoria del usuario. Incluyendo la comunicación periférica dedicada al envío adecuado de las mediciones al módulo central de monitoreo.
2. Diseñar e implementar un sistema que obtenga información de la red de sensores, la coloque en un formato entendible por el teléfono celular y se la envíe al móvil.
3. Diseñar y construir los circuitos encargados de extraer las mediciones realizadas por los instrumentos de medición de concentración de glucosa en la sangre, presión arterial y temperatura corporal.
4. Diseñar una aplicación para un teléfono celular que se encargue de la comunicación entre el paciente y el médico.

III. MARCO TEÓRICO

A. Tecnología Bluetooth

Para la implementación de la red completa fue necesario basarse en la tecnología Bluetooth para más detalles referirse a Konócete red sensorial de signos vitales: Central de Información (Rivera, 2011) y Konócete red sensorial de signos vitales: central de monitoreo y comunicación (Cruz, 2011).

La tecnología Bluetooth fue creada por la compañía Bluetooth SpecialInterestGroup (SIG) quienes se encargan de administrar todo lo relacionado con esta tecnología como lo es el desarrollo y actualización del mismo. Esta compañía no fabrica los productos que emplean esta tecnología sino que se encargan de publicar las especificaciones de la tecnología, administrar el programa de acreditación y proteger las patentes de la marca para que esté disponible a todos. A continuación se muestran las características más importantes de esta tecnología que fueron empleadas en el desarrollo de este trabajo. (SIG, 2011)

1. Redes Bluetooth Piconet y saltos de frecuencia. Una Piconet es una red ad-hoc que une a un grupo de dispositivos usando el canal físico del Bluetooth y consiste en un maestro y uno o varios esclavos. El maestro es el dispositivo que proporciona el reloj para que los esclavos se sincronicen y además proporciona el patrón del cambio de frecuencias. También es importante mencionar que el dispositivo maestro nunca solicita la conexión, este solo espera a que dispositivos se quieran conectar a él, y es el esclavo quien busca los dispositivos cercanos y se conecta con el maestro (SIG, 2011).

Una vez se haya establecido la conexión, los dispositivos utilizarán un patrón de salto de frecuencias que se calcula utilizando un algoritmo que utiliza la dirección Bluetooth del dispositivo y el reloj del maestro. El salto de frecuencias termina siendo 79 frecuencias escogidas en un proceso cuasi-aleatorio en el rango de la banda ISM.

Este mismo patrón puede ser modificado por el dispositivo para excluir frecuencias que estén causando interferencia debido a otros dispositivos que utilizan esta banda pero no utilizan el algoritmo de salto de frecuencia (SIG, 2011).

2. Transmisión de datos Full-Duplex. Durante la conexión entre dos o más dispositivos el canal físico se divide en unidades de tiempo o espacios para la transmisión de datos. Los datos se transmiten en paquetes que se envían dentro de estos espacios de tiempo. Cuando los paquetes de datos son más grandes y si las circunstancias lo permiten, un número de estos espacios de tiempo pueden ser asignados de manera contigua para que formen un solo paquete. Es importante notar que el salto de frecuencias ocurre únicamente entre la transmisión y recepción de datos, nunca ocurre durante una transmisión. Además, la tecnología Bluetooth utiliza un esquema de división de tiempo dúplex para tener una comunicación full dúplex(SIG, 2011).

3. Módulo Robotech Bluetooth serial RBT-001. Además de ser necesario comprender las especificaciones de la tecnología Bluetooth, fue muy importante estudiar el funcionamiento del dispositivo RBT-001 que es el que se utilizará en la Central de Información para comunicar el teléfono celular con el microcontrolador que contendrá la información de los sensores. Este dispositivo es compatible con las especificaciones 1.x y 2.0 de Bluetooth y es un dispositivo de bajo consumo de potencia. Al cumplir con las especificaciones de Bluetooth, el dispositivo soporta AFH (Salto de Frecuencias Adaptables por sus siglas en inglés) y los perfiles SDAP y SPP de Bluetooth. El perfil SDAP –ServiceDiscoveryApplicationProfile describe cómo un dispositivo debe buscar los servicios de otro dispositivo Bluetooth para determinar de qué manera pueden comunicarse. El perfil SPP define cómo configurar puertos seriales para la transferencia de datos (RoboTech, 2010).

4. Descripción de bluetooth en Android. En la plataforma Android el paquete android.Bluetooth provee las clases que manejan la funcionalidad. Este paquete permite al usuario, configurar el Bluetooth, encontrar dispositivos disponibles,

conectar dispositivos y transferir datos entre dispositivos. Es necesario utilizar las clases:

- a. `BluetoothAdapter`: Clase principal para interactuar con el servicio Bluetooth.
- b. `BluetoothDevice`: Representa un dispositivo Bluetooth encontrado. Se utiliza para iniciar una conexión por medio de un socket.
- c. `BluetoothSocket`: Ésta es la conexión que permite que la aplicación intercambie datos con otros dispositivos. Para esto se utiliza `InputStream` y `OutputStream`.
- d. `BluetoothServerSocket`: Ésta es la conexión que está esperando solicitudes.
- e. `BluetoothClass`: Describe características generales y capacidades de un dispositivo Bluetooth. Contiene las propiedades que definen a un dispositivo y a sus servicios. (Developers, Bluetooth | AndroidDevelopers, 2011) (Developers, Using Hardware Devices | AndroidDevelopers, 2011)

Para que se puedan compartir datos vía Bluetooth entre el teléfono celular y el Módulo de Monitoreo estos deben de estar conectados a un Socket en el mismo canal RFCOMM. En esta aplicación se va a utilizar un socket tipo RFCOMM, pues este es el tipo que soporta el API de Android. Cuando ya están conectados, cada dispositivo puede obtener un input y un output stream con los cuales se van a transferir los datos. (Developers, Bluetooth | AndroidDevelopers, 2011)

B. Parámetros de medición de signos vitales

Para definir si las mediciones que se obtiene del módulo Central de Información se encuentran dentro de los parámetros normales es necesario conocer los rangos de los parámetros normales de las mediciones de signos vitales. Para más detalles referirse a *Konócete red sensorial de signos vitales: Sistema de medición de temperatura, glucosa y presión arterial* (Juárez, 2011) y *Konócete red sensorial de signos vitales: Sistema de medición de peso y frecuencia respiratoria*. (Mejía, 2011)

1. Presión. La presión arterial es un signo vital importante. Está expresada como presión sistólica/diastólica. La presión sistólica contiene el número mayor y mide la presión en las arterias cuando el corazón se contrae. La presión Diastólica es el número menor y mide la presión en las arterias cuando el músculo del corazón antes y después de contraerse. (Pusajo, Egurrola, & Hernandez, 2011)

Como se puede ver en la Figura 2, se considera 120/80 como presión sanguínea normal, sin embargo estos valores pueden variar. Los límites normales de presión arterial sistólica para una persona adulta saludable se encuentran entre 90mmHg y 120mmHg. Los límites normales de presión arterial diastólica se encuentran entre 60mmHg y 80mmHg. Cualquier valor que se encuentra fuera de estos rangos se considera anormal. (Disabled World, 2008)

Se llama hipertensión arterial al aumento de la presión arterial sistólica o diastólica. Se llama hipotensión arterial al descenso de la presión arterial. (Pusajo, Egurrola, & Hernandez, 2011) (Disabled World, 2008) (Marcano, 2011)

Figura 2 Rangos de presión sanguínea para adultos(Department Of Health and Human Services, 2004)

Table 3. Classification of blood pressure for adults

BLOOD PRESSURE CLASSIFICATION	SBP MMHg	DBP MMHg
NORMAL	<120	and <80
PREHYPERTENSION	120–139	or 80–89
STAGE 1 HYPERTENSION	140–159	or 90–99
STAGE 2 HYPERTENSION	≥160	or ≥100

SBP, systolic blood pressure; DBP, diastolic blood pressure

2. Temperatura. La temperatura normal del cuerpo para un adulto se encuentra alrededor de 37°C, en condiciones normales. Estos valores pueden variar por varios factores como el lugar del cuerpo en el que se realiza la medición. Como se puede ver en la Figura 3, el rango de temperatura oral normal para un individuo varía entre 36.44°C y 37.55°C. Cuando la temperatura está por encima de los límites normales, se considera como fiebre. (Healthcare Online, 2011) (Elert, 2005)

Figura 3 Rangos Normales de temperatura corporal

NORMAL BODY TEMPERATURE RANGES				
°F	0 - 2 years	3 - 10 years	11 - 65 years	> 65 years
Oral	—	95.9 - 99.5	97.6 - 99.6	96.4 - 98.5
Rectal	97.9 - 100.4	97.9 - 100.4	98.6 - 100.6	97.1 - 99.2
Axillary	94.5 - 99.1	96.6 - 98.0	95.3 - 98.4	96.0 - 97.4
Ear	97.5 - 100.4	97.0 - 100.0	96.6 - 99.7	96.4 - 99.5
Core	97.5 - 100.0	97.5 - 100.0	98.2 - 100.2	96.6 - 98.8

3. Respiración. La frecuencia respiratoria normal en un adulto saludable varía de 12 a 20 respiros por minuto. Estos valores pueden incrementar si hay fiebre o enfermedad. (Ohio StateUniversity Medical Cente, 2011) (Dhearts, 2011)

En un adulto, cualquier frecuencia respiratoria menor a 12 respiros por minuto se llama bradicardia, lo cual indica respiración lenta. Cualquier frecuencia respiratoria mayor a 25 respiros por minuto se llama taquicardia, lo cual indica respiración rápida. (Dhearts, 2011)

4. Glucosa. La medición normal de glucosa cuando la muestra se toma a lo largo del día sin importar que el usuario haya comido, se encuentra entre 135 y 140 miligramos por decilitro. (Mason Medical Center, 2011) (Healthwise, 2009)

5. Peso. Se utiliza el IMC (Índice de Masa Corporal) para evaluar el estado nutricional de una persona. Este número se calcula a partir del peso y la altura de una

persona e indica si una persona está abajo del peso normal, con sobre peso u obesidad. El índice de masa corporal se obtiene mediante la siguiente fórmula:

$$\text{peso}(kg)/[\text{altura}(m)]^2$$

Para adultos se según la Figura 4 se puede ver que un IMC menor a 18.5 indica que esta abajo del peso normal, de 18.5 a 24.9 indica que tiene un peso normal, de 25 a 29.9 indica que tiene sobre peso y mayor a 30 indica obesidad. (Prevention, 2011)

Figura 4 Categorías de peso con base a BMI(Prevention, 2011)

BMI	Weight Status
Below 18.5	Underweight
18.5 – 24.9	Normal
25.0 – 29.9	Overweight
>30.0	Obese

6. Pulso cardiaco. Para un adulto que no está realizando actividad física, el pulso cardiaco varía entre 60 a 100 latidos por minuto. Si se tiene un pulso cardiaco mayor a 100 latidos por minuto se da la condición que se llama taquicardia. Si se tiene un pulso cardiaco menor a 60 latidos por minuto se da la condición bradicardia. En ambas condiciones se debe consultar a un médico. (Laskowski, 2010) (NEMA, 2003)

C. **Plataforma Android**

Android es una plataforma para dispositivos móviles como smartphones o tablets. Fue desarrollado para permitir a los desarrolladores crear aplicaciones móviles que puedan aprovechar al máximo todas las funcionalidades de un celular. Por ejemplo, una aplicación Android puede realizar llamadas, enviar mensajes de texto, utilizar la cámara del dispositivo, etc. (Open Handset Alliance, 2011)

Esta plataforma está construida sobre el Kernel de Linux. Utiliza una máquina virtual personalizada diseñada para optimizar recursos de memoria y hardware en un dispositivo móvil. Esta plataforma continuará evolucionando a medida que la comunidad de desarrolladores trabajan juntos para crear aplicaciones móviles innovadoras. (Open Handset Alliance, 2011)

Android SDK provee las herramientas y el API necesario para desarrollar aplicaciones para la plataforma Android utilizando el lenguaje de programación Java. (Haseman, 2010)

IV. ANTECEDENTES

En los últimos años se ha visto un creciente interés en el desarrollo de sistemas de telemedicina a nivel mundial. De esta manera, en 2003 el gobierno de Estados Unidos invirtió aproximadamente \$270 millones entre seis y ocho diferentes programas de desarrollo en telemedicina. De igual manera actualmente el programa de Veterans Health Administration, que es el mayor proveedor de telemedicina en Estados Unidos, para el 2003 tenía previsto la cobertura de casi 350,000 pacientes de manera remota (American Telemedicine Association, 2011).

Por otro lado, en 2004 la NASA, en su programa Ames Astrobionics, desarrolló un dispositivo conocido como LifeWard, que tiene como objetivo la adquisición de signos vitales (temperatura, electrocardiogramas, concentración de oxígeno en la sangre, presión arterial, etc.) que luego son enviados inalámbricamente a una computadora para su visualización en tiempo real o posterior análisis. El dispositivo se caracteriza por su compacto tamaño que le permite ser portado por el usuario cómodamente (Mundt, 2004).

Por su parte, Ontario ha trabajado durante casi 10 años en la formación de una red (OTN) que se ha especializado en servicios de video conferencia en hospitales en las cuales los especialistas pueden inspeccionar pacientes de manera remota. En los últimos 3 años aproximadamente unos 7,000 pacientes han utilizado esta red, la cual se calcula que ha ahorrado a los pacientes unos \$18 millones (Smith, 2010).

A nivel latinoamericano, dentro de los avances que se han logrado en este campo de estudio se encuentra el desarrollo en 2005 por parte de la Universidad Pontificia Bolivariana de una interfaz para la visualización de signos vitales de manera remota por medio de una Red IP. Dentro de los signos vitales que este proyecto es capaz de obtener se encuentran la presión sanguínea, oxígeno en la sangre, pulso cardiaco y electrocardiogramas. Así mismo los usuarios pueden acceder a funciones del dispositivo de manera remota (Mejia, Giraldo, Castrillon, & Cardon, 2005).

V. DELIMITACIÓN E IMPACTO DEL TEMA

Este proyecto busca incursionar en Guatemala en el campo de la telemedicina, un país en el que las facilidades de acceso a lugares remotos están limitadas, dando como resultado un servicio de salud de baja calidad. Con este proyecto se espera proporcionar una herramienta que brinde una mejor calidad de vida, facilitando la medición de signos vitales en comunidades de difícil acceso y pacientes que deben ser monitoreados constantemente.

Este documento consolidado no pretende entrar en detalle de todos los módulos individuales sino detallar el proceso de integración de los módulos en la implementación general de la red sensorial Konócete. Así mismo, se exponen los resultados más relevantes que validan los objetivos propuestos para el proyecto. Para más detalles dirigirse a las fuentes. (Rivera, 2011), (Cruz, 2011), (Juárez, 2011) y (Mejía, 2011).

Cabe mencionar que con la implementación de este sistema, no se espera sustituir el equipo profesional en la medición de estos signos vitales que se encuentran en los hospitales o centros de atención médica, sino más bien dar al personal encargado de la salud del paciente un cuadro general de la salud del mismo, que les permita tomar las medidas precautorias en caso se registren valores fuera de los parámetros normales con el sistema implementado.

VI. METODOLOGÍA

Estos fueron los pasos para la adecuada integración de los módulos individuales:

- A. Garantizar el funcionamiento de cada sistema de sensores implementando el sensor con una interfaz gráfica que permitiera comparar el resultado medido contra mediciones de otro sensor comercial.
- B. Incorporar la comunicación inalámbrica Bluetooth a cada uno de los sensores. Además, enviar datos conocidos desde los sensores hasta la Central de Información para corroborar la recepción correcta y completa de los datos.
- C. Conectar la Central de Información con el Módulo de Monitoreo y comunicación utilizando la tecnología Bluetooth. Estableciendo un canal de comunicación para transferir datos entre el módulo de información y el dispositivo móvil, verificando que se mantenga la conexión durante un tiempo de 5 minutos y que los datos enviados por la Central de Información sean recibidos correctamente en la aplicación del teléfono celular.
- D. Investigar sobre rangos normales de mediciones de cada signo vital. Determinando algunas relaciones importantes que existen entre las mediciones como el Índice de Masa Corporal para diagnosticar posibles padecimientos del usuario.
- E. Implementar la comparación de mediciones en la aplicación del teléfono celular para el envío del mensaje de texto. Ésta verifica que cada medición se encuentre dentro de los límites normales, para notificar al usuario de aquellos que no lo están.
- F. Diseñar y construir contenedores ergonómicos y adecuados para los sistemas, de manera que provean portabilidad y protección al equipo.
- G. Realizar en 20 individuos la adquisición de los signos vitales de manera automática y remota con el equipo, para comprobar la validez del proyecto.

VII. RESULTADOS SISTEMA DE MEDICIÓN DE PESO

En el Cuadro 1, se puede comparar los pesos obtenidos con la balanza digital y los pesos de referencia de la balanza analógica. Encontrando que la mayor desviación es de 4.9 libras.

Cuadro 1 Medición de peso corporal con la balanza caracterizada

Usuario	Peso de referencia [libras]	Valor obtenido por el sistema [libras]	Desviación de la referencia [libras]
1	180	179.46	0.538
2	180	179.69	0.314
3	156	156.22	0.222
4	156	157.34	1.342
5	156	157.06	1.062
6	156	157.51	1.51
7	130	131.13	1.134
8	115	114.84	0.162
9	115	114.17	0.834
10	115	113.72	1.282
11	115	114.05	0.946
12	166	168.15	2.15
13	166	169.38	3.382
14	166	170.89	4.894
15	166	170.33	4.334
16	185	186.07	1.07
17	185	186.63	1.63
18	185	185.57	0.566
19	180	181.20	1.198
20	180	181.93	1.926

VIII. RESULTADOS SISTEMA DE MEDICIÓN DE FRECUENCIA RESPIRATORIA

En el Cuadro 2, se pueden comparar los resultados de respiraciones por minuto medidas por el cinturón implementado y las mediciones tomadas con el método convencional de medición.

Cuadro 2 Mediciones de frecuencia respiratoria con el Cinturón

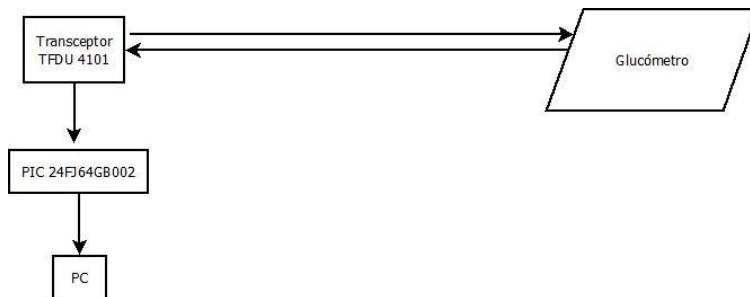
Usuario	Teórico	Medido	Desviación teórico
	[rpm]	[rpm]	[rpm] ⁱ
1	14	14	0
2	14	14	0
3	20	19	1
4	14	14	0
5	10	10	0
6	20	18	2
7	8	7	1
8	14	15	1
9	14	13	1
10	18	20	2
11	16	15	1
12	10	9	1
13	16	19	3
14	14	14	0
15	14	13	1
16	16	16	0
17	4	4	0
18	24	24	0
19	16	14	2
20	16	15	1

ⁱLa dimensional de rpm se refiere a respiraciones por minuto.

IX. RESULTADOS SISTEMA DE MEDICIÓN DE GLUCOSA

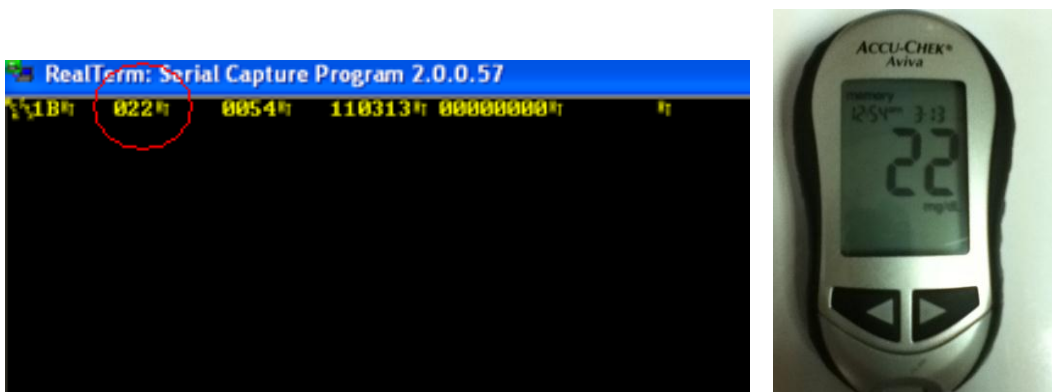
Para obtener las mediciones del glucómetro este se conecta utilizando infrarrojo con el circuito encargado de la codificación y envío de los datos al módulo Central de Información. Es por esto que la exactitud de la medición está sujeta al fabricante del glucómetro. En la Figura 5 se puede observar la topología utilizada para la comunicación entre el glucómetro y el circuito implementado.

Figura 5 Topología de red implementada para la comunicación con glucómetro



En la Figura 6 se puede observar un ejemplo de la codificación de la medición almacenada en la memoria del glucómetro.

Figura 6 Adquisición de la medición realizada por el glucómetro



X. RESULTADOS SISTEMA DE MEDICIÓN DE PRESIÓN ARTERIAL

En este sistema se realizaron solamente tres mediciones para verificar su desviación ya que el equipo adquirido fue invadido en su memoria de mediciones y no se tiene control sobre el funcionamiento completo del circuito. En otras palabras está sujeto a la exactitud del método desarrollado por el fabricante del esfigmomanómetro. Los resultados se pueden observar en los Cuadros 3, 4 y 5.

Cuadro 3 Medición de exactitud en la medición de presión sistólica del esfigmomanómetro invadido

Usuario	Presión sistólica referencia [mmHg]ⁱⁱ	Presión sistólica [mmHg]	Desviación [mmHg]
1	116	118	2
2	107	110	3
3	121	123	2

Cuadro 4 Medición de exactitud en la medición de presión diastólica del esfigmomanómetro invadido

Usuario	Presión diastólica referencia [mmHg]	Presión diastólica [mmHg]	Desviación [mmHg]
1	73	69	4
2	71	70	1
3	83	81	2

ⁱⁱLa dimensional mmHg se refiere a la medida de presión en milímetros de mercurio

Cuadro 5 Medición de exactitud en la medición de pulso cardíaco del esfigmomanómetro invadido

Usuario	Pulso cardíaco referencia [Latidos por Minuto]	Pulso cardíaco [Latidos por Minuto]	Desviación [Latidos por Minuto]
1	58	60	2
2	70	76	6
3	69	72	3

XI. RESULTADOS SISTEMA DE MEDICIÓN DE TEMPERATURA

Cuadro 6 Medición de exactitud del termómetro diseñado

Individuo	Temperatura referencia [°C]	Temperatura diseño [°C]	Tiempo de adquisición [S]	Desviación [°C]
1	36.4	36.3	90	0.1
2	37	37	70	0.0
3	36.5	36.4	80	0.1
4	37	36.8	70	0.2
5	36.4	36.4	100	0.0
6	37	37	90	0.0
7	36.8	36.4	70	0.4
8	36.8	36.9	90	0.1
9	36.6	36.4	70	0.2
10	37	37	100	0.0
11	36.7	36.6	70	0.1
12	37.0	36.9	120	0.1
13	36.4	36.4	70	0.0
14	36.4	36.5	80	0.1
15	36.8	36.6	70	0.2

XII. RESULTADOS APLICACIÓN TELÉFONO CELULAR

La aplicación del teléfono celular funcionando como cliente, recibe datos del usuario de la interfaz presentada en la Figura 7, luego realiza una búsqueda de los dispositivos disponibles, al momento de encontrar el Módulo de Monitoreo inicia el proceso para realizar la conexión Bluetooth.

Figura 7 Interfaz aplicación Android



Se realizaron pruebas en distintos dispositivos de diferentes marcas, todos con la plataforma Android 2.1 y como se puede ver en el Cuadro 7. Se dieron problemas con la compatibilidad de la aplicación en distintos modelos de celular. Con estos resultados se llegó a la conclusión que la aplicación solo se puede ejecutar en dispositivos Sony Xperia. Esto se debe a la funcionalidad Bluetooth la cual tiene especificaciones diferentes para cada marca. Para corregir este problema y permitir que la aplicación sea compatible con la mayor cantidad de dispositivos es necesario investigar sobre la funcionalidad Bluetooth en cada marca, ver las especificaciones y cómo desarrollar aplicaciones Bluetooth que cumplan con ellas. Luego con esto desarrollar una aplicación más robusta que tome en cuenta el hardware del dispositivo.

Cuadro 7 Pruebas aplicación teléfono celular

Teléfono celular	Modelo	Conexión Bluetooth	SMS
0	Sony Ericsson XperiaU20a	si	si
1	Sony Ericsson GT-I5500L	si	si
2	Sony EricssonXperiaA853	si	si
3	SamsungGT-I5500L	no	no
4	SamsungGT-I5500L	no	no
5	MotorolaA853	no	no

XIII. RESULTADOS ADQUISICIÓN DE DATOS

Cuadro 8 Tiempo de adquisición de medición por sensor

Sensor	Tiempo de adquisición de medición mm:ss ⁱⁱⁱ
Esfigmomanómetro	01:30
Termómetro	02:00
Glucómetro	00:06
Frecuencia respiratoria	00:42
Balanza	00:16

Cuadro 9 Formato de datos que maneja cada sensor

Sensor	Bytes	Tipo de dato	Detalle de Formato
Esfigmomanómetro	3	Variable entero	Presión diastólica: ### Presión sistólica: ### Pulso cardiaco: ###
Termómetro	4	ASCII	Temperatura: DD.D ^{iv}
Glucómetro	11	ASCII	Concentración glucosa: DDD Hora DD Minuto DD Mes DD Día DD
Cinturón de Frecuencia respiratoria	1	Variable entero	Respiraciones por minuto ###
Balanza	6	ASCII	Peso: DDD.DD

ⁱⁱⁱmm:ss dimensional de tiempo, describe minutos con 2 dígitos y después de los dos puntos se colocan los segundos con dos dígitos.

^{iv}D se refiere a un dígito 0-9.

XIV. DISCUSIÓN

De acuerdo a la metodología se observa en los primeros resultados presentados en este reporte, la desviación de las mediciones de los módulos individuales de peso corporal, frecuencia respiratoria y temperatura corporal. En lo que se refiere a los resultados de presión arterial y glucosa, las tablas mostradas buscan ilustrar que posterior a la intervención de dichos dispositivos, estos siguen funcionando adecuadamente de acuerdo a las especificaciones del fabricante. Para una mejor comprensión de la metodología utilizada en cada sensor, se sugiere consultar los trabajos de graduación individuales, encontrados en las referencias (Rivera, 2011), (Cruz, 2011), (Juárez, 2011) y (Mejía, 2011).

Durante el diseño de la aplicación móvil, se probó etapa a etapa para garantizar su funcionamiento en por lo menos un teléfono celular. La primera parte a implementar fue la interfaz gráfica, y con base en esta se agregó la funcionalidad Bluetooth y envío de mensajes de texto. Durante la implementación de comunicación Bluetooth se presentó el problema de no poder abrir el puerto de la conexión, por lo que no se lograba comunicación entre el teléfono y el módulo Central de Información ver detalle que muestra Cruz (Cruz, 2011). Realizando diversas pruebas con configuraciones distintas de la librería Bluetooth, finalmente se estableció la conexión y se transmitieron y recibieron datos correctamente.

De igual manera, se implementó el envío de mensajes de texto ingresando el número telefónico desde la interfaz gráfica. Concluyendo la implementación, se comprobó la portabilidad de la aplicación del teléfono celular al realizar pruebas en diferentes modelos de teléfono de la plataforma Android. Se realizaron pruebas en las marcas Sony, Samsung y Motorola. Se puede ver en el Cuadro 7 que en los celulares Sony la funcionalidad Bluetooth y SMS funciona al 100%, sin embargo en las otras dos marcas no se logró establecer la comunicación Bluetooth. Esto se debe a la funcionalidad Bluetooth la cual tiene especificaciones diferentes para cada marca.

Para corregir este problema y permitir que la aplicación sea compatible con la mayor cantidad de dispositivos es necesario investigar sobre la funcionalidad Bluetooth en cada marca, ver las especificaciones y como desarrollar aplicaciones Bluetooth que cumplan con ellas. Luego con esto desarrollar una aplicación más robusta que tome en cuenta el hardware del dispositivo.

La primera etapa de la consolidación se estableció una comunicación exitosa entre el módulo de información y la aplicación del teléfono celular utilizando tecnología Bluetooth. Para determinar si los datos se recibían correctamente, se optó porque el teléfono enviara un eco de los datos recibidos nuevamente a la Central de Información, a manera de comprobar que los paquetes se enviaran y recibieran correctamente para mayor detalles ver (Rivera, 2011).

Una vez establecida esta comunicación, se agregaron individualmente los sensores adaptando sus formatos para el envío a través de la red. El algoritmo de adquisición de mediciones de los signos vitales de peso y frecuencia respiratoria como se puede ver en el Cuadro 8 se ejecuta por 16 y 42 segundos respectivamente, por lo cual se tuvo que implementar una interrupción para contestar al módulo Central de Información cuando éste hiciera la solicitud de recepción de datos. Al hacerlo de esta manera se evitó el inconveniente de que el Módulo Central de Información necesite esperar a que se tome la medición completa para que el equipo responda. Considerando que la interrupción se realiza en menos de 300 milisegundos, esto no altera el proceso de medición de estos signos vitales. Para el envío de la frecuencia respiratoria como se puede ver en el Cuadro 9, se envía una variable entera por el módulo Bluetooth y para el peso, se envían 6 caracteres ASCII que corresponden al peso real del instrumento.

En cuanto a la integración del sistema de medición de glucosa a la red se refiere, se integró al código implementado para la adquisición de las mediciones realizadas del glucómetro, las funciones que habilitan los Módulos Bluetooth, así como su configuración. Así mismo, se habilitó la interrupción en la recepción del módulo

UART2, en el cual se configuró el módulo Bluetooth, de tal forma que cuando el Módulo Central de Información envía la secuencia 0xFF, el glucómetro entra en modo de transferencia de datos de manera infrarroja. Cabe mencionar que para lograr este funcionamiento, fue necesaria la intervención de la placa de circuitos del glucómetro, soldándose cables a los contactos de los botones que el usuario debe presionar simultáneamente para cerrar el circuito y entrar al modo antes mencionado. De esta manera, cuando el módulo Central de Información solicita una medición, mediante el uso de un transistor 2N3904, se cierra el circuito conectando a tierra los contactos de los botones. Una vez el glucómetro ha entrado en modo de transferencia de datos, se procede establecer la comunicación con el mismo y obtener la medición almacenada en la memoria, siguiendo el protocolo encontrado en la fuente (Juárez, 2011). Finalmente, el dato adquirido se envía a la unidad Central de Información en un paquete de 3 bytes, siendo cada uno de estos el carácter ASCII correspondiente a las centenas, decena y unidades de la concentración en mg/dl medida más recientemente por el dispositivo. Es de resaltar, que si el glucómetro se encuentra en el proceso de realización de una medición, lo cual bloquea el modo de transferencia de datos, el sistema de medición de glucosa envía la secuencia 0xEEE indicando el fallo en la comunicación.

Por otro lado, en el proceso de integración del esfigmomanómetro a la red Konócete, se incluyeron, de la misma forma como se realizó con el sistema de medición de glucosa, las funciones necesarias para el código implementado previamente para la detección y obtención de una nueva medición por el dispositivo, para la configuración y habilitación del Módulo Bluetooth. De esta forma, cuando el circuito implementado para la adquisición de la presión arterial obtiene una nueva medición, este pone en alto una bandera que indica al módulo Central de Información que una nueva medición se ha adquirido. Posteriormente, el módulo de medición de presión arterial como se puede ver en el Cuadro 9 envía una secuencia de 3 bytes, correspondiendo cada uno de estos a la presión sistólica, diastólica y pulso cardíaco, respectivamente.

En el caso de la integración del sistema de medición de temperatura, se optó por incorporar el termómetro digital oral diseñado e implementado al Módulo de Medición de Presión Arterial. De esta forma, se pudo dar un mejor aprovechamiento de las capacidades del microcontrolador implementado en dicho módulo. Así mismo, con esta decisión se contribuyó a mantener la portabilidad del sistema, uno de los objetivos que se busca caractericen a este proyecto. Teniendo lo anterior en cuenta, se incorporó en el Módulo de Medición de Presión Arterial un botón para inicializar una nueva medición de temperatura, de tal forma que cuando una lectura nueva es completada, una bandera es lógica es puesta en alto, indicando que la medición está disponible. En este punto, el Módulo Central de Información solicita la medición, siendo esta enviada en una cadena de 4 caracteres, correspondiente a los valores ASCII de la temperatura en grados Celsius, por ejemplo 36.4 C.

La aplicación del teléfono celular es la encargada de mantener informado al médico sobre la condición del paciente. Las mediciones de los signos vitales tomadas por los diferentes sensores llegan a la aplicación del teléfono celular a través del Módulo Central de Información. Estos datos se utilizan para definir el mensaje de texto que se va a enviar al médico, para manejar estos datos se definieron rangos de los parámetros normales de las mediciones de presión, temperatura, glucosa, frecuencia respiratoria, peso y pulso cardiaco. Estos son:

- Presión arterial sistólica: Mayor a 90mmHg y menor a 120mmHg.
- Presión arterial diastólica: Mayor a 60mmHg y menor a 80mmHg.
- Temperatura corporal: Mayor a 36.44°C y menor a 37.55°C.
- Frecuencia respiratoria: Mayor a 14 y menor a 18 respiraciones por minuto.
- Nivel de glucosa en la sangre: Mayor a 135 miligramos por decilitro y menor a 140 miligramos por decilitro.
- Índice de masa corporal: Mayor a 18.5 y menor a 24.9.
- Pulso cardiaco: Entre 60 y 100 latidos por minuto.

Si las mediciones se salen de estos rangos, esto se indica en el mensaje de texto que se envía al médico.

XV. CONCLUSIONES Y RECOMENDACIONES

A. Conclusiones

1. Se logró implementar un sistema de medición de peso que tiene una desviación máxima de 4.9lb del peso real.
2. El sistema de frecuencia respiratoria implementado tiene una desviación máxima de 3 respiraciones por minuto con respecto al método convencional.
3. Se logró presentar un sistema que mide presión sistólica, presión diastólica, pulso cardíaco, concentración de glucosa en la sangre, temperatura, frecuencia respiratoria y peso en un paciente en menos de 5 minutos.
4. Con los resultados de las pruebas de la aplicación del teléfono celular se llegó a la conclusión que esta solo se puede ejecutar en dispositivos Sony Xperia. Esto se debe a la funcionalidad Bluetooth la cual tiene especificaciones diferentes para cada marca.

B. Recomendaciones

1. Se recomienda agregar un módulo independiente capaz de medir oxigenación de la sangre, estatura del paciente, así como un sistema capaz de realizar electrocardiogramas.
2. Para que la aplicación del teléfono celular sea compatible con la mayor cantidad de dispositivos es necesario investigar sobre la funcionalidad Bluetooth en cada marca, ver las especificaciones y como desarrollar aplicaciones Bluetooth que cumplan con ellas.
3. Se recomienda desarrollar un algoritmo de tarado automático para la balanza a manera de evitar el proceso de caracterización por un operador.

4. Se recomienda buscar un material con adecuadas capacidades de transferencia de calor, no tóxico para el humano, de tal manera que éste remplace el protector del termómetro digital, el cual puede resultar tóxico para el usuario dado su contenido de estaño.

XVI. BIBLIOGRAFÍA

- [1] American Telemedicine Association. (2011). *Telemedicine*. Recuperado el 24 de 9 de 2011, de American Telemedicine Association:
<http://www.americantelemed.org/i4a/pages/index.cfm?pageid=1>
- [2] Crowder, C. (2010). *How to Use a Micro SD Memory Card*. Recuperado el 30 de 10 de 2011, de eHow: http://www.ehow.com/how_4759596_use-micro-sd-memory-card.html
- [3] Cruz, C. L. (2011). *Konócte: Red Sensorial de Signos Vitales, Módulo de Comunicación y Monitoreo*. Universidad del Valle de Guatemala, Departamento de Ingeniería Electrónica, Guatemala.
- [4] Department Of Health and Human Services. (8 de 2004). *Prevention, Detection, Evaluation, and Treatment of High Blood Pressure*. Recuperado el 20 de 9 de 2011, de U.S. Department Of Health and Human Services. :
www.nhlbi.nih.gov/guidelines/hypertension/jnc7full.pdf
- [5] Developers, A. (2011). *Bluetooth | Android Developers*. Recuperado el 31 de 3 de 2011, de [http:// developer.android.com/guide/topics/fundamentals.html](http://developer.android.com/guide/topics/fundamentals.html)
- [6] Developers, A. (2011). *Using Hardware Devices | Android Developers*. . Recuperado el 27 de 4 de 2011, de Android Developers:
<http://developer.android.com/guide/developing/device.html>
- [7] Dhearts. (2011). *Normal Range of Respiration Rate*. Recuperado el 21 de 9 de 2011, de TheHearts: <http://dhearts.com/heart-health/healthy-0110.html>
- [8] Disabled World. (20 de 2 de 2008). *Blood Pressure Chart. Disability News, Information and Resources Disabled World*. Recuperado el 21 de 9 de 2011, de Disabled World: <http://www.disabled-world.com/artman/publish/bloodpressurechart.shtml>
- [9] Elert, G. (2005). *Temperature of a Healthy Human (Body Temperature)*. Recuperado el 21 de 9 de 2011, de hypertextbook.com:
<http://hypertextbook.com/facts/LenaWong.shtml>
- [10] Haseman, C. (2010). *Let Your Android Application Out of the Box with SMS*

- Integration .DevX*. Recuperado el 18 de 5 de 2011, de Your Information Source for Enterprise ApplicationDevelopment: <http://www.devx.com/wireless/Article/39495/1954>
- [11] Healthcare Online. (18 de 9 de 2011). *Normal BodyTemperature*. Recuperado el 21 de 9 de 2011, de OfficialHealthcare Online: <http://www.healthcare-online.org/Normal-Body-Temperature.html>
- [12] Healthwise. (14 de 7 de 2009). *BloodGlucoseLevels: Testing and Normal Range*. Recuperado el 21 de 9 de 2011, de WebMD Diabetes Center: Types, Causes, Symptoms, Tests, and Treatments: <http://diabetes.webmd.com/blood-glucose>
- [13] Juárez, L. A. (2011). *Konócete: Red Sensorial de Signos Vitales, Sistema de Medición de Temperatura, Glucosa y Presión Arterial*. Universidad del Valle de Guatemala, Guatemala.
- [14] Laskowski, E. (30 de 9 de 2010). *Heartrate: What's normal?*Recuperado el 22 de 9 de 2011, de Mayo Clinic: <http://www.mayoclinic.com/health/heart-rate/AN01906>
- [15] Marcano, R. (1 de 20 de 2011). *Medicina Preventiva*. Recuperado el 22 de 9 de 2011, de La Tensión o Presión Arterial: <http://www.medicinapreventiva.com.ve/auxilio/signos/tension.htm>
- [16] Mason Medical Center. (2011). *What are Normal BloodGlucoseLevels*Recuperado el 21 de 9 de 2011, de Team Medicine: <https://www.virginiamason.org/service.cfm?id=511>
- [17] Mejía, I. E. (2011). *Konócete: Red Sensorial de Signos Vitales, Sistema de Medición de Peso y Frecuencia Respiratoria*. Universidad del Valle de Guatemala, Departamento de Ingeniería Electrónica, Guatemala.
- [18] Mejia, S., Giraldo, D., Castrillon, C., & Cardon, O. (2005). *Desarrollo de una Interfaz para la Visualizacion de Signos Vitales*. Recuperado el 24 de 9 de 2011, de Universidad Pontificia Bolivariana de Colombia: <http://www.hab2005.sld.cu/arrepdf/T096.PDF>
- [19] Mitchell, B. (2010). *MAC Addressing*. Recuperado el 30 de 10 de 2011, de Introduction to the MAC Address: <http://compnetworking.about.com/od/networkprotocolsip/l/aa062202a.htm>

- [20] Mundt, C. (2004). *A Wearable Vital Signs Monitoring System*, Stanford University. Recuperado el 24 de 9 de 2011, de LifeWar:
http://lifeguard.stanford.edu/lifeguard_writeup_medium.pdf
- [21] NEMA. (2003). *HealthCare - HeartRate or Pulse*. Recuperado el 22 de 9 de 2011, de National Emergency Medicine Association NEMA:
http://www.nemahealth.org/programs/healthcare/heart_rate_pulse.htm
- [22] Ohio State University Medical Center. (2011). *Vital Signs (Body Temperature, Pulse Rate, Respiration Rate, Blood Pressure)*. Recuperado el 21 de 9 de 2011, de Ohio State University Medical Center:
http://medicalcenter.osu.edu/patientcare/healthcare_services/emergency_services/non_traumatic_emergencies/vital_signs/Pages/index.aspx
- [23] Open Handset Alliance. (2011). *Android Overview | Open Handset Alliance*. Recuperado el 21 de 11 de 2011, de Open Handset Alliance: http://www.openhandsetalliance.com/android_overview.html
- [24] Prevention, D. C. (13 de 9 de 2011). *Healthy Weight: Assessing Your Weight: BMI: About Adult BMI | DNPAO | CDC*. Recuperado el 21 de 9 de 2011, de Centers for Disease Control and Prevention:
http://www.cdc.gov/healthyweight/assessing/bmi/adult_bmi/index.html
- [25] Pusajo, Egurrola, & Hernandez. (2011). *Parámetros Hemodinámicos*. *Web Medica Argentina*. Recuperado el 21 de 9 de 2011, de Parámetros Hemodinámico:
http://www.webmedicaargentina.com.ar/TEMAS/parametros-hemodinamicos_htm.htm
- [26] Research, S. f. (2011). *For Biomedical Diabetes Research*. Recuperado el 21 de 11 de 2011, de Conversion of Glucose Values from mg/dl to mmol/l. SBDR:
http://www.soc-bdr.org/rds/authors/unit_tables_conversions_and_genetic_dictionaries/e5184/index_en.html
- [27] Rivera, J. M. (2011). *Konócte: Red Sensorial de Signos Vitales*, Central de Información. Universidad del Valle de Guatemala, Departamento de Ingeniería Electrónica, Guatemala.
- [28] RoboTech. (2010). *Bluetooth Serial Module User Manual, ver. 1.2*. Italia: Parallax

Inc.

[29] SIG, B. (2011). *Bluetooth WirelessTechnology Profiles*. Recuperado el 20 de 08 de 2011, de Bluetooth Developer Portal:

<http://developer.bluetooth.org/KnowledgeCenter/Pages/Technology-Overview.aspx>

[30] Smith, M. H. (2010). *Advances in Telemedicine*. Recuperado el 24 de 9 de 2011, de Bishop University: <http://thebucampus.ca/author/matthew-hamilton-smith/page/3/>

XVII. APÉNDICE A: Módulo Central de Información

A. Metodología

El diseño e implementación de la red de monitoreo requirió realizar varios experimentos con el propósito de comprender el funcionamiento de los dispositivos de comunicación Bluetooth para establecer comunicación entre los microcontroladores representantes del control y sensores. La metodología seguida para estos experimentos consistió en plantear el objetivo de la prueba, determinar qué dispositivos se iban a utilizar y diseñar e implementar el programa en el microcontrolador. A continuación se describen tres experimentos cuyos resultados permitieron plantear la topología final de la red. Por último se describe el algoritmo implementado y los resultados obtenidos al armar la red con comunicación hacia los sensores y el teléfono celular. Además se incluye una discusión que analiza la importancia de todos estos resultados y su importancia en el diseño final así como las conclusiones del proyecto.

B. Comunicación entre dos módulos Bluetooth (RBT-001)

1. Diseño general. El objetivo de este experimento era comunicar dos módulos Bluetooth Robotech RBT-001. Para alcanzar dicho objetivo se implementaron dos programas en lenguaje ensamblador para comunicar dos microcontroladores. El primer programa actúa como maestro de la comunicación y se utiliza la configuración de modo transparente de operación. El segundo programa actúa como esclavo de la comunicación buscando al dispositivo maestro y solicitando la comunicación. Para lograr este tipo de conexión, fue necesario conocer previamente la dirección Bluetooth única del dispositivo maestro. Esto se logró utilizando una computadora para realizar una búsqueda de dispositivos Bluetooth, identificar el módulo RBT-001 y anotar su dirección. Una vez establecida la comunicación, el esclavo le envía información de un contador al maestro para que este lo despliegue en una serie de LED's. Es importante notar que como el maestro funciona en Modo Transparente, éste nunca solicita la

información del esclavo, sólo despliega el dato recibido.

2. Diseño del Programa Maestro

a. Inicio

- 1) Se configura el reloj interno, el puerto serial con un baudrate de 9600, el módulo Bluetooth, y los puertos de entrada y salida del micro-controlador.

b. Ciclo principal del programa.

- 1) Revisa si se recibió un dato por comunicación serial.
- 2) Si se recibió un dato:
 - a) Despliegue en LEDs del dato.
 - b) Clear del bit de recepción serial.
 - c) Si NO se recibió un dato:
 - a. Parpadeo de un LED para mostrar cuando no se reciben datos.

3. Diseño del Programa Esclavo

a. Inicio

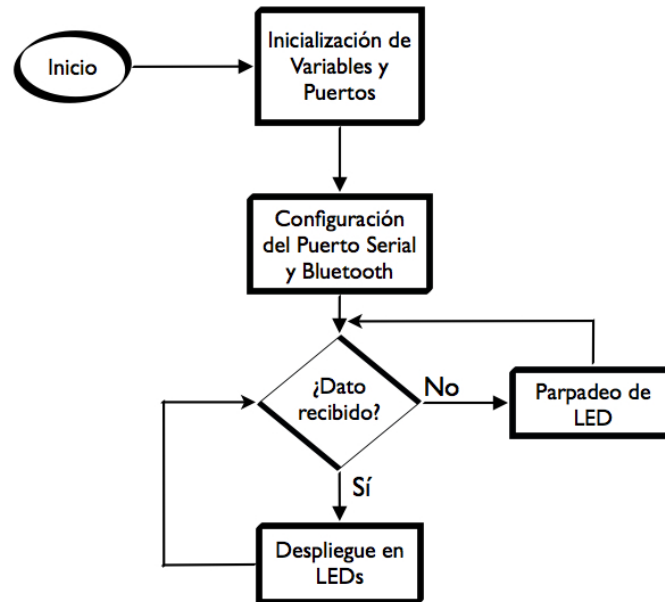
- 1) Se configura el reloj interno, el puerto serial con un baudrate de 9600 y los puertos de entrada y salida del microcontrolador.
- 2) Configuración del Módulo Bluetooth
- 3) Indicar que se desea establecer conexión con el dispositivo maestro.
- 4) Esperar 5 segundos para que el dispositivo realice la conexión.
- 5) Establecer el Modo Transparente del RBT-001.
- 6) Esperar 3 segundos

b. Ciclo principal.

- 1) Envío serial del contador.
- 2) Incremento del contador.
- 3) Retraso de 3ms.

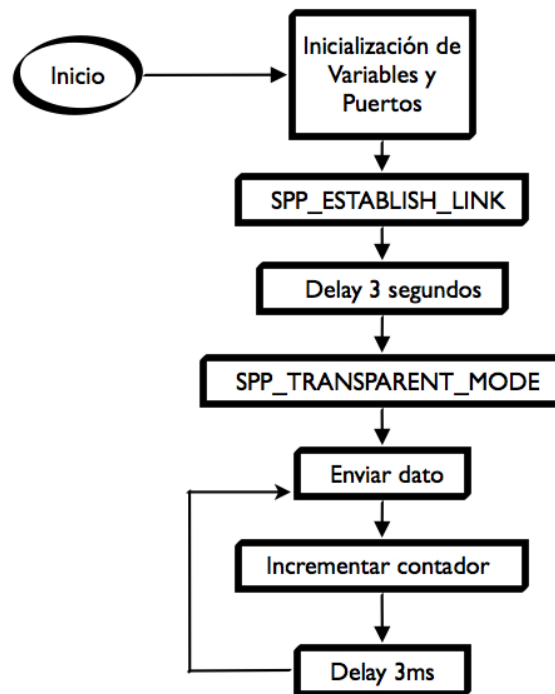
4. Resultados. El objetivo del experimento se alcanzó exitosamente pues se lograron comunicar dos microcontroladores utilizando la tecnología Bluetooth por medio de los módulos RBT-001. Se implementaron dos programas en lenguaje ensamblador, uno para el maestro controlador y otro para el sensor esclavo. Para el dispositivo maestro, se configuró el módulo Bluetooth para que funcionara en el modo transparente y actuara como un cable de comunicación RS-232 (full cable replacement). El programa únicamente despliega en LED's el dato recibido del módulo RBT-001. En caso haya instantes donde el microcontrolador no reciba ningún dato, un LED parpadea. (Cruz, 2011) La Figura 8 muestra el diagrama de flujo de este programa.

Figura 8 Diagrama de flujo del programa maestro.



Para realizar exitosamente la conexión se utilizó una computadora para averiguar las direcciones Bluetooth de ambos dispositivos RBT-001 y se encontraron que el dispositivo maestro tiene la dirección 00-17-a0-01-63-18 y el dispositivo esclavo tiene la dirección 00-17-a0-01-62-e4. También se implementó un programa en lenguaje ensamblador para el microcontrolador esclavo de la comunicación. En este programa se utilizaron dos configuraciones para el módulo Bluetooth. La primera configuración utiliza la instrucción SPP_ESTABLISH_LINK para establecer el enlace del puerto serial Bluetooth y debe especificarse la dirección del dispositivo con el cual se desea comunicar. Antes de continuar el programa se espera 5 segundos para que el dispositivo busque al maestro y se establezca la conexión. Luego se configuró nuevamente el módulo para establecer el modo transparente utilizando la instrucción SPP_TRANSPARENT_MODE que mantiene la conexión entre los dispositivos pero permite que el flujo de datos sin utilizar el *framing* para la comunicación entre el RBT-001 y el microcontrolador. La Figura 9 muestra el diagrama de flujo de este programa. Luego realizar la configuración de la comunicación, el programa entra en el ciclo principal donde envía el valor de un contador al microcontrolador maestro y lo incrementa.

Figura 9 Diagrama de flujo del programa esclavo.



C. Uso del modo comando para el programa maestro

1. Diseño. El propósito de este experimento fue conectar dos dispositivos Bluetooth, pero cambiando la configuración del maestro para que éste pudiera controlar el enlace Bluetooth y tuviera la capacidad de conectarse con hasta 6 dispositivos. Al igual que en capítulo anterior, se implementaron dos programas en lenguaje ensamblador para comunicar dos microcontroladores. A diferencia de este caso, esta vez se implementó la comunicación full-dúplex y se cambió la configuración del maestro y esclavo. El primer programa actúa como esclavo de la comunicación y se utiliza la misma configuración mencionada en la sección anterior. El segundo programa actúa como maestro de la comunicación buscando al dispositivo esclavo y solicitando la comunicación. Una vez establecida la conexión, ambos

microcontroladores envían y reciben información.

2. Diseño del Programa Maestro

a. Inicio

- 1) Se configura el reloj interno, el puerto serial con un baud rate de 9600, el módulo Bluetooth, y los puertos de entrada y salida del micro-controlador.
- 2) Configuración del Módulo Bluetooth.
- 3) Indicar que se desea establecer conexión con el dispositivo esclavo.
- 4) Esperar 5 segundos para que el dispositivo realice la conexión.

b. Ciclo principal del programa.

- 1) Enviar el contador utilizando SPP_SEND_DATA.
- 2) Incremento del contador.
- 3) Despliegue de los datos recibidos en forma serial.

3. Diseño del Programa Esclavo

c. Inicio

- 1) Se configura el reloj interno, el puerto serial con un baud rate de 9600, el módulo Bluetooth, y los puertos de entrada y salida del micro-controlador.
- 2) Configuración del Módulo Bluetooth utilizando el comando SPP_TRANSPARENT_MODE.

d. Ciclo principal del programa.

- 1) Envío serial del contador.
- 2) Incremento del contador.
- 3) Despliegue de los datos recibidos en forma serial.

4. Resultados. Para alcanzar exitosamente los objetivos de este experimento fue necesario estudiar detalladamente los comandos SPP_SEND DATA,

SPP_INCOMING_DATA y SET_EVENT_FILTER para crear una Piconet con los distintos módulos del proyecto. La Figura 10 muestra las especificaciones del comando SPP_SEND_DATA. En este comando es muy importante especificar el puerto RFCOMM que se utilizará para que la comunicación sea exitosa.

Figura 10 Detalles del comando SPP_SEND_DATA

Description		Send data on a SPP link to remote Bluetooth device
PacketType	REQ	
Opcode	SPP_SEND_DATA	
DataLength	3 + <PayloadSize>	
Data	LocalPort 1 byte	Local RFCOMM port number. Range 1-30
	PayloadSize 2 bytes	Number of data bytes to send. Valid range is 1 to 330 bytes..
	PayloadData <PayloadSize> bytes	The data to send.

La Figura 11 muestra las especificaciones del comando SPP_INCOMING DATA. Este comando es importante porque no se genera desde el microcontrolador sino que lo envía automáticamente el RBT-001 cada vez que recibe un dato. El microcontrolador recibe todos los 11 bytes y debe poder determinar cuál byte contiene la información enviada del otro dispositivo y cuáles bytes pertenecen al *framing* del RBT-001.

Figura 11 Especificaciones del comando SPP_INCOMING_DATA

Description		Incoming data on a DLC link, from a remote Bluetooth device
PacketType	IND	
Opcode	SPP_INCOMING_DATA	
DataLength	3 + <PayloadSize>	
Data	LocalPort 1 byte	Local RFCOMM port number. Range 1-30
	PayloadSize 2 bytes	Number of data bytes to send. Valid range is 1 to 330 bytes..
	PayloadData <PayloadSize> bytes	The data to send.

El comando SET_EVENT_FILTER fue utilizado (ver Figura 12) para evitar que el RBT-001 respondiera a cada comando del microcontrolador. Sin embargo, en esta etapa se descubrió que es necesario cambiar el parámetro para que no funcione en el

modo “full cable replacement” sino que el RBT-001 envíe indicadores y confirmaciones al microcontrolador.

Figura 12 Descripción de las especificaciones del comando SET_EVENT_FILTER.

Description	This command is used to set the event filter. The setting is stored in NVS.	
PacketType	REQ	
Opcode	SET_EVENT_FILTER	
DataLength	2	
Data	Filter 1 byte	0x00: All events reported 0x01: No ACL Link Indicators (default) 0x02: No events reported, UART break still generated and detected. 0x03: No events generated, UART break not generated or detected (full cable replacement)

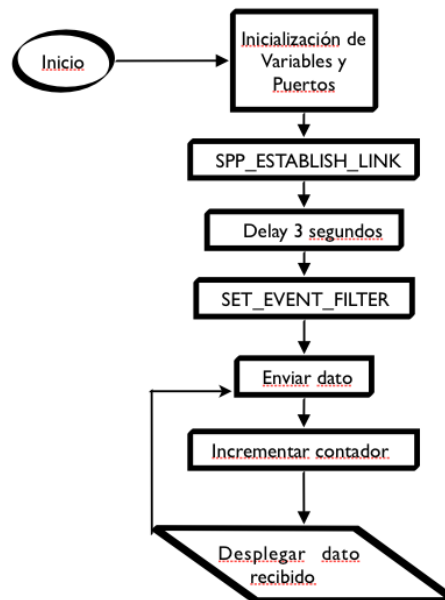
Además de los comandos, fue necesario estudiar las configuraciones del dispositivo para comprender cuándo actúa este como maestro o esclavo de la conexión mostrada en la Figura 13. En ellas se observa que para crear una Piconet, es necesario que al encender el dispositivo, este sea el que realice las conexiones hacia los demás dispositivos. Además, la tabla muestra que al ser maestro de una Piconet con hasta 7 esclavos, no se puede utilizar el modo transparente de operación, por lo que siempre se usará el *framing* para enviar y recibir datos con el microcontrolador. La figura también muestra la columna describiendo el funcionamiento de un Esclavo Transparente, el cual será el caso de todos los microcontroladores, a excepción del microcontrolador de control y monitoreo, y del teléfono celular. De esta columna lo más importante es que sí es posible utilizar el modo transparente para la comunicación entre el microcontrolador y el RBT-001.

Figura 13 Extracto que muestra las características del maestro Piconet y esclavo transparente del RBT-001.

Parameter / State	Piconet Master	Transparent Slave
UART Mode	Command	Transparent
Automatic Operation	-	-
Discoverable ¹	yes	no
Connectable ²	yes	no
Bluetooth Role	Master to x Slaves ³	Slave to 1 Master
Possible to search for devices (Inquiry)	yes	no
Connect to remote devices (actively)	yes ⁵	no
Send Raw Data	no	yes
State after incoming link	Scatternet Master	-
State after outgoing link	Piconet Master	-
State after sending "Transparent Mode" ⁶	Transparent Master	-
State after UART BREAK	-	Single Slave

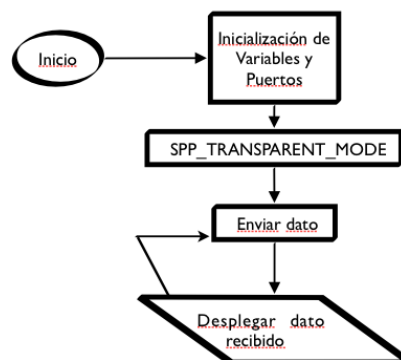
Con todo esto comprendido, se implementó el programa en lenguaje ensamblador para el microcontrolador que actuará como Maestro de la comunicación (ver Figura 14). Utilizando el *framing* del RBT-001, se utilizó el comando SPP_ESTABLISH_LINK de la misma manera que se utilizó en el experimento anterior para realizar la conexión con el otro módulo Robotech, y además se utilizó el comando SET_EVENT_FILTER para indicarle al módulo Robotech que sí debe comunicarse con el microcontrolador. Luego de realizar la conexión, el programa envía el valor de un contador por medio del Bluetooth y despliega el valor recibido en el puerto B.

Figura 14 Diagrama de flujo del programa Maestro.



Por último, también se implementó un programa en lenguaje ensamblador para el microcontrolador Esclavo de la comunicación (ver Figura 15). En este programa se utilizaron dos configuraciones para el módulo Bluetooth. La primera configuración utiliza la instrucción SET_EVENT_FILTER para configurar el módulo a “full cable replacement” utilizado con anterioridad. Luego se envió el comando SPP_TRANSPARENT_MODE para mantener la conexión SPP con el otro dispositivo pero sin tener que utilizar el *framing* para comunicarse con el RBT-001. Luego de realizar estas configuraciones, se entra en un ciclo que envía el valor de una variable utilizando RS-232 y despliega en el puerto B el valor recibido por la interfaz serial.

Figura 15 Diagrama de flujo del programa esclavo.



D. Comunicación entre el teléfono celular y dos módulos Bluetooth

1. Diseño. Este experimento tenía dos objetivos, primero abrir los puertos del RFCOMM del dispositivo maestro para poder establecer más de una conexión y segundo, establecer una conexión exitosa entre el teléfono celular, el microcontrolador maestro y el “sensor” esclavo. Para alcanzar el primer objetivo, se estudiaron detenidamente las características del Serial Port Profile en la hoja de datos de los dispositivos RBT-001(Cruz, 2011) para comprender cómo configurar los puertos del RFCOMM. Para comprobar que en efecto ambas conexiones, con el celular y con el sensor, se realizaran exitosamente, se monitoreó el puerto serial de entrada al microcontrolador maestro. Una vez alcanzado este objetivo, se configuró el dispositivo maestro para permitir estas conexiones y se realizaron pruebas para constatar que el segundo objetivo también fuera alcanzado.

2. Resultados. Para alcanzar el primer objetivo, se estudió el manual del dispositivo Robotech RBT-001(Cruz, 2011). El dato más importante encontrado fue la instrucción SET_PORTS_TO_OPEN (ver Figura 16) que permite abrir hasta 30 puertos RFCOMM distintos para la comunicación del módulo RBT-001 con otros dispositivos. En el caso del módulo de control, se necesitan tener 6 puertos abiertos, 1 para la comunicación con el teléfono celular y 5 para la comunicación con cada uno de los sensores.

Figura 16 Detalle de la instrucción SET_PORTS_TO_OPEN.

PacketType	REQ	
Opcode	SET_PORTS_TO_OPEN	
DataLength	4	
Data	PORTS 4 Bytes	This field is a 32-bit mask indicating which RFCOMM ports the RBT-001 has to open. Bit 30 and 31 must be set to 0. Bit 0 is RFCOMM port 1 and bit 29 is port 30 e.g. if this field has the value 0x00000007, port 1 to 3 will be opened. All other ports will be closed if open.

El segundo dato importante encontrado en el manual del Robotech RBT-001 fue la existencia de una memoria EEPROM a la cual se refieren como NVS (Non-Volatile Storage por sus siglas en inglés). El manual contiene una tabla donde presenta todas las localidades de la memoria y la información que se guarda en cada localidad.

En la Figura 17 se muestra un extracto de esta tabla donde se encontró la localidad de memoria donde se guarda la configuración de los puertos que el dispositivo abre al encenderse. Entonces, para corroborar que la instrucción SET_PORTS_TO_OPEN guardara la configuración de los puertos, se apagó el dispositivo y, al encenderlo nuevamente, utilizando la instrucción READ_NVS se leyeron 4 bytes de la localidad 0x56 para obtener la configuración de los puertos. La respuesta de esta lectura se muestra en la Figura 18, indicando que en efecto, se están habilitando los 7 puertos necesarios para la red. Una observación importante es que estos 7 puertos sólo es necesario habilitarlos en el Módulo Central de Monitoreo puesto que al establecer enlaces SPP, los dispositivos RBT-001 permiten que el puerto local sea distinto al puerto remoto.

Figura 17 Extracto de la tabla de localidades NVS.

No.	Address	Parameter	Default Value	Description	SW Reset required
11	0056	SppPorts-ToOpen	0x00000001	Bitmask defining the RFCOMM channels to open. For each channel one RFCOMM instance will be created.	no

Figura 18 Resultados de la lectura de la memoria NVS del módulo maestro.

Start	Tipo de Paquete	Opcod	Data	Length	Checksum	Paquete de datos	End	Descripción
0x02	0x69	0x25	0x05	0x00	0x93	0x04 0x30 0x32 0x31 0x32	0x03	Después del Reset, el RBT-001 está listo para recibir comandos.
0x02	0x43	0x72	0x08	0x00	0xBD	0x00 0x56 0x00 0x04 0x01 0x00 0x000x00	0x03	Lectura sin error de la localidad 0x0056. Leyó 4 bytes 0x00 00 00 01

Una vez se configuraron los puertos en el Módulo de Monitoreo Central, se realizaron pruebas para enlazar al teléfono celular con el Módulo Central y al Módulo

Central con el otro microcontrolador. La Figura 18 muestra una secuencia de bits recibida en la última prueba realizada donde se conectaron los tres dispositivos en una misma red. Se colocó únicamente esta tabla puesto que al conectar los 3 dispositivos se encuentran los mismos datos que al conectar cada uno por separado. La Figura 19 muestra, separado por columnas, el significado de cada byte obtenido y una breve descripción de la función del paquete entero siguiendo el *framing* mencionado anteriormente y mostrado en la Figura 3. De esta manera se explica cómo el módulo central de monitoreo inicia la conexión con el otro módulo RBT-001 utilizando el puerto local 0x02 y el puerto remoto 0x01 y cómo se establece esta conexión exitosamente. También se ve cómo, antes de que el celular se conecte al módulo, la central intenta enviarle datos, pero no hay conexión. Luego, se realiza exitosamente la conexión con el teléfono celular con el puerto local 0x01 y el puerto remoto 0x01. Una vez establecida la conexión con el teléfono, el envío y recepción de datos ocurre sin errores tanto con el teléfono como con el otro módulo RBT-001.

Además de la Figura 19 se incluyen abajo las Figuras 14, 15 y 16 que muestran fotografías de la aplicación en el teléfono celular y la recepción de datos del módulo RBT-001 esclavo. La Figura 13 muestra un resultado muy importante. Al momento de tomar la fotografía, se tenían 3 dispositivos Bluetooth, aparte del teléfono celular, encendidos: una computadora personal, el módulo RBT-001 maestro y el módulo RBT-001 esclavo. Sin embargo, cuando el celular hizo la búsqueda de dispositivos únicamente encontró 2: la computadora (Dirección Bluetooth 00-25-00-5E-C5-60) y el módulo RBT-001 maestro (Dirección Bluetooth 00-17-a0-01-63-18). Esto comprueba que la Piconet creada entre el maestro, esclavo y celular, funciona correctamente puesto que el esclavo no debe ser un dispositivo Bluetooth visible y que permita conexiones (ver Figura 13). A pesar que este resultado se documentó con la aplicación Java del celular, los mismos resultados se obtuvieron al probar con la aplicación Android. La Figura 15 muestra una fotografía de la aplicación cuando recibió el carácter 'J' del módulo maestro luego de que se realizara la conexión Bluetooth. De manera similar la Figura 16 muestra los LED's del puerto B del

dispositivo esclavo donde éste recibe el byte '01001010', binario del caracter 'J', del dispositivo maestro.

Figura 19 Conexión, envío y recepción de datos entre el módulo maestro, el esclavo y el teléfono celular.

Start	Tipo de paquete	Opcode	Data	Length	Checksum	Paquete	End	Descripción
0x02	0x43	0x0A	0x02	0x00	0x4F	0x00 0x02	0x03	Inicia el intento de conexión SPP en el puerto 0x02.
0x02	0x69	0x3E	0x04	0x00	0xAB	0x02 0x0C 0x00 0x00	0x03	Cambia el estado del puerto SPP 0x02
0x02	0x69	0x0B	0x09	0x00	0x7D	0x00 0xE4 0x62 0x01 0xA0 0x17 0x00 0x02 0x01	0x03	Establece el link SPP exitosamente con el otro RBT-001.
0x02	0x69	0x10	0x04	0x00	0x7D	0x02 0x01 0x00 0xF0	0x03	SPP_INCOMING_DATA: Puerto Local 0x02, Puerto Remoto 0x01
0x02	0x43	0x0F	0x02	0x00	0x54	0x1F 0x01	0x03	SPP_SEND_DATA al puerto 0x01 pero da el error 0x1F que indica que no hay conexión establecida en ese puerto.
0x02	0x43	0x0F	0x02	0x00	0x54	0x00 0x02	0x03	SPP_SEND_DATA por el puerto 0x02 sin error.
0x02	0x69	0x10	0x04	0x00	0x7D	0x02 0x01 0x00 0xF0	0x03	SPP_INCOMING_DATA: Puerto Local 0x02, Puerto Remoto 0x01
Continúa enviando datos primero al puerto 0x01 con error 0x1F y luego al puerto 0x02 sin error. Recibe por el puerto 0x02 el valor 0xF0								
0x02	0x69	0x0C	0x07	0x00	0x7C	0x1E 0xC7 0xF0 0x1A 0x4E 0x44 0x01	0x03	Conexión SPP solicitada remotamente fue establecida correctamente en el puerto 0x01
0x02	0x43	0x0F	0x02	0x00	0x54	0x1E 0x02	0x03	SPP_SEND_DATA por el puerto 0x02 con el error ERROR_CURRENTLY_NO_BUFFER
0x02	0x69	0x3E	0x04	0x00	0xAB	0x01 0x8C 0x00 0x00	0x03	Cambia el estado del puerto SPP 0x01 para comunicación con celular. En Alto: DSR, CTS, DLC.
0x02	0x43	0x0F	0x02	0x00	0x54	0x00 0x01	0x03	SPP_SEND_DATA por el puerto 0x01 sin error.
Recibe datos del celular una sola vez. Envía constantemente datos al celular por el puerto 0x01, y al otro módulo por el puerto 0x02 sin error. Recibe del otro módulo datos sin error.								

Figura 20 Dispositivos encontrados por la aplicación Java del teléfono celular.

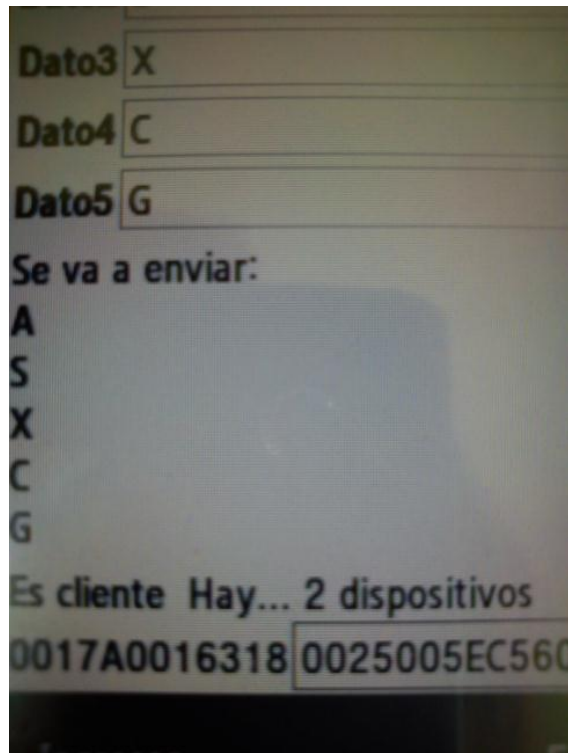


Figura 21 Datos recibidos por la aplicación Java del teléfono celular vía Bluetooth.

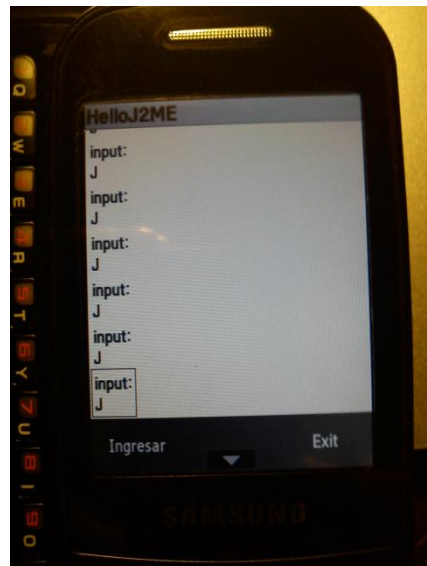
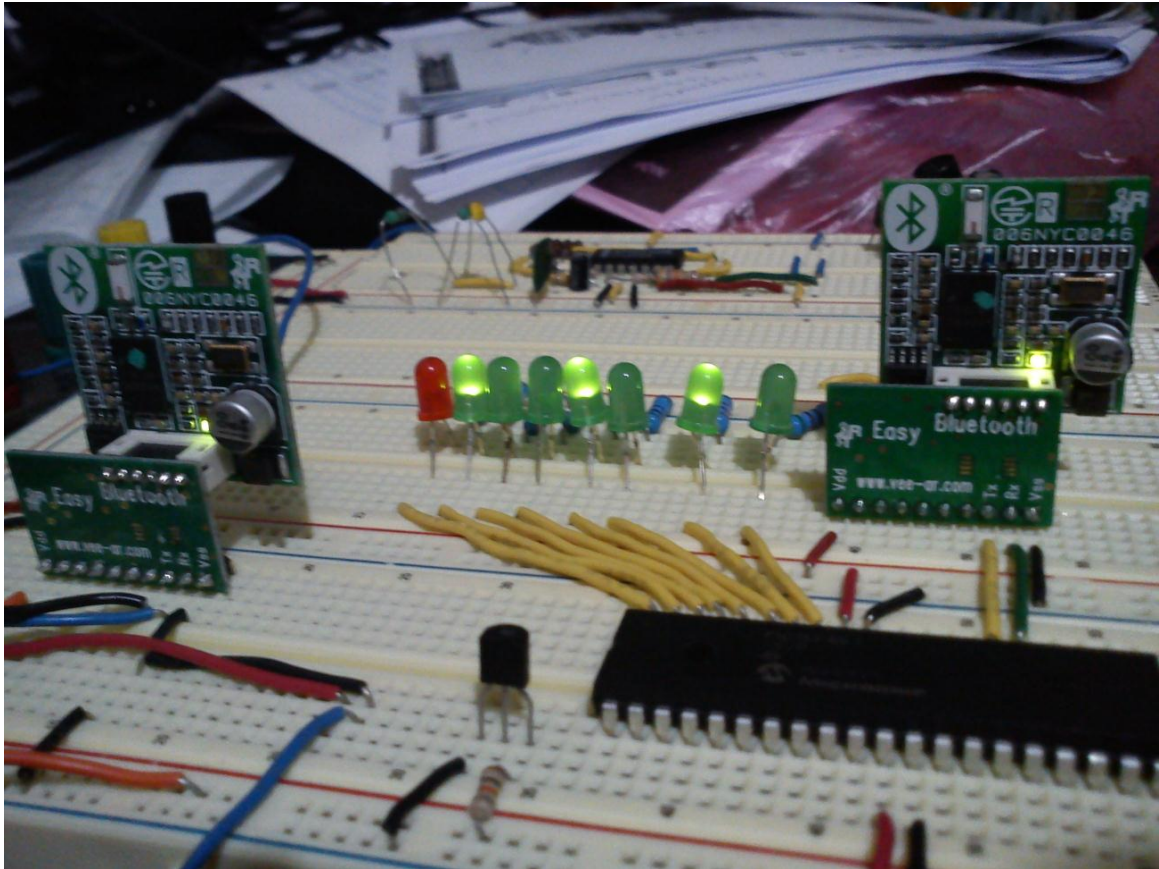


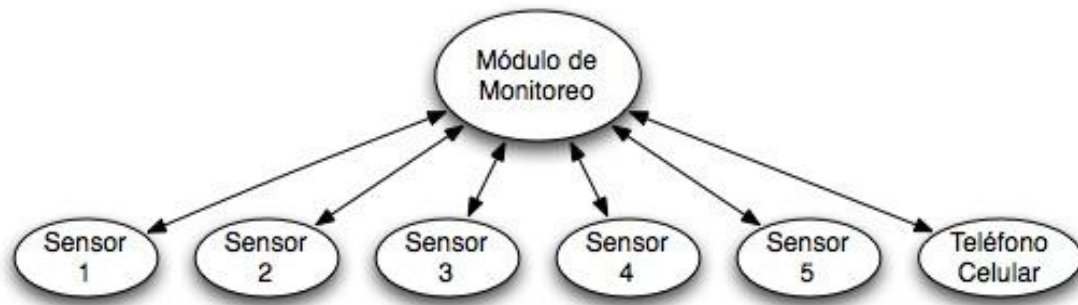
Figura 22 Dato recibido por el módulo esclavo vía Bluetooth del módulo maestro.



E. Implementación final de la red sensorial

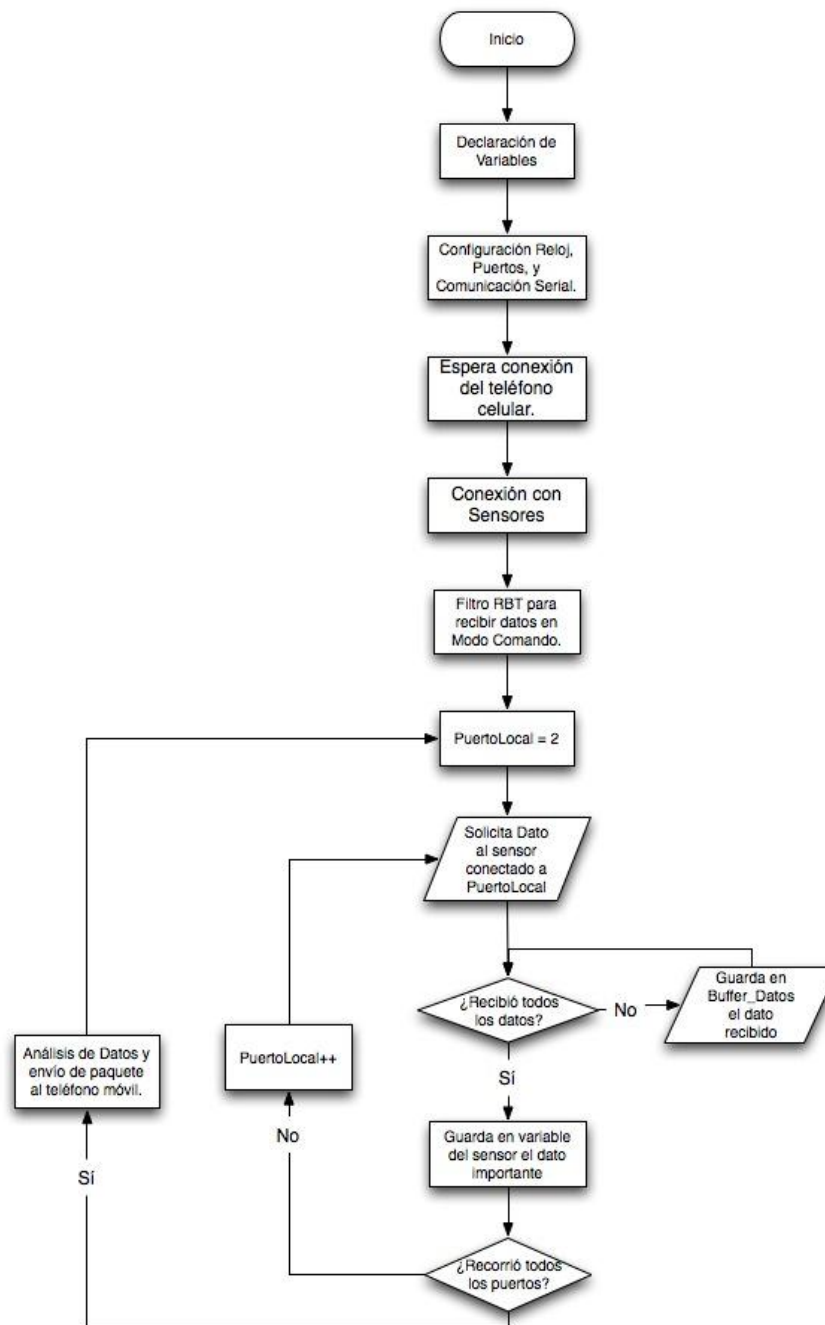
1. Diseño. Para lograr la implementación exitosa de la red sensorial fue necesario definir dos aspectos primordiales: la topología de la red y el algoritmo para el programa del microcontrolador maestro. La topología de la red se definió con base en el funcionamiento de los módulos RBT-001 en modo comando utilizando como base los resultados descritos en el capítulo “Comunicación entre el Teléfono Celular y dos Módulos Bluetooth”. De acuerdo con estos criterios, se necesitaba una topología que permitiera al maestro de la red obtener los datos de cada uno de los sensores y enviarlos al teléfono celular. La topología escogida fue la mostrada en la Figura 23, la cual muestra una topología de árbol que permite realizar un barrido de revisión de los sensores, para luego empaquetar la información y enviarla al teléfono.

Figura 23 Topología de la red de comunicación.



De acuerdo con la topología, también se diseñó el algoritmo del microcontrolador de monitoreo que revisará cada sensor para obtener la información. El diagrama del algoritmo a implementar se muestra en la Figura 24. Una de las partes cruciales del algoritmo se encuentra en la conexión con el teléfono celular. Debido al tipo de implementación que será el trabajo completado, se decidió que la mejor opción era esperar una conexión entrante del teléfono antes de iniciar la conexión con los sensores y la obtención de datos. Adicionalmente, es importante mencionar que después de cada conexión el programa revisa que la conexión sea exitosa, y de no serlo detiene el flujo del programa. Este programa fue escrito en lenguaje C de alto nivel utilizando la versión Lite del compilador High-Tech C de Microchip (Mejía, 2011).

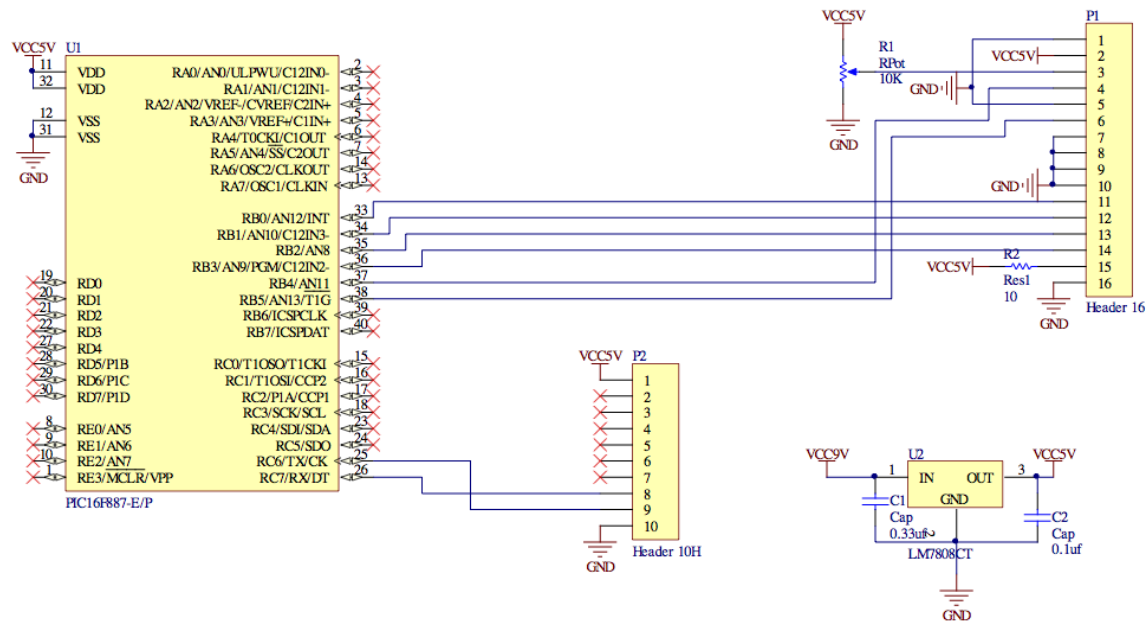
Figura 24 Algoritmo del programa del microcontrolador maestro.



2. Resultados. Siguiendo el diagrama de flujo mostrado en la Figura 24 se programó el microcontrolador 16F887, mismo que se ha utilizado en las secciones anteriores. Adicionalmente, se agregó un regulador de voltaje LM7805 para regular una batería de 9V a los 5V con los que funciona el microcontrolador y también se

agregó una pantalla LCD de 2x16 caracteres donde se despliegan mensajes para informar al usuario de las acciones que está ejecutando el programa. El circuito implementado se muestra en la Figura 25.

Figura 25 Circuito del Módulo de Monitoreo.



Una vez armado este circuito, se inició con la etapa de pruebas de conexión con los otros módulos Bluetooth y con el teléfono celular. En las Figuras 26, 27 y 28 se muestran los mensajes desplegados en la pantalla cuando se configura el módulo RBT-001, espera la conexión del teléfono y se conecta el módulo central al primer sensor.

Figura 26 Mensaje desplegado al ejecutar la instrucción SET_EVENT_FILTER.



Figura 27 Mensaje mostrado cuando el maestro espera la conexión del teléfono celular.



Figura 28: Mensaje mostrado al iniciar la conexión utilizando la instrucción SPP_ESTABLISH_LINK.



La primera etapa de pruebas incluyó únicamente pruebas de conexión, envío y transmisión de datos de dos sensores de la red. Durante estas pruebas fue necesario realizar una tabla que mostrara todos los posibles errores que pueden ocurrir durante la conexión para facilitarle al usuario la comprensión cuando algo no funciona correctamente en la comunicación. La Figura 29 se muestra el mensaje que será desplegado en la pantalla en caso de error y el tipo de error que representa. Es importante mencionar que cuando cualquiera de estos errores ocurra, la única manera de volver a establecer la conexión es apagando el Módulo de Monitoreo y volviéndolo a encender. Sin embargo, cabe destacar que durante todas las pruebas realizadas nunca ocurrió ninguno de estos errores.

Figura 29 Mensajes y códigos de error que pueden ocurrir

Mensaje	No. Error	Condición significado
Configurando Bluetooth...	--	Inicia la configuración del módulo Bluetooth RBT-001
Configuración Correcta	0x00	No ocurrió ningún error
Error No. __	0x01	Error en los parámetros de configuración.
	0x1B	Se excedieron los límites en los parámetros enviados.
Conectando Sensor __	--	Inicia la conexión con el sensor indicado.
Conexión Correcta	0x00 y	No ocurrió ningún error
	0x00	
Error No. __	0x22	Puerto SPP está ocupado.
	0x21	Puerto SPP no está habilitado
	0x20	Puerto SPP inválido
	0x26	Se solicitó iniciar una conexión que ya se estaba realizando.
Y __	0x01	No existe conexión DLC.
	0x02	No existe el puerto RF.
	0x03	No se pudo establecer el DLC.
	0x04	No se autorizó acceso al puerto.
	0x05	No existe una comunicación DLC/L2CAP

Otra observación realizada durante las pruebas se dio cuando se incorporó el teléfono celular a la red. En las primeras pruebas realizadas, se lograba establecer comunicación con el teléfono, pero luego era imposible establecer comunicación con los sensores. Para solucionar este inconveniente se monitoreó el puerto de recepción serial del microcontrolador y se encontró que cuando el celular se conectaba con el módulo RBT-001, este entraba en Modo Transparente, lo cual causaba que la única comunicación activa se diera entre el teléfono y el módulo, pero este no podía realizar conexiones con los demás sensores. Para eliminar este error fue necesario consultar nuevamente la hoja de datos del funcionamiento del RBT-001. En esta hoja se encontró la Figura 30 y 31, las cuales explicaron lo que sucedía al momento de realizar el teléfono la conexión. En la Figura 30 se encontró que el módulo RBT-001 está configurado por defecto para funcionar en modo automático, lo cual causa que al conectarse otro dispositivo a él, por ejemplo el celular, éste cambia automáticamente a funcionar en Modo Transparente y le es imposible realizar conexiones a otros dispositivos, según se puede ver circulado en verde en la Figura 30. Por ende, se investigó más a fondo la hoja de datos para cambiar esta configuración y permitir que el RBT-001 maestro pudiera recibir la conexión del teléfono y conectarse a los sensores. Se encontró que para lograr el funcionamiento deseado era necesario cambiar el RBT-001 maestro del funcionamiento automático al no-automático utilizando la función `WRITE_OPERATION_MODE` (Figura 32) del dispositivo Bluetooth. Una vez realizado este cambio el Módulo de Monitoreo fue capaz de esperar la conexión del teléfono y, una vez conectados, realizar las conexiones hacia los demás sensores de la red tal como lo muestra la Figura 31.

Figura 30 Extracto del funcionamiento del RBT-001 en modo automático.

Parameter / State	Idle Automatic	Transparent Slave
UART Mode	Command	Transparent
Automatic Operation	0x01 (On)	-
Discoverable ¹	yes	no
Connectable ²	yes	no
Bluetooth Role	-	Slave to 1 Master
Possible to search for devices (Inquiry)	yes	no
Connect to remote devices (actively)	yes	no
Send Raw Data	-	yes
State after incoming link	Transparent Slave	-
State after outgoing link	Piconet Master	-
State after sending "Transparent Mode" ⁶	-	-
State after UART BREAK	-	Single Slave

Figura 31 Extracto del funcionamiento del RBT-001 en modo No-Automático.

Parameter / State	Idle Non-Automatic	Single Slave
UART Mode	Command	Command
Automatic Operation	0x00 (Off)	-
Discoverable ¹	yes	yes
Connectable ²	yes	yes
Bluetooth Role	-	Slave to 1 Master
Possible to search for devices (Inquiry)	yes	yes
Connect to remote devices (actively)	yes	yes ⁴
Send Raw Data	-	no
State after incoming link	Single Slave	Scatternet Slave
State after outgoing link	Piconet Master	Scatternet Master
State after sending "Transparent Mode" ⁶	-	Transparent Slave
State after UART BREAK	-	-

Figura 32 Especificaciones del comando WRITE_OPERATION_MODE.

Description	This command will change the operation mode stored in NVS. The new setting will take effect after a reset.	
PacketType	REQ	
Opcode	WRITE_OPERATION_MODE	
DataLength	1	
Data	Mode 1 Byte	0x00 Automatic Operation OFF 0x01 Automatic Operation ON

F. Discusión

El proyecto Konócete tiene como objetivos diseñar e implementar una red sensorial para la medición de signos vitales con una plataforma para la comunicación por medio de un teléfono celular para el envío de mensajes de texto con reportes acerca del estado de salud del paciente. Este trabajo se enfocó en la implementación de la red de comunicación inalámbrica que comunica los sensores de signos vitales con la Central de Información y esta a su vez se conecta con el teléfono celular. Para lograr una implementación adecuada de dicha red se escogió la tecnología Bluetooth como interfaz de comunicación, se realizaron varios experimentos para determinar la mejor implementación de la red y por último se incluyó una pantalla LCD para mostrar al usuario mensajes acerca del funcionamiento de la red y los datos obtenidos de los sensores. De esta manera se logró implementar la red propuesta y alcanzar los objetivos individuales establecidos.

Previo al diseño de la red, fue necesario estudiar y elegir el medio por el cual se comunicarían los dispositivos. El medio, por preferencia y simplicidad física para la conexión de dispositivos fue el medio inalámbrico. Una vez elegida esta opción, se estudiaron distintas tecnologías de comunicación inalámbrica superficialmente, y se escogió la tecnología Bluetooth desde el inicio por haber sido diseñada para la comunicación en redes con pocos dispositivos. Seguidamente se buscaron dispositivos que permitieran la comunicación con esta tecnología y los cuales tuvieran una interfaz sencilla. Esta búsqueda resultó en el dispositivo RBT-001 ampliamente descrito a lo largo de este trabajo. Una vez encontrado este no se consideró necesario realizar alguna comparación con otros dispositivos o continuar la búsqueda pues este cumplía los requerimientos. Incluso, a pesar de que en efecto se encontraron otros dispositivos no se incluyeron sus características o comparaciones pues ningún otro presentó la facilidad de interfaz con microcontroladores como lo hizo el RBT-001. Esto llevó a la conclusión de que las comparaciones realmente no tenían sentido alguno pues al ser Bluetooth un estándar, todos los dispositivos tenían similares características a excepción de la interfaz de comunicación RS-232.

Una vez establecido el medio se realizó la compra de los dispositivos RBT-001 y se inició con la etapa de pruebas. De todas las pruebas y experimentos realizados, prevalecen los resultados de los experimentos mencionados en este trabajo. El primer experimento fue una exploración del funcionamiento de los módulos y sus resultados justifican la facilidad de comunicación entre el microcontrolador y el RBT-001 mencionada con anterioridad. Los siguientes experimentos documentan el proceso de diseño de la red implementada y demuestran cómo se llegó al resultado final. Una parte primordial en el diseño de la red fue la parte del monitoreo del puerto serial de recepción del microcontrolador. Esta etapa se considera de suma importancia pues permitió generar las Figuras 18, 19 y 30 que presentan las respuestas del RBT-001 a los comandos de conexión descritos en los resultados. De estas pruebas se desarrolló el algoritmo de implementación final pues se conoce cómo va a responder el RBT-001 a cada configuración solicitada. Las últimas pruebas y experimentos fueron dedicados a unificar los sensores a la red y comprobar que la Central de Información funcionara correctamente utilizando la interfaz de la LCD. También en esta última etapa se realizaron las pruebas con la aplicación Android del teléfono celular. De todas las 20 pruebas realizadas, no se encontró pérdida de datos, sólo se observó un retardo en la conexión cuando habían demasiados dispositivos Bluetooth en el salón de pruebas y se concluye que este retardo en la conexión se debe a que para encontrar el dispositivo solicitado debe revisar cada dirección Bluetooth encontrada.

Además, las pruebas realizadas corroboraron que la topología propuesta fue adecuada para la aplicación propuesta. La primera razón por la cual fue un éxito la topología propuesta fue porque simplificó el proceso de unificación de la red. En la unificación, los sensores tuvieron que incluir dos instrucciones para configurar su módulo RBT-001 en Modo Transparente y con reemplazo completo de cable para recibir toda la información directamente en RS-232. Esto facilitó inmensamente el trabajo debido a que cada sensor utiliza un lenguaje y flujo de programa diferente, por lo cual utilizar el modo comando hubiera complicado la integración e incluso podría afectar el algoritmo de obtención de la medición. En cambio, utilizando la

comunicación RS-232, los microcontroladores no pierden mucho tiempo enviando el dato y pueden dedicarse a realizar las mediciones. Además, para evitar que los sensores tuvieran complicaciones para saber quién se comunica con ellos, cada sensor tiene una única conexión con la Central de Información y la central es la que controla con quiénes está conectado y quién debe enviarle información. Fue así como la integración se simplificó y según las pruebas los datos se recibieron dentro del período de tiempo propuesto y sin errores.

Además de la integración de los sensores, fue muy importante corroborar el funcionamiento del algoritmo de la Central de Información. Esto se logró por medio de las pruebas y utilizando los mensajes desplegados en la pantalla LCD. También se compararon los resultados de la Figura 19 con los mensajes desplegados para corroborar que verdaderamente no hubiera errores en la comunicación. Después de las pruebas y de corroborar que los mensajes mostraran los sucesos reales pasando en la comunicación, se concluyó que la implementación del algoritmo fue el adecuado y cumple con los objetivos propuestos. Ésta es tal vez la parte que más se podría optimizar pues a pesar de que la implementación actual funciona adecuadamente para la aplicación propuesta, si en algún punto se desea expandir el proyecto, es importante mejorar ciertos aspectos. Uno de los aspectos más importantes a considerar en una futura expansión es el dispositivo Bluetooth RBT-001 debido a la cantidad máxima de dispositivos que se pueden conectar en la red. En la configuración de red actual el máximo número de sensores que se pueden conectar es 7 lo cual deja únicamente 2 sensores que se pudieran agregar en una expansión. Si fuera necesario agregar más, es imperativo cambiar la topología de la red para lograr la expansión. Adicionalmente, debe revisarse las capacidades del microcontrolador implementado para determinar si su capacidad de procesamiento es suficiente para la teórica expansión.

Fue a través de estos experimentos y pruebas que se logró implementar exitosamente el Módulo Central de Información. Los resultados de los experimentos muestran el avance progresivo del módulo hasta su implementación final. La pantalla

LCD proporciona una interfaz gráfica apropiada para mostrar los mensajes en el avance del programa. Unificando todos estos detalles se entregó el Módulo Central de Información funcionando con comunicación cumpliendo los objetivos y con un éxito excepcional en la transmisión y recepción de datos.

XVIII. APÉNDICE B: Módulo de Monitoreo y comunicación

A. Metodología

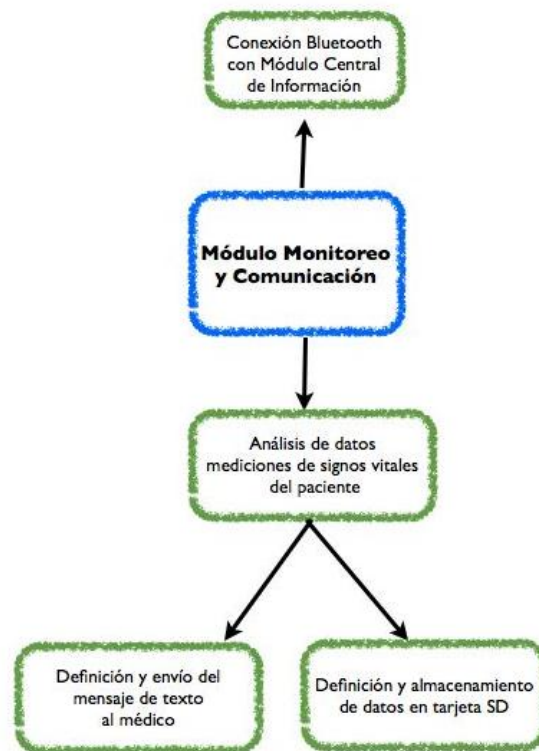
1. Investigación inicial. La investigación inicial tiene como objetivo adquirir un panorama general del problema que se está enfrentando con el desarrollo del trabajo de graduación. Durante esta etapa se realizaron investigaciones sobre tecnología Bluetooth, SMS, diseño y desarrollo de aplicaciones para celulares y plataformas disponibles para desarrollar aplicaciones para celulares.

2. Investigación temática. Este tipo de investigación se realizó varias veces a lo largo del trabajo de graduación, incluye la investigación necesaria para realizar cada fase del proyecto. En esta fase se adentró en temas más específicos del desarrollo de la solución como: Plataforma J2ME, ciclo de vida de una aplicación J2ME, tecnología Bluetooth en aplicaciones J2ME, tecnología SMS en aplicaciones J2ME, plataforma Android, ciclo de vida de una aplicación Android, "Activity" en Android, tecnología Bluetooth en aplicaciones Android y tecnología SMS en aplicaciones Android así como otros temas necesarios.

3. Diseño. Con base en los resultados de las dos etapas anteriores se logró definir el lenguaje de programación a utilizar, la plataforma para la cual se desarrollaría la aplicación, así como el flujo del programa a implementar.

Como se puede ver en el diagrama de bloques de la aplicación presentado en la Figura 33. Esta aplicación debe desarrollar un módulo que realice y mantenga la conexión Bluetooth con el módulo Central de Información. Además debe contener un módulo que analice los datos obtenidos y desarrollar la funcionalidad de mensajes de texto para informar al médico sobre la condición del paciente, así como la funcionalidad de almacenamiento en la tarjeta micro SD del teléfono celular para mantener un registro del paciente.

Figura 33 Diagrama de bloques Konócte módulo monitoreo y comunicación



4. Construcción y evaluación. En esta etapa se desarrolló y se probó la herramienta de software. Se espera que al final del proyecto de graduación en esta aplicación se reciban datos del Módulo de Monitoreo y a partir de estos seleccionar una plantilla y enviarla al médico.

Luego de desarrollar la aplicación se van a realizar pruebas en 5 teléfonos celulares de diferentes marcas con plataforma Android para comprobar la portabilidad de la aplicación.

5. Mantenimiento. Con base en lo investigado, luego de desarrollar la aplicación para la plataforma Android, se decidió trabajar con plataforma Android para aprovechar tecnología de smartphones.

6. Desarrollo de recomendaciones y trabajo escrito. Para concluir el Megaproyecto se dejan algunas recomendaciones sobre mejoras que se podrían realizar a este Módulo de Monitoreo y Comunicación. Esta documentación va a ayudar en el trabajo a quienes quieran continuar con el proyecto en un futuro. También se incluirá una sección de recomendaciones las cuales se cree que se

deben de tomar en cuenta en fases futuras del proyecto o en la implementación del proyecto en la medicina de Guatemala.

B. Aplicación J2ME

1. Diseño

a. Conexión Bluetooth. Se quiere desarrollar una aplicación Bluetooth para la plataforma J2ME, el objetivo de esta aplicación es realizar una búsqueda de los dispositivos Bluetooth cercanos y encontrar el módulo con el que se van a intercambiar los datos ingresados por el usuario. Esta aplicación funciona como cliente y sigue el siguiente procedimiento:

1) Inicio

a) Definir variables

b) Definir nombre del dispositivo con el que se van a intercambiar datos

c) startApp()

1] Crear botón de exit y de ingresar

2] Crear textField para ingresar datos del paciente

3] Agregar componentes al Form principal y agregar "commandlistener"

4] commandAccion()

a] Si se selecciona la opción "exit"

1- Salir de la aplicación.

b] Si se selecciona la opción "ingresar"

1- Obtener información de cada textField.

2- Crear cadena de texto a enviar por Bluetooth

3- doDeviceDiscovery()

4- Crear objetos LocalDevice y DiscoveryAgent (Con estos objeto se puede tener acceso a las propiedades de Bluetooth del dispositivo como la dirección, el "Friendlyname" y las propiedades de visibilidad)

- proceso de búsqueda)
- 5- startInquiry() (inicializar el proceso de búsqueda)
- 6- inquiryCompleted() (termina el proceso de búsqueda)
- encontrados
- 7- Recorrer lista de dispositivos encontrados
- 8- Buscar su nombre y comparar con nombre del dispositivo con el que se van a intercambiar datos.
- 9- ¿Se encontró el dispositivo?
- 10- doServiceSearch() (Buscar los servicios que ofrece el dispositivo)
- Al momento de encontrar el servicio que se que se necesita, se establece una conexión con el servidor, para esto se utiliza un objeto "Connector".
- 11- Inicializar objeto Connector
- 12- openOutputStream(): Con la conexión se abre un output stream para poder enviar datos al otro dispositivo.
- 13- openInputStream(): Con la conexión se abre un output stream para poder recibir datos del otro dispositivo.
- 14- Envía los datos del paciente con el método write()
- 15- Inicializa thread para estar esperando datos constantemente.
- 16- Utiliza el método read() para leer el dato recibido.
- 17- Verifica el tamaño del dato recibido.
- 18- Almacena el dato recibido en un arreglo de bytes.
- 19- Convierte bytes en String y despliega en pantalla el resultado.

En las Figuras 34 y 35 se puede ver el flujo que sigue el programa para la plataforma J2ME para realizar la conexión Bluetooth con el Módulo Central de Información.

Figura 34 Diagrama de flujo Command Action en J2ME

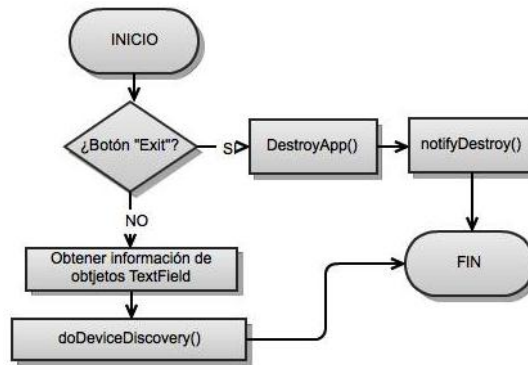
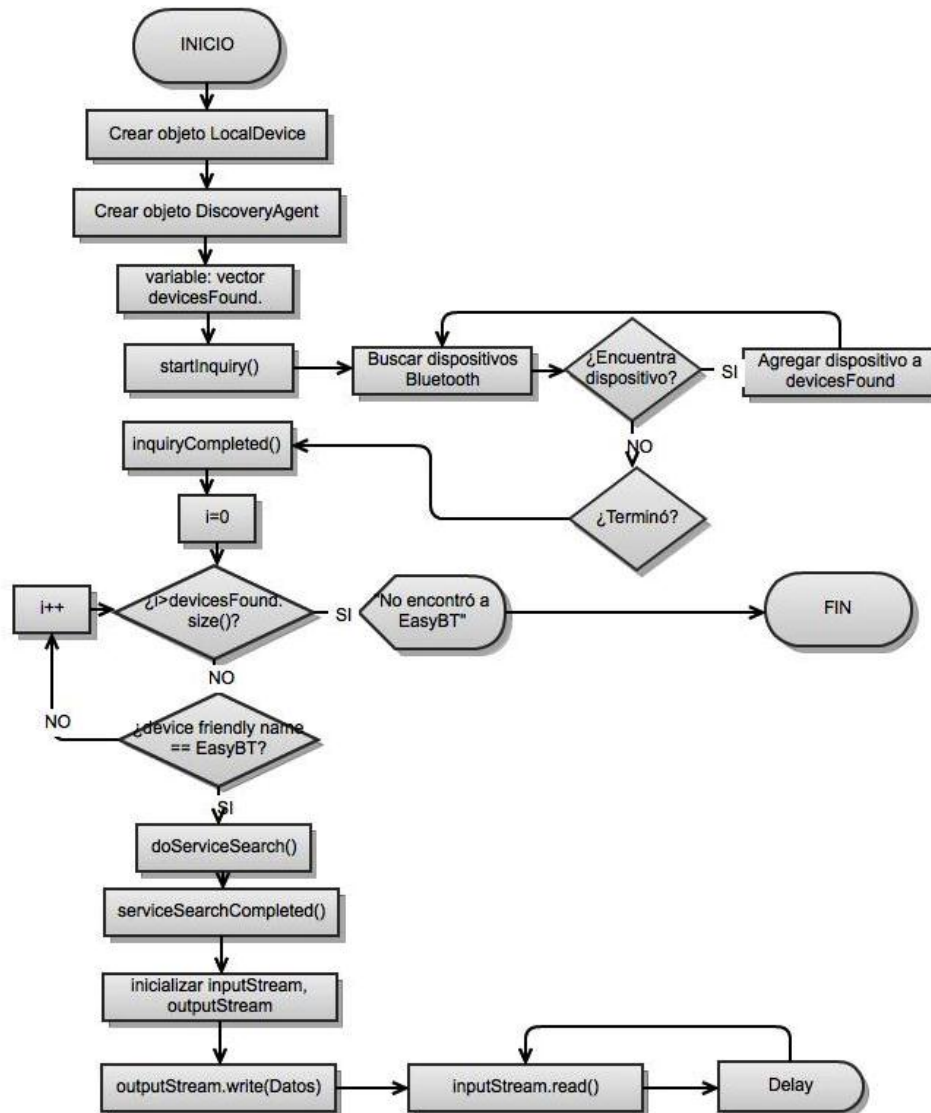
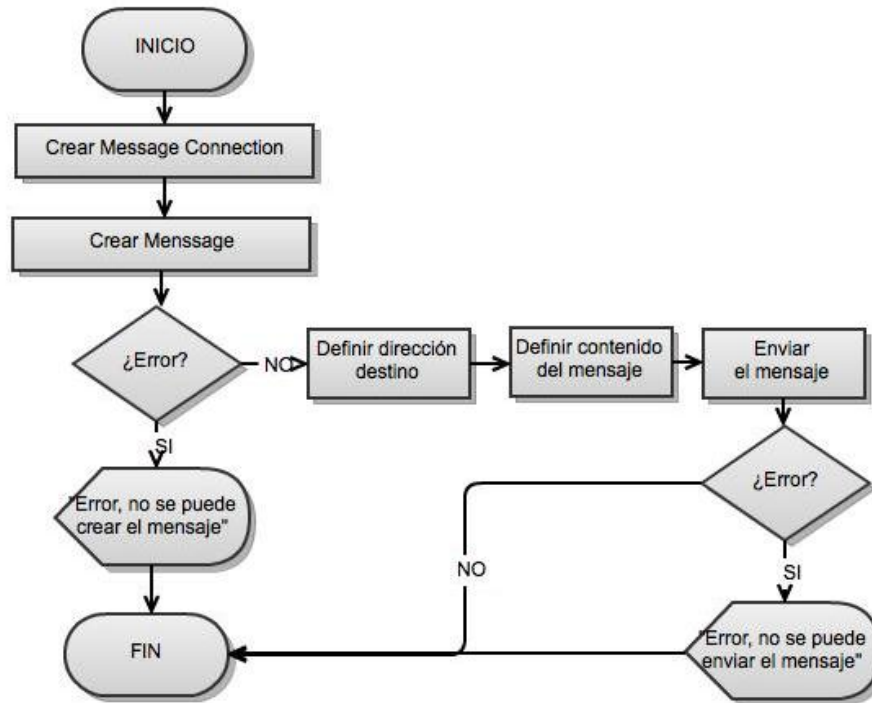


Figura 35 Diagrama de flujo Bluetooth en J2ME



b. SMS En la Figura 36 se puede ver el flujo que sigue el programa para la plataforma J2ME para crear y enviar un mensaje de texto.

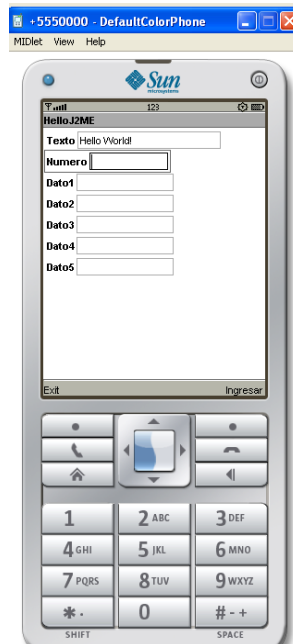
Figura 36 Diagrama de flujo funcionalidad SMS en J2ME



2. Resultados. En la primera versión de la aplicación se trabajó con la herramienta J2ME Wireless Toolkit 2.5.2_01 para desarrollar la aplicación encargada de la comunicación con el médico. Para más detalles revisar código Konócete J2ME en el apéndice.

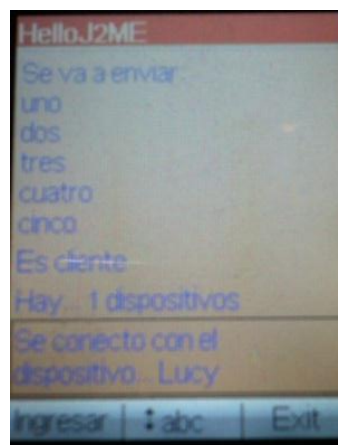
a. Interfaz de Usuario. Para que el médico pueda ingresar datos del paciente se diseñó la interfaz de usuario con varios cuadros de texto en los que se deben de ingresar los datos solicitados por el sistema. Esta interfaz se puede ver en la Figura 10. Aquí se pide al usuario que ingrese un texto, un número de teléfono y 5 datos del paciente los cuales se modifican en las siguientes fases del megaproyecto para adaptarse a los datos necesarios para comprobar que las mediciones de signos vitales se encuentren dentro de los límites establecidos.

Figura 37 Interfaz de usuario de la aplicación ejecutándose en el simulador J2ME



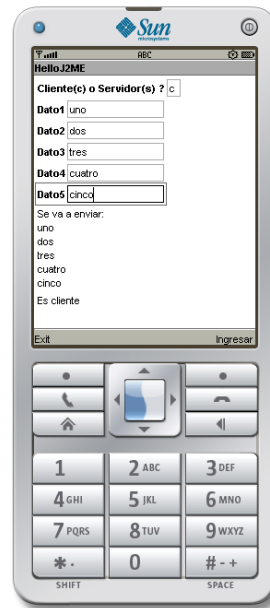
b. Funcionalidad Bluetooth. Ya que se definió la interfaz de usuario se prosiguió a implementar funcionalidad Bluetooth. Esta conexión permite que el teléfono celular intercambie datos con el Módulo Central de Monitoreo. Se obtuvo una aplicación ejecutada en un teléfono celular “ZTE” que luego del ingreso de datos, indica la cantidad de dispositivos disponibles y encuentra el dispositivo con el que se van a intercambiar los datos. Esto se puede ver en la Figura 38. Conexión Bluetooth Aplicación J2ME.

Figura 38 Conexión Bluetooth aplicación J2ME



En esta primera versión de la aplicación se definió que al iniciar la conexión Bluetooth se iba a enviar al Módulo Central de Información los rangos aceptables de las mediciones de signos vitales. Se puede ver en la Figura 39 el envío de datos a Módulo Central de Monitoreo aplicación J2ME.

Figura 39 Envío de datos a módulo central de monitoreo aplicación J2ME



Se realizaron pruebas de la aplicación en un teléfono celular “LG” que luego del ingreso de datos, indica la cantidad de dispositivos disponibles, encuentra el dispositivo con el que se van a intercambiar los datos le envía los datos del paciente y luego recibe datos para desplegarlos en pantalla.

a. SMS. Ya que se tiene funcionalidad Bluetooth se prosiguió a implementar funcionalidad para enviar mensajes de texto. Se diseñó y desarrolló una aplicación J2ME para un teléfono celular la cual envía un mensaje de texto al médico del paciente cuando el módulo Central de Monitoreo lo indica.

Para esto se utilizó tecnología SMS. Se utilizó una librería de J2ME llamada Java Wireless Messaging API (WMA) la cual proporciona métodos para envío de mensajes de texto.

Se crearon dos objetos importantes, para manejar el envío de mensajes. `MessageConnection` que crea la conexión y contiene los métodos de `newMessage()` y `send()`. El otro objeto utilizado fue `Message` que define el contenido del mensaje a enviar, este contenido se define utilizando el método de `setPayloadText()`.

3. Discusión. Desde el inicio del Megaproyecto en junio 2010 se trabajó en la aplicación del teléfono celular que permite la comunicación con el médico y con el Módulo de Monitoreo. En el ciclo junio – noviembre 2010 se desarrolló esta aplicación para la plataforma J2ME. Para evitar los errores de compatibilidad obtenidos en la aplicación J2ME se decidió desarrollar una nueva aplicación para plataforma Android para que ésta pueda ser ejecutada en la mayor cantidad de dispositivos. En el ciclo enero – mayo 2011 se creó una nueva versión de la misma aplicación, pero para la plataforma Android.

La primera fase del Megaproyecto se construyó una aplicación para un teléfono celular con plataforma J2ME que permite la comunicación entre el módulo Central de Monitoreo y Comunicación. Se logró establecer esta conexión que permite el intercambio de datos entre los dos módulos utilizando la tecnología Bluetooth JSR82.

También se logró agregar la funcionalidad de enviar un mensaje de texto a un número específico para que en el momento en que alguna medición este fuera de los parámetros normales, se le pueda avisar al médico.

Se tuvieron problemas de compatibilidad ya que la funcionalidad Bluetooth no se logró ejecutar en el celular “ZTE” pero sí en el celular “LG”. Se quiere que la aplicación pueda ser ejecutada en la mayor cantidad de dispositivos por lo que llegó a la conclusión que es necesario desarrollar una nueva aplicación para plataforma Android para evitar errores de compatibilidad.

C. Aplicación Android

1. Diseño

a. Bluetooth. La aplicación del teléfono celular va a funcionar como cliente, en las Figuras 41, 42 y 43 se puede ver los procesos a seguir para lograr una conexión Bluetooth exitosa. Dicho proceso lleva varios pasos: configurar Bluetooth, encontrar dispositivos disponibles, realizar conexión con un dispositivo y transferir datos. Se utilizó la librería android.bluetooth En las siguientes figuras se puede ver el procedimiento utilizado.

Figura 40 Diagrama de flujo configuración Bluetooth Android

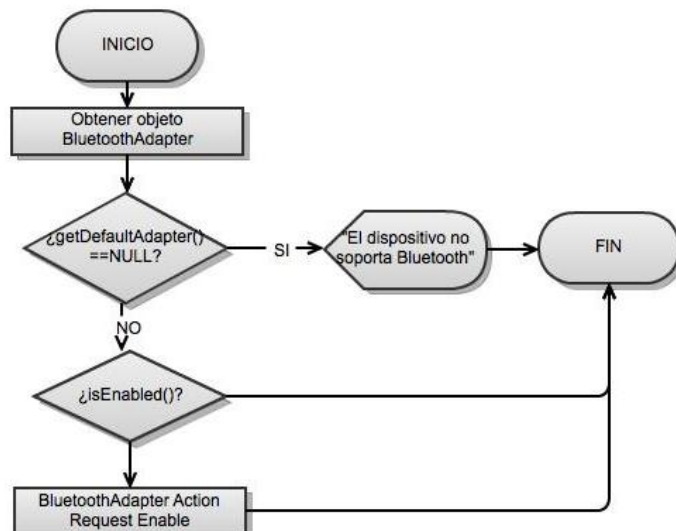


Figura 41 Diagrama de flujo búsqueda de dispositivos Bluetooth Android

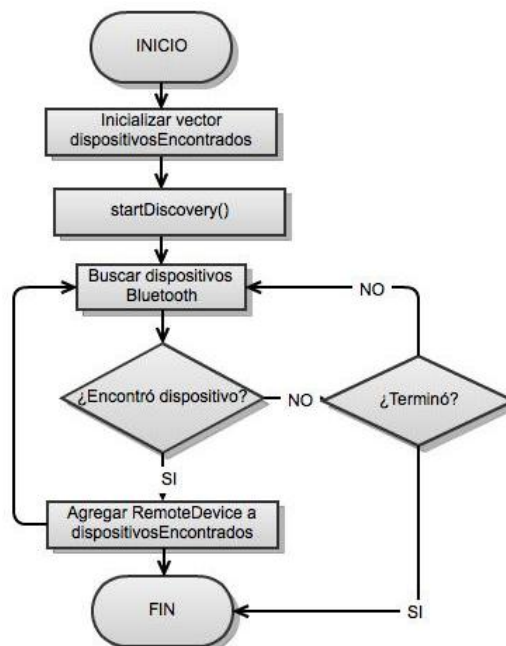
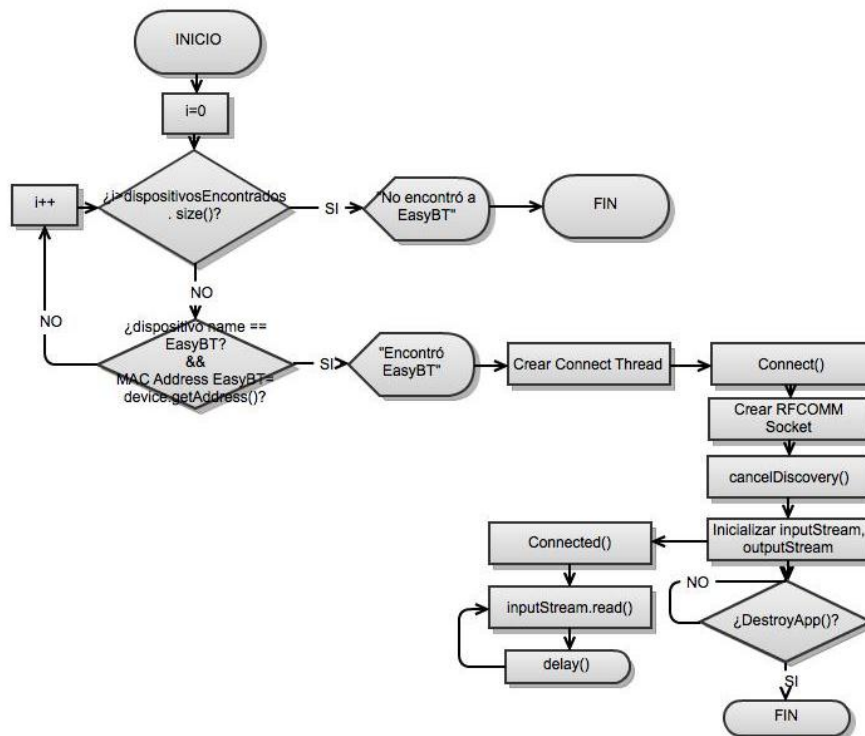
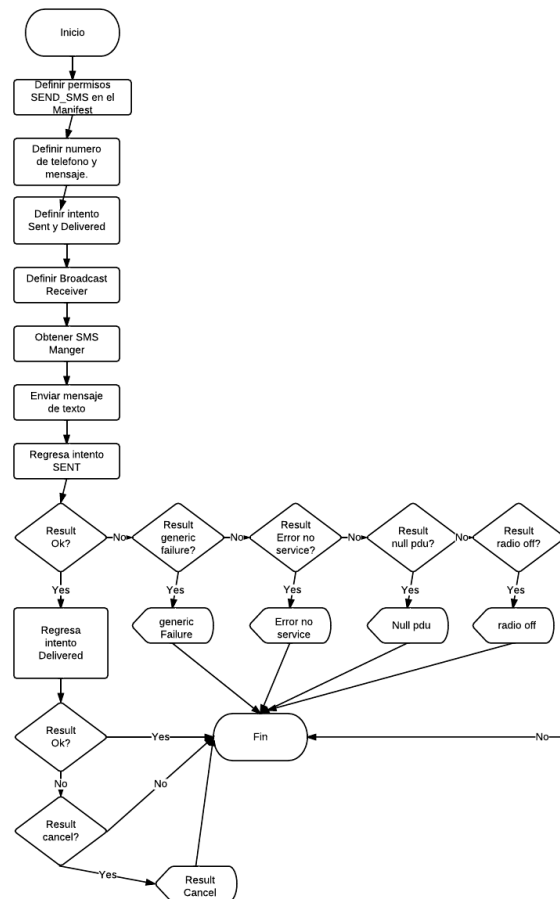


Figura 42 Diagrama de flujo conexión Bluetooth con Módulo Central de Información



b. Envío de mensajes SMS. Para enviar un mensaje de texto utilizando la librería `android.telephony.SmsManager`. Ésta necesita implementar algunos métodos como se puede ver en la Figura 44.

Figura 43 Diagrama de flujo Envío de mensajes

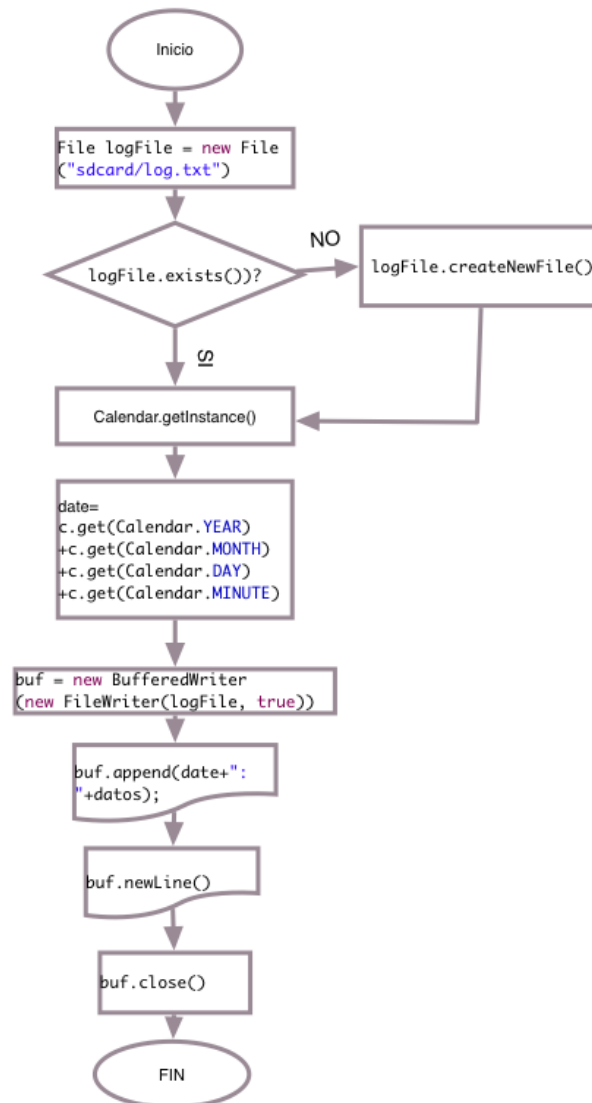


Se definió que el mensaje de texto debe indicar por cada una de los valores obtenidos de los sensores lo siguiente:

1. ¿Ésta es una nueva medición?
2. ¿La medición se encuentra dentro de los parámetros normales?
3. ¿Cuáles el valor?

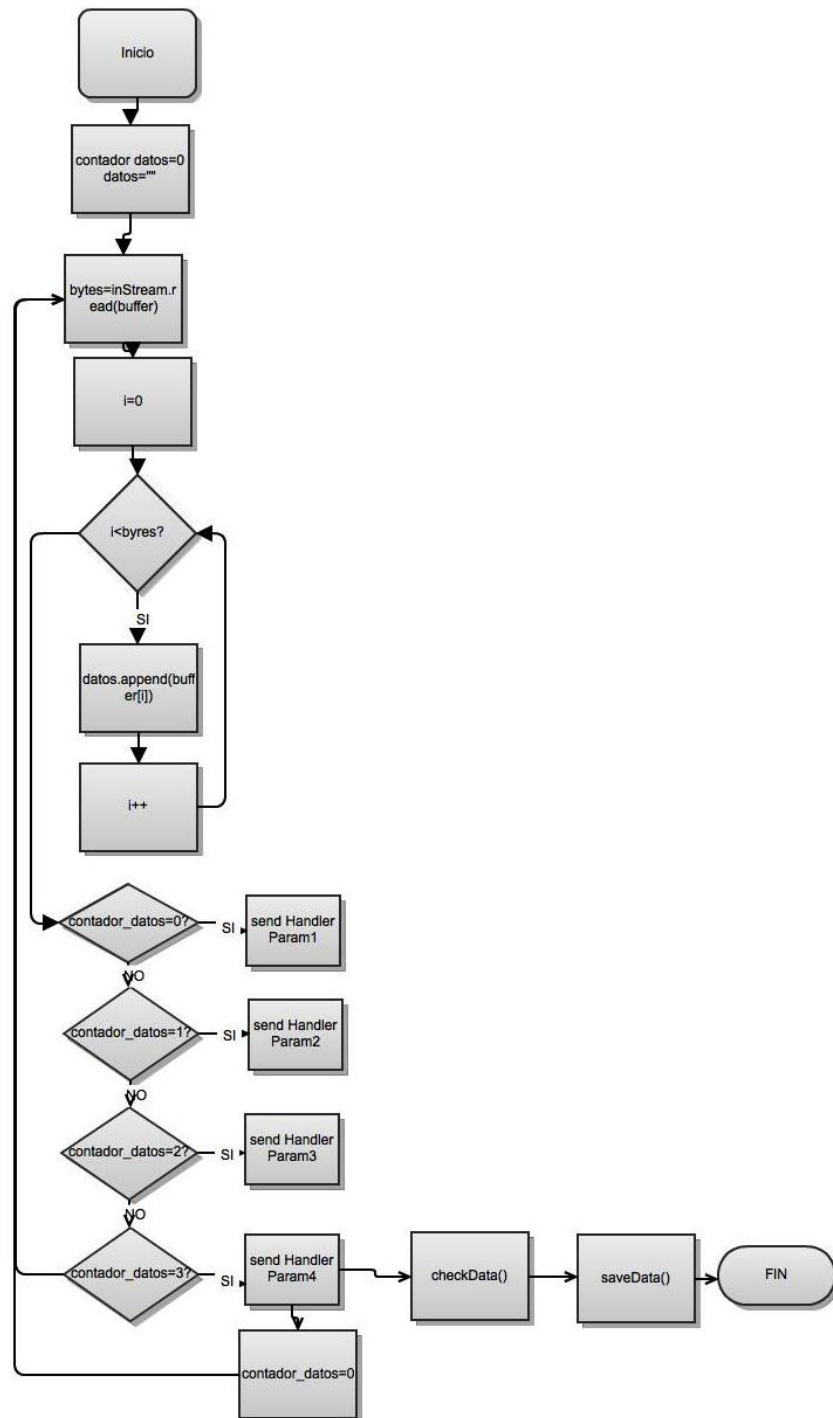
c. Almacenamiento de datos tarjeta micro SD. Para implementar la funcionalidad de almacenamiento de datos en la tarjeta micro SD del teléfono celular se utilizó la clase `FileWriter()`, en la siguiente Figura 45. Se puede ver el diagrama de flujo de esta funcionalidad.

Figura 44 Diagrama de flujo almacenamiento de mediciones en tarjeta micro SD



d. Recepción de datos del Módulo Central de Comunicación. Al recibir los datos del módulo Central de Información es necesario procesarlos y analizarlos poder hacer uso de ellos. Para esto se sigue el procedimiento mostrado en el diagrama de flujo de la Figura 46.

Figura 45 Diagrama de flujo recepción de mediciones en Módulo Central de Monitoreo

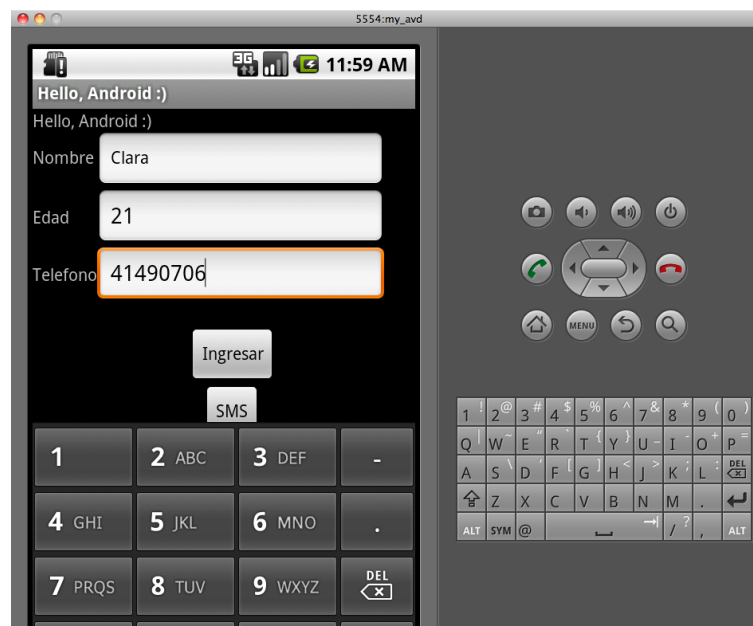


e. Comparación de mediciones de signos vitales. Ya que se tienen las mediciones de los signos vitales del paciente en la aplicación del teléfono celular, es necesario comparar si estos se encuentran dentro de los rangos normales, para definir el contenido del mensaje de texto a enviar.

2. Resultados

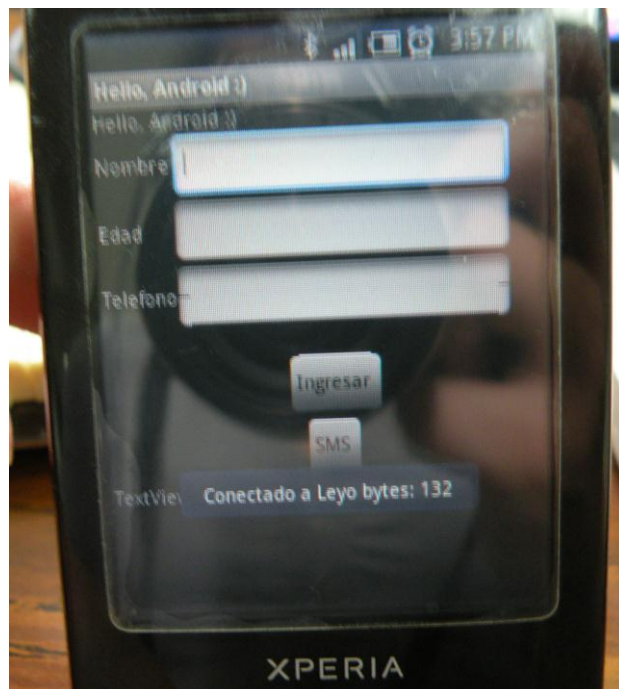
a. Interfaz de usuario. El primer paso para el desarrollo de la aplicación Android fue la interfaz de usuario como se puede ver en la Figura 47 pide al usuario que ingrese nombre del paciente, su estatura y el teléfono celular al que se van a enviar los mensajes de texto. Se agregó un botón para el inicio de la conexión Bluetooth con el módulo Central de Información.

Figura 46 Interfaz de usuario de aplicación Android ejecutada en emulador



b. Conexión Bluetooth. Se llevó a cabo una investigación profunda sobre programación de aplicaciones para la plataforma Android, la funcionalidad Bluetooth y sobre cómo realizar una conexión desde el lado del cliente. Luego se diseñó y desarrolló la aplicación para conectar el módulo central de monitoreo con el teléfono celular vía Bluetooth como se puede ver en la Figura 48. En este diseño la aplicación del teléfono celular actúa como cliente. Esta aplicación revisa si el dispositivo soporta el servicio Bluetooth, lo activa, realiza una búsqueda de los dispositivos cercanos con Bluetooth activado, obtiene sus nombres y MAC address. Y a partir de la búsqueda inicia conexión para intercambio de datos con el dispositivo EasyBT el cual tiene la dirección: 00:17:A0:01:63:18.

Figura 47 Aplicación Android Bluetooth funcionando en teléfono celular

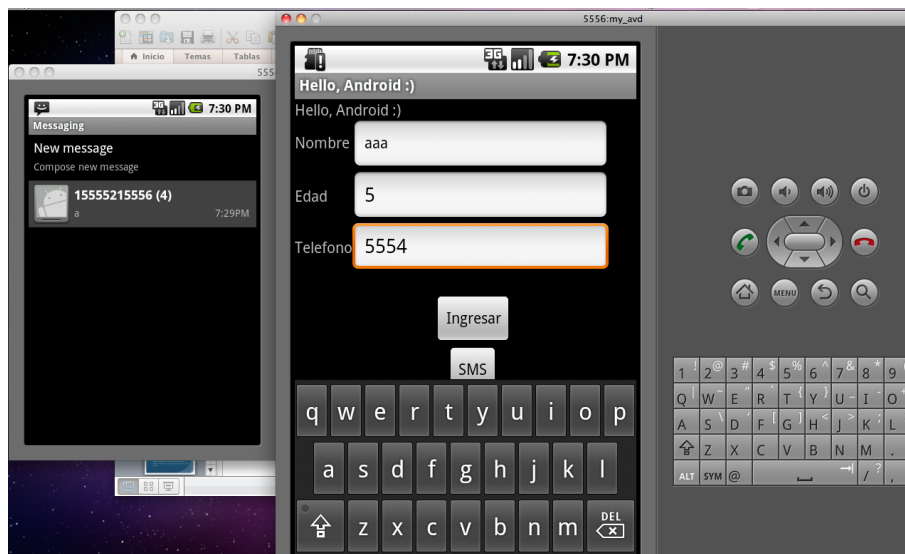


La aplicación del teléfono celular funcionando como cliente, realiza una búsqueda de los dispositivos disponibles, al momento de encontrar el modulo de monitoreo inicia el proceso para realizar la conexión Bluetooth. Cuando ya están conectados empieza a recibir datos, desplegarlos en pantalla como se puede ver en la Figura 48 y regresarlos al Módulo de Monitoreo.

Para esta aplicación es necesario crear un thread que se encargue principalmente de los flujos de datos de la conexión Bluetooth. Es importante que estos métodos de read y write estén en un thread separado pues ambos son métodos que bloquean el flujo del programa.

c. Envío de mensaje SMS. Luego de establecer comunicación Bluetooth con el Módulo de Monitoreo se prosiguió a implementar funcionalidad de envío de mensajes de texto. Se realizó una investigación profunda sobre Android telephony Sms Manager, la librería de Android que permite enviar mensajes de texto. Se logró diseñar y desarrollar la aplicación que envía el mensaje de texto al número de teléfono del médico y se realizaron pruebas de la aplicación en el emulador en la computadora y en el teléfono celular como se puede ver en la Figura 49.

Figura 48 Envío de mensajes de texto funcionando en simulador Android



3. Almacenamiento de datos tarjeta micro SD. Luego de establecer una comunicación Bluetooth con el Módulo Central de Monitoreo y de definir el método que envía el mensaje de texto, se prosiguió a diseñar el método que almacena los datos de las mediciones para que se pueda llevar un control del paciente.

Estos datos se van almacenando en un archivo .txt en la tarjeta micro SD del teléfono celular. Se almacena la fecha y hora, las cuales se obtienen utilizando la librería Calendar de Android. También se almacenan los valores de las mediciones y si éstas están dentro o fuera de los rangos normales.

4. Comparación de mediciones de signos vitales. La aplicación del teléfono celular recibe del Módulo de Monitoreo las mediciones de los signos vitales del paciente. Se quiere enviar un mensaje de texto que contenga información relevante sobre las mediciones.

El Módulo de Monitoreo pide a cada uno de los sensores que envíen la última medición realizada. Esta medición se envía al teléfono celular en donde se define el mensaje de texto a enviar al médico.

Para poder determinar si se tiene una nueva medición en el glucómetro, es necesario comparar la hora actual con la hora de la última medición. Si la última medición se realizó hace menos de 10 minutos entonces se considera que no existe una nueva medición. Para el resto de sensores se revisa el valor obtenido, si éste es 255 indica que no existe una nueva medición.

D. Discusión

Originalmente la aplicación del teléfono celular se diseñó para la plataforma J2ME, sin embargo a lo largo del camino surgieron varios problemas de compatibilidad. Fue por esto que se decidió desarrollar una nueva aplicación para plataforma Android para que la aplicación se adapte a la nueva tecnología de Smartphones y que ésta pueda ser ejecutada en la mayor cantidad de dispositivos.

Se desarrolló la aplicación para la plataforma Android 2.1, a lo largo del desarrollo se dieron varios problemas con la funcionalidad Bluetooth, se lograba establecer la comunicación pero luego de aproximadamente un minuto esta se perdía. Uno de estos problemas se dio debido al formato de la dirección del módulo de Bluetooth EasyBT, el cual es diferente en cada librería utilizada.

A lo largo del desarrollo del megaproyecto se pudo ver que las mediciones de signos vitales y su análisis podrían servir no sólo para enviar un mensaje de texto al médico sino que también para proporcionar un registro de los datos del paciente, de esta forma el médico, en caso de que sea necesario podría revisar el historial de la condición de salud del paciente. Fue por esto que se definió que era importante almacenar un registro de los datos de las mediciones del paciente por lo que se desarrolló la funcionalidad de almacenamiento de datos obtenidos del Módulo Central de Monitoreo. Estos datos se almacenan en un archivo en la tarjeta micro SD del teléfono celular. Se almacena la fecha y hora las cuales se obtienen utilizando la librería Calendar de Android. También se almacenan los valores de las mediciones y si éstas están dentro o fuera de los rangos normales.

En la aplicación original, se definió que el Módulo Central de Información se iba a encargar de comparar los datos de las mediciones de signos vitales, sin embargo debido a la facilidad de comparar datos en la aplicación del teléfono celular se decidió que la aplicación se va a encargar de la comparación de los datos.

Se definió el contenido de los mensajes de texto, tomando en cuenta que las mediciones de signos vitales obtenidas del Módulo Central de Información pueden ser mediciones recientes o mediciones tomadas un tiempo atrás y que estas pueden estar dentro o fuera de los parámetros.

Ya que en la aplicación del teléfono celular se van a realizar las comparaciones para definir si las mediciones de los signos vitales se encuentren dentro de los parámetros normales fue necesario realizar investigación sobre cuáles son estos valores. Se consultaron varias fuentes de sitios médicos para poder definir los rangos normales de las mediciones de los signos vitales. No todas las fuentes coincidían exactamente con los valores, sin embargo estos tenían coherencia entre ellos, con base en estos datos se definieron rangos normales de mediciones. Estos rangos se utilizaron para analizar las mediciones y definir el contenido del mensaje de texto que se le va a enviar al médico.

Finalmente se logró desarrollar la aplicación para la plataforma Android 2.1, ésta establece una comunicación vía Bluetooth con el módulo Central de Información del cual obtiene las mediciones de los signos vitales del paciente, las analiza, define y envía un mensaje de texto al médico con información sobre estos. Además almacena un registro de las mediciones en la tarjeta micro SD del teléfono. Esta aplicación podría adaptarse para funcionar con otros sensores, al incrementar el número de parámetros que se reciben del Módulo Central de Información.

Buscando compatibilidad se optó por desarrollar la aplicación para la plataforma Android con el objetivo de que ésta se pudiera ejecutar en la mayor cantidad de dispositivos posible. Sin embargo al finalizar la aplicación, se

realizaron pruebas en distintos dispositivos de diferentes marcas, todos con la plataforma Android 2.1 y como se puede ver en la Figura 50. No se logró cumplir el objetivo de compatibilidad. Con estos resultados se llegó a la conclusión que la aplicación solo se puede ejecutar en dispositivos Sony Xperia. Esto se debe a la funcionalidad Bluetooth la cual tiene especificaciones diferentes para cada marca. Para corregir este problema y permitir que la aplicación sea compatible con la mayor cantidad de dispositivos es necesario investigar sobre la funcionalidad Bluetooth en cada marca, ver las especificaciones y como desarrollar aplicaciones Bluetooth que cumplan con ellas. Luego con esto desarrollar una aplicación más robusta que tome en cuenta el hardware del dispositivo.

Figura 49 Pruebas aplicación Android en diferentes modelos

Teléfono celular	Modelo	Conexión Bluetooth	SMS
0	Sony Ericsson Xperia U20a	✓	✓
1	Sony Ericsson GT-I5500L	✓	✓
2	Sony Ericsson Xperia A853	✓	✓
3	Samsung GT-I5500L	✗	✗
4	Samsung GT-I5500L	✗	✗
5	Motorola A853	✗	✗

XIX. APÉNDICE C. Sistema de Medición de Temperatura, Glucosa y Presión Arterial

A. Metodología general

Para alcanzar el objetivo de implementar los medidores capaces de adquirir la concentración de glucosa en la sangre, presión arterial y temperatura corporal, se procedió de la siguiente manera:

1. Se recolectó información en medios electrónicos referente a las soluciones comerciales que existen actualmente para la adquisición de los signos vitales propuestos.

2. Se seleccionó e hizo efectiva la adquisición de los medidores comerciales a utilizar en el módulo, tomando en cuenta criterios que incluyen: confiabilidad de las mediciones, precio del dispositivo, comodidad en su uso por parte del usuario, así como capacidad de almacenamiento y de comunicación con otros dispositivos.

3. En el caso del glucómetro y esfigmomanómetro seleccionados, se desmontó la placa electrónica principal de los dispositivos, de tal manera que se tuviera acceso a los códigos de identificación de los circuitos integrados que la conforman.

4. Se consultó en referencias electrónicas los códigos de identificación de los circuitos integrados encontrados en el glucómetro y esfigmomanómetro, de tal manera que se comprendiera la función que estos cumplen en la operación del medidor.

5. Se procedió a la intervención de las señales presentes en los componentes de interés utilizando los instrumentos del laboratorio electrónico, así como el diseño de circuitos especializados.

6. Se tabularon señales y paquetes de datos encontrados durante las mediciones, que con ayuda de las fuentes consultadas y el análisis de estas señales, permitieron identificar la estructura del manejo de los datos adquiridos por cada uno de los instrumentos. Para la medición de la concentración de glucosa en la sangre y presión arterial, se implementó un circuito diseñado de tal manera que éste se pueda integrar al instrumento sin afectar su funcionamiento

normal, y al mismo tiempo cumplir el protocolo de comunicación impuesto por la unidad Central de Información.

7. Se diseñó e implementó un termómetro, utilizando el sensor de temperatura seleccionado, capaz de adquirir la temperatura corporal de manera oral, buscando la comodidad del mismo durante su uso.

8. Se evaluó la confiabilidad de los datos adquiridos mediante la comparación de los resultados obtenidos contra las mediciones de control realizadas por dispositivos no intervenidos.

B. Esfigmomanómetro

1. Diseño experimental

a. Se consultó la hoja de datos de la memoria EEPROM HT24LC16(Holtek Semiconductors Inc, 2003) para determinar su protocolo de comunicación y características generales.

b. Se seleccionó el PIC16F887(Microchip Inc., 2011)para implementar el protocolo I2C(Philips Semiconductors, 1995) por medio del cual es posible establecer comunicación con la memoria EEPROM HT24LC16 integrada en el esfigmomanómetro ADC 6015.

c. Se realizaron mediciones de presión arterial con el esfigmomanómetro ADC 6015, en cada una de las cuales, utilizando el osciloscopio, se observó las señales que se presentan en cada uno de los pines de la memoria EEPROM HT24LC16, mediante cables soldados a esta como se muestra en la Figura 51.

d. A partir de las señales obtenidas en los pines de la memoria HT24LC16, se diseñó un circuito y se implementó un código en lenguaje C, utilizando el compilador MikroC(Mikroelektronica Inc, 2011), en el PIC16F887, capaz de leer 20localidades de la memoria HT24LC16, de tal manera que se encontrara un patrón en el almacenamiento de los datos

e. Se implementó un circuito y un código en lenguaje C, utilizando el compilador MikroC, en el PIC16F887, capaz de identificar que una nueva medición fue realizada y almacenada en memoria, para luego adquirirla y enviarla a una pantalla LCD para su comparación.

f. Se realizaron 3 mediciones con el esfigmomanómetro para verificar la confiabilidad del sistema en la adquisición automática de la información durante cada medición, siendo estas comparadas con las adquiridas por el esfigmomanómetro digital de muñeca LifeSource UB-512(HealthChek Systems, 2011), espaciando ambas mediciones en 5 minutos en cada paciente, de manera que la presión se normalice en cada caso. Así mismo, en cada medición se realizó la primera adquisición con el dispositivo de referencia.

2. Resultados. Se encontró que la memoria EEPROM HT24LC16 con que cuenta el dispositivo es una memoria EEPROM serial que posee un protocolo de comunicación I2C y una capacidad de almacenamiento de 16K (Holtek Semiconductors Inc, 2003). A continuación, en la Figura 51, se presenta el diagrama de pines de dicho componente.

Figura 50: Intervención de memoria HT24LC16 del esfigmomanómetro.

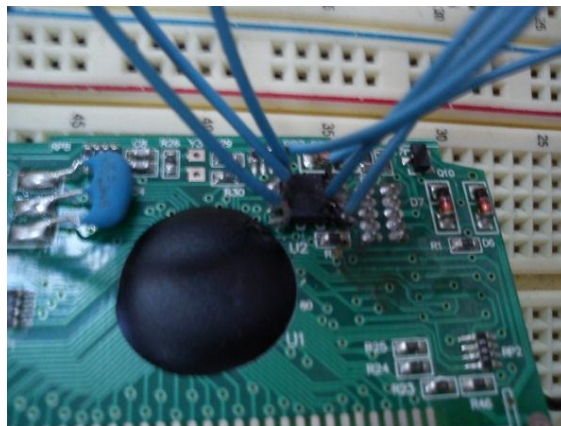
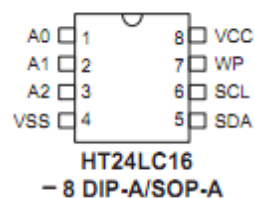


Figura 51: Diagrama de memoria HT24LC16

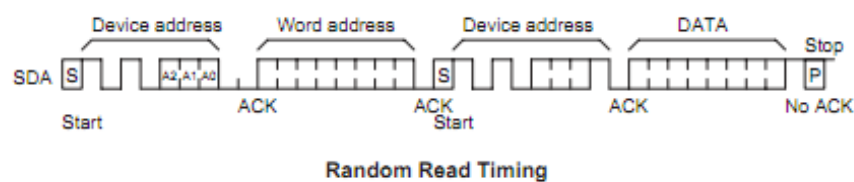


Para leer una localidad de memoria es necesario alimentar el integrado entre 2.2 y 5.5 V en el pin VCC. Así mismo los pines A0, A1 y A2 representan la dirección del dispositivo en el bus del protocolo I²C. Por otro lado los pines SCL y SDA, que corresponden al reloj y al bus de datos bidireccional, respectivamente, deben ser conectados a “Pull Ups”, de acuerdo a la topología del bus I²C.

Finalmente el pin WP (“Write Protected”), no tiene función en modo lectura(Holtek Semiconductors Inc, 2003).

Como parte del protocolo de comunicación, en modo lectura aleatoria, es necesario el envío de una condición de inicio, seguido de la dirección del dispositivo en el bus y el octavo bit, el de escritura o lectura, desactivado, con lo cual la memoria responderá con una señal de recibido (0) en el noveno pulso de reloj. Luego se debe enviar la dirección de memoria a leerse, con lo cual la memoria volverá a responder con una señal de recibido. Posteriormente debe enviarse una nueva condición de inicio, seguida de por la dirección del dispositivo con el bit de escritura y lectura activado, ante esto la memoria responderá de nuevo con una señal de recibido, seguida del dato en dicha localidad. Finalmente el maestro deberá enviar una condición de finalización(Holtek Semiconductors Inc, 2003), como se muestra a continuación en la Figura 53.

Figura 52: Secuencia para acceso aleatorio de la memoria HT24LC16.



Al estudiar las señales que se presentaban en los pines de la memoria EEPROM HT24LC16 durante cada medición, se encontró que mientras el esfigmomanómetro se encuentra en un estado de espera (con baterías pero sin realizar mediciones), el voltaje de alimentación es retirado de la memoria, como se muestra en la Figura. 16. No obstante, cuando el botón “Start” es presionado y la medición se lleva a cabo, la memoria es alimentada con 4V, en el pin etiquetado como VCC en la Figura. 14. Así mismo, los pines SDA, SCL y WP son puestos en alto durante ese mismo período, como se muestra en la Figura. 54. Cabe mencionar que este estado en alto del pin WP indica que la memoria se encuentra protegida contra escritura, mas no contra lectura.

Figura 53: Señal en pin de alimentación en memoria HT24LC16 en modo de espera del esfigmomanómetro.

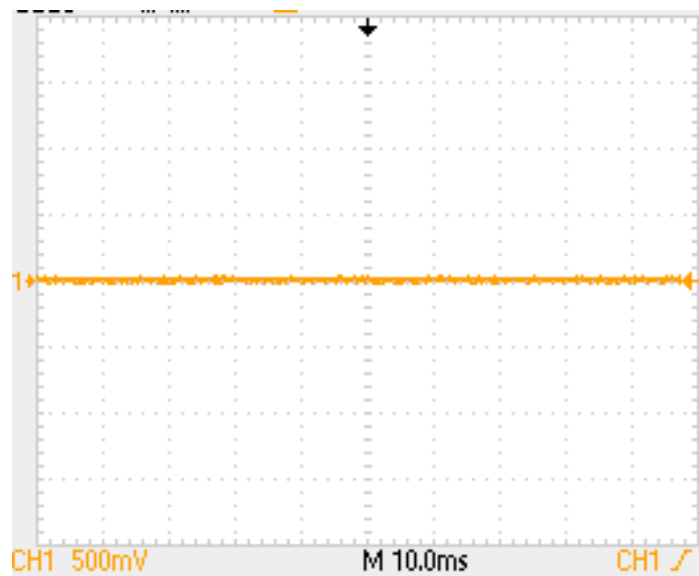
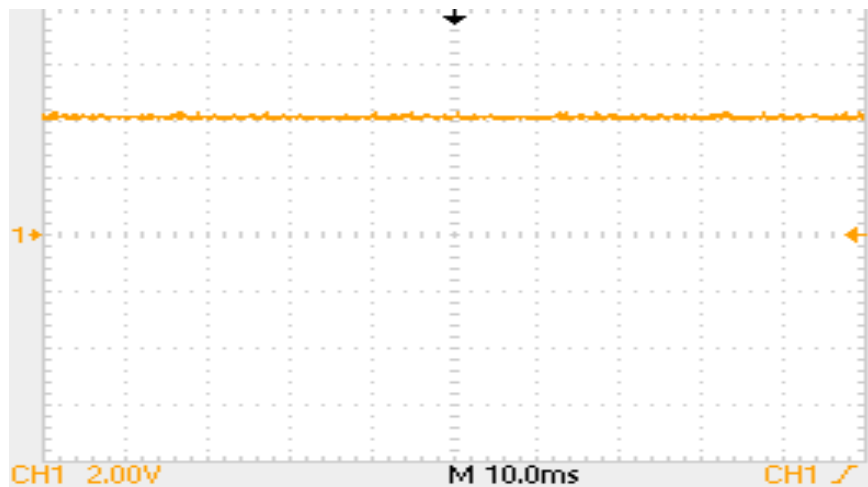


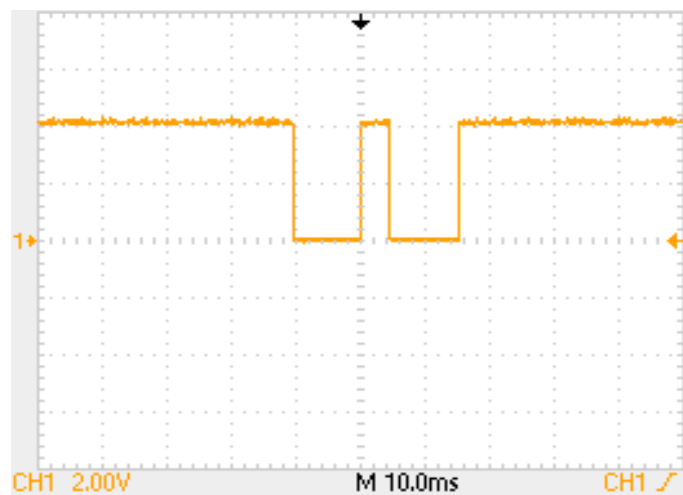
Figura 54: Señal en pin WP de la memoria HT24LC16 en estado activo del esfigmomanómetro.



Se encontró que cuando la medición es concluida por el dispositivo, inmediatamente el pin de WP es puesto en bajo durante un dos periodos de 10.40 ms cada uno, como se observa en la Figura 56. Durante este tiempo en el que el pin WP se mantiene en bajo, la memoria no se encuentra protegida contra escritura y por lo tanto es durante este periodo en el cual a través del pin SDA se

envía la información a ser almacenada en memoria, siguiendo el protocolo I²C. Finalmente, luego que la información ha sido enviada, la memoria es protegida contra escritura, quedándose la memoria alimentada durante aproximadamente 15 segundos, siempre y cuando el botón “Start” no sea presionado, para luego desactivarse de manera automática hasta que otra medición sea realizada. De esta forma, es durante el transcurso de este periodo de tiempo, en el cual se procede a realizar la adquisición de los datos almacenados en la memoria, aprovechando el hecho que la memoria se encuentra alimentada.

Figura 55: Señal en pin WP en memoria HT24LC16 durante almacenamiento de lectura.



A partir del comportamiento de las señales en los pines de la memoria HT24LC16 descritas previamente, y mediante la implementación de un código en MikroC capaz de establecer comunicación con la memoria, de acuerdo al protocolo I²C, fue posible registrar los datos almacenados en las primeras 20 localidades, las cuales se presentan en la Figura 57.

Figura 56 Datos adquiridos de memoria HT24LC16

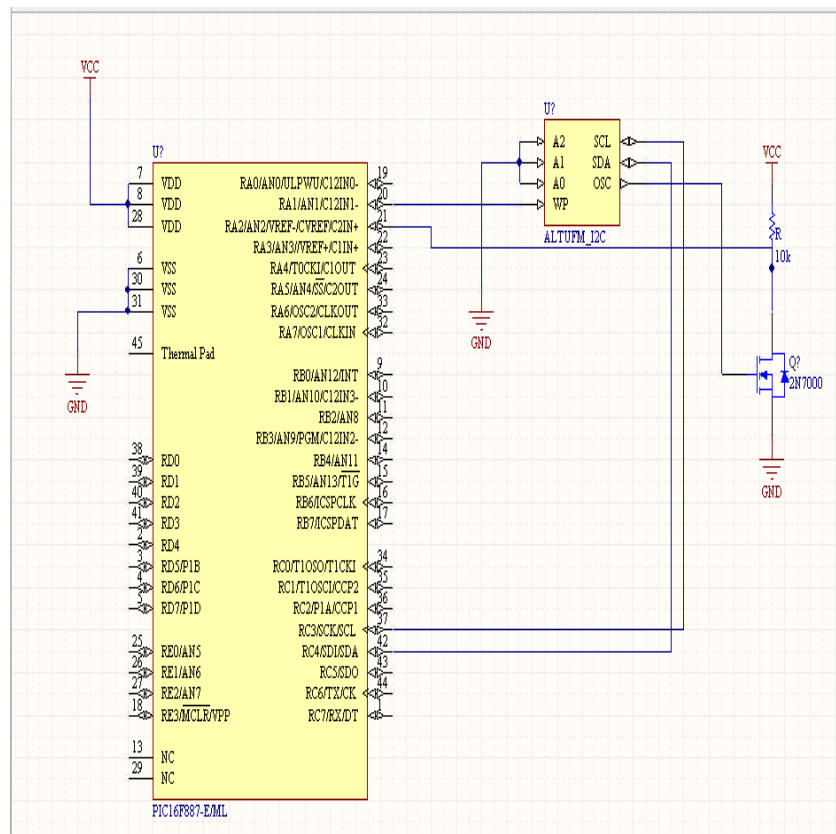
Localidad en memoria HT24LC16	Dato adquirido
0	59
1	99
2	32
3	32
4	32
5	32
6	0
7	30
8	169
9	107
10	86
11	65
12	0
13	1
14	3
15	175
16	102
17	63
18	56
19	65
20	0

En la Figura 57 se puede observar que en la localidad de memoria 0 se almacena el número de mediciones almacenadas, siendo 59 el máximo cuando la memoria se encuentra llena. Por otro lado, se encontró que en la localidad 9 se almacena la presión sistólica más antigua adquirida, mientras que en la localidad 10 y 11 se encuentran la presión diastólica y al pulso cardíaco, respectivamente, de dicha medición. Finalmente, se observa que este patrón vuelve a repetirse 4 localidades después, es decir, en la localidad 16. Cabe resaltar que las presiones son almacenadas en localidades de 8 bits, siendo su valor decimal el valor de la presión sistólica, cardíaca y pulso cardíaco adquiridos. Así mismo, se encontró

que la estructura de almacenamiento es del tipo FIFO, es decir, las mediciones más recientes son almacenadas en las localidades más altas, desplazando hacia abajo en la pila las mediciones más antiguas.

Con la comprensión de la estructura de almacenamiento en memoria por parte del dispositivo, así como del comportamiento de las señales en los pines de la memoria, fue posible la integración de un circuito capaz de la adquisición automática de la última medición realizada por el dispositivo. En la Figura 57 se muestra el circuito implementado, en el cual el microcontrolador PIC16F887 se encarga de la detección del almacenamiento en memoria mediante la detección de la desactivación del pin de WP de la memoria durante el almacenamiento, siendo necesario el monitoreo del pin de alimentación de la memoria para evitar que la desactivación de pin WP que se dé cuando el dispositivo entra en modo de espera sea detectado como un almacenamiento en memoria. A partir de la detección de esta condición, el microcontrolador establece comunicación con la memoria por medio del bus I²C que el esfigmomanómetro implementa.

Figura 57: Circuito implementado para detección de condición de almacenamiento en memoria del esfigmomanómetro y lectura de última medición realizada.



De esta forma, cuando una medición es almacenada, el código implementado accede al dato almacenado en la localidad 0 de la memoria, de tal manera que se establece el número de memorias utilizadas, y el cual al ser multiplicado por 8, permite la adquisición de la última presión sistólica medida, para luego, de manera secuencial, adquirir la diastólica y pulso cardíaco. En la Figura. 59 y 60, se observa que el circuito implementado cumple con la función de su diseño.

Figura 58: Presión sistólica, diastólica y pulso cardíaco adquiridos.

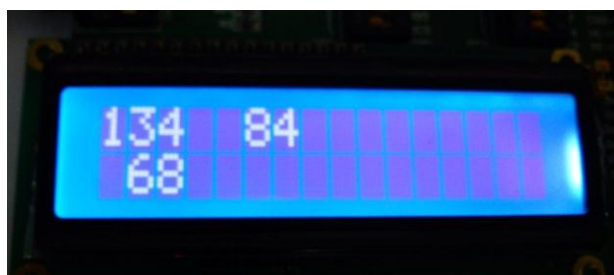


Figura 59: Medición almacenada en memoria de esfigmomanómetro ADC 6015



Figura 60: Medición de exactitud en medición de presión sistólica del esfigmomanómetro invadido.

Individuo	Presión sistólica referencia [mmHg]	Presión sistólica [mmHg]	% Error
1	116	118	1.72
2	107	110	2.8
3	121	123	1.65

Figura 61 Medición de exactitud en medición de presión diastólica del esfigmomanómetro invadido.

Individuo	Presión sistólica referencia [mmHg]	Presión sistólica [mmHg]	% Error
1	73	69	5.48
2	71	70	1.41
3	83	81	2.41

Figura 62 Medición de exactitud en la medición de pulso cardíaco del esfigmomanómetro invadido

Individuo	Pulso Cardíaco referencia [Latidos por Minuto]	Pulso cardíaco [Latidos por Minuto]	% Error
1	58	60	3.45
2	70	76	8.57
3	69	72	4.35

Al comparar las mediciones realizadas por el esfigmomanómetro ADC 6015 invadido con las obtenidas con el esfigmomanómetro LifeSource UB-512, se encontró, como se observa en la Figura 61, Figura 62 y Figura 63, que el instrumento invadido sigue presentando mediciones adecuadas de presión sistólica, diastólica y pulso cardíaco, encontrándose el mayor porcentaje de error en la detección del pulso cardíaco, siendo este de hasta 8.57%. Por su parte, el mayor porcentaje de error en la medición de presión diastólica y sistólica fue de 2.8% y 5.48%, respectivamente.

C. Glucómetro

1. Diseño experimental

a. Se estableció comunicación infrarroja entre la PDA Tungsten E2 (Palm Inc, 2011) y el glucómetro Accu-Chek Aviva y se descargaron las mediciones realizadas para observar características del protocolo, como el tiempo de descarga de datos.

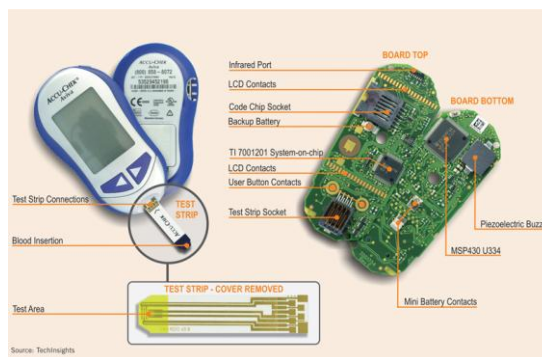
b. Utilizando el transceptor infrarrojo TFDU 4101 (Vishay Inc, 2011) y el microcontrolador PIC24FJ64GB002 (Microchip Inc, 2011) se implementó un analizador de paquetes capaz de capturar la secuencia de datos intercambiados de manera infrarroja durante la conexión entre la PDA y el glucómetro, siguiendo las características físicas que el estándar IrDA (Infrared Data Association, 2011) impone.

c. A partir de la interpretación de la secuencia encontrada, se implementó un circuito utilizando el PIC24FJ64GB002 capaz de establecer comunicación infrarroja con el glucómetro para descargar las mediciones almacenadas en la memoria y enviarlas por medio del protocolo serial a una computadora.

d. Se implementó un circuito capaz de adquirir de manera automática las mediciones realizadas por el glucómetro cuando una secuencia fuera enviada de manera serial desde una computadora.

2. Resultados. A diferencia de esfigmomanómetro digital ADC 6015, al desmontar la placa principal del circuito se encuentra una mayor integración de los componentes, como se observa en la Figura 64.

Figura 63: Circuito principal glucómetro Accu-Check Aviva.



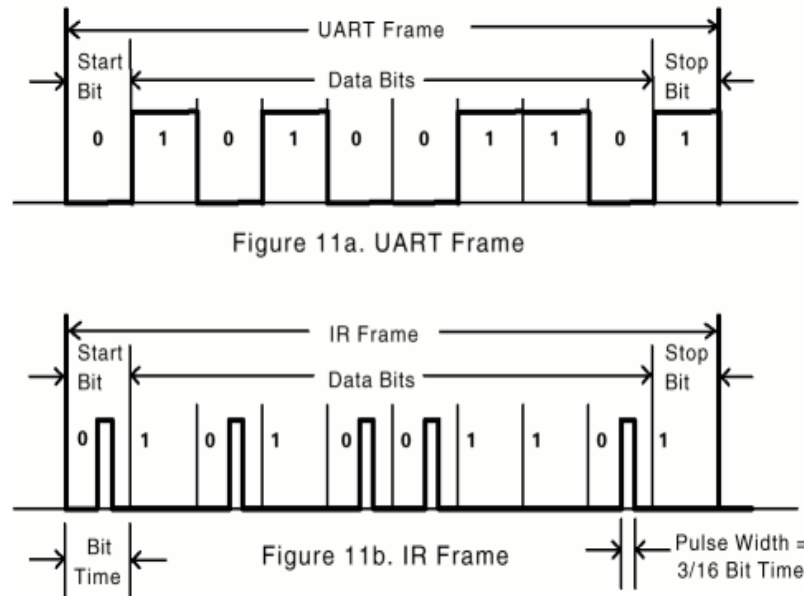
De donde se resaltan los componentes MSP430 U334 y el TI7001201, siendo el primero el microcontrolador del dispositivo, y el segundo el amperímetro encargado de la medición de la corriente inducida por la glucosa en la sangre de la muestra. A diferencia del esfigmomanómetro, no es posible encontrar una integrado dedicado al almacenamiento de las mediciones, siendo estas almacenadas en la memoria interna del microcontrolador MSP430 U334 (Bitton, 2011). De esta manera, resultó necesario el aprovechamiento que el dispositivo ofrece en el uso de un puerto infrarrojo para la adquisición de sus mediciones.

Para la descarga de las mediciones almacenadas en la memoria del dispositivo mediante la comunicación infrarroja con la PDA Tugsten E2, fue necesaria la adquisición del software para dicho fin, siendo éste el AccuChekCompass Pocket 3.0 (Roche Diagnostics, 2010). Una vez instalado dicho software, y al tratar de establecer comunicación con el glucómetro, se encontró la necesidad de que tanto la hora como la fecha se encuentren sincronizados entre ambos dispositivos, de lo contrario la comunicación se ve interrumpida. Al inicializarla comunicación, el PDA permanecerá buscando dispositivos hasta que encuentra al dispositivo, indicando la hora y fecha correspondiente a la última sincronización con el dispositivo. Una vez la transferencia de datos es finalizada, el glucómetro es desactivado automáticamente, quedando los datos registrados en la PDA, donde estos pueden ser manipulados de diferentes formas.

En cuanto a la implementación del analizador de paquetes transferidos de manera infrarroja se refiere, se implementó el circuito que se muestra en la Figura 65, en la cual se utilizó el PIC24FJ64GB002 y el transceptor TFDU 4101.

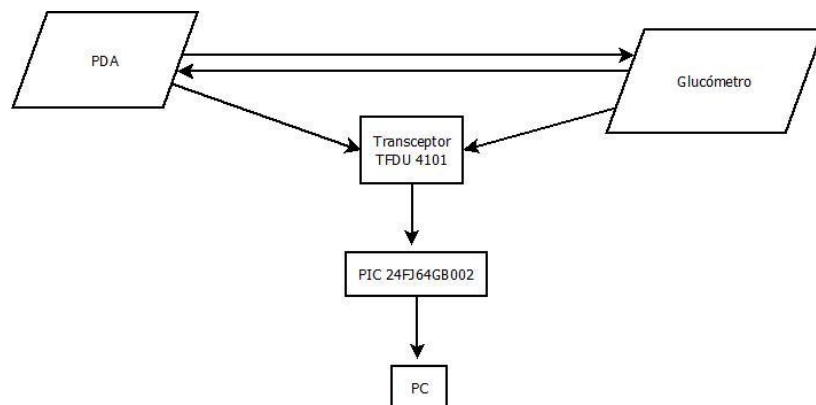
consumo de energía durante la transmisión de información, el estándar IrDA utiliza un ancho de pulso que equivale a $3/16$ del ancho de pulso del UART, aparte de implementar una lógica invertida (MySlik, 2003), como se muestra en la Figura 66

Figura 65: Comparación entre ancho de pulso UART e IrDA.



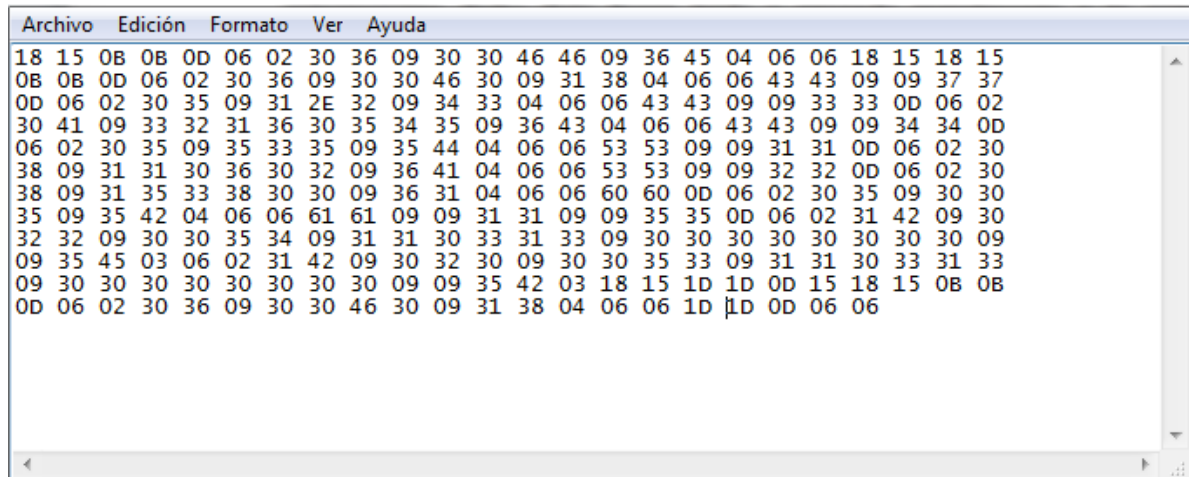
Así mismo se debe resaltar, que para la implementación de este analizador de paquetes, se inició con una tasa de transferencia de 9600 baudios con un solo bit de paro y ninguna paridad, siendo esta misma configuración aplicada para el módulo UART2 para el envío de los datos recolectados a la computadora. De esta manera en la Figura 67 se presenta la topología implementada.

Figura 66: Topología para la captura del protocolo entre glucómetro y PDA.



Una vez implementado el circuito, y tras la captura de los datos transferidos entre la PDA y el glucómetro, se encontró que el protocolo de comunicación es el que se muestra en la Figura 68, siendo esta su representación hexadecimal. Es de mencionar que en la Figura 68 no se muestra el “0x” antes de cada byte adquirido, aun cuando estos son valores hexadecimales.

Figura 67: Datos adquiridos con el analizador de paquetes infrarrojos implementado.



De esta manera, al observar las secuencias de bytes registrados, y compararlos con el valor hexadecimal del código ASCII, es posible determinar un patrón en los datos. A continuación, en la Figura 69, se muestran algunos de los caracteres que corresponden al valor hexadecimal recibido, y que suponen una estructura del protocolo.

Figura68: Representación ASCII de datos capturados del protocolo infrarrojo implementado por glucómetro.

Valor Hexadecimal	Carácter ASCII
0x06	Positive Acknowledge
0x02	Start of String
0x04	End of transmission
0x09	Horizontal Tab
0x18	Cancel
0x15	NegativeAcknowledge
0x0D	Vertical Tab

De esta forma en la secuencia registrada, se puede concluir que luego de cada instrucción enviada por la PDA, donde el valor 0x0D equivale a un introducir, se tiene una respuesta positiva, 0x06, por el glucómetro, seguida de un encabezado, 0x02, que indica el inicio de una secuencia que culminará con un 0x04, equivalente a un final de transmisión. Finalmente la PDA responderá con una afirmación positiva si esta ha recibido la información. Es de notar que durante la transmisión desde la PDA de la secuencia de caracteres que conforman una instrucción, el glucómetro responde con el eco de dicho carácter enviado.

Con el apoyo del código encontrado en línea para la descarga de los datos en un glucómetro del mismo fabricante, pero diferente modelo (American Telemedicine Association, 2011), se determinó que la secuencia: 0x60, 0x0D, solicita al glucómetro la cantidad de mediciones almacenadas en memoria, ante lo cual el glucómetro responde con la siguiente secuencia: 0x06, 0x02, 0x30, 0x35 0x09, 0x30, 0x30, 0x35, 0x09 0x35 0x42, 0x 04. Donde de acuerdo a la estructura planteada previamente, los bytes 6,7 y 8, del patrón anterior, corresponden al número de mediciones almacenadas, en este caso 005, por su valor ASCII.

Así mismo un mismo patrón se encuentra al solicitar al glucómetro las mediciones almacenadas en memoria, en donde la secuencia: 0x61, 0x09, 0x31, 0x09, 0x35, 0x0D, le indica al glucómetro que retorne el dato almacenado en la localidad 5, como el quinto dato de la secuencia anterior indica. Finalmente, como última instrucción encontrada en este protocolo de comunicación, se encuentra la instrucción 0x1D, que indica al glucómetro entrar en modo reposo.

A partir de la decodificación del protocolo antes presentado, se implementó dicha secuencia en el microcontrolador PIC24FJ64GB002 para adquirir las mediciones realizadas por el glucómetro. De nueva cuenta, se utilizó el transceptor TFDU 4101 para el envío y recepción de los datos enviados de manera infrarroja en el módulo UART1 y el envío de los datos adquiridos a la computadora en el módulo UART2, en el circuito que se muestra en la Figura 65. La topología del circuito implementado es la que se aprecia en la Figura 70.

Figura 69: Topología de circuito implementado para la comunicación infrarroja con glucómetro.



De esta manera, como se observa en la Figura. 71, al implementar la secuencia de comandos encontrada para la comunicación infrarroja con el glucómetro, fue posible adquirir la última medición almacenada por el dispositivo encontrándose que en la secuencia de caracteres devuelta por el dispositivo, es en los bytes 6, 7 y 8 de la cadena, en la cual la concentración de azúcar en la sangre medida es presentada, como se aprecia en la Figura 72, donde se muestra la quinta y más reciente medición realizada, correspondiente a 022, por su valor ASCII. Así mismo, cabe mencionar que en la misma cadena de caracteres se indica la hora y fecha en que se realizó la medición, en este caso a las 12:54, el día 3 de noviembre de 2013.

Figura 70: Medición adquirida mediante comunicación infrarroja con glucómetro.

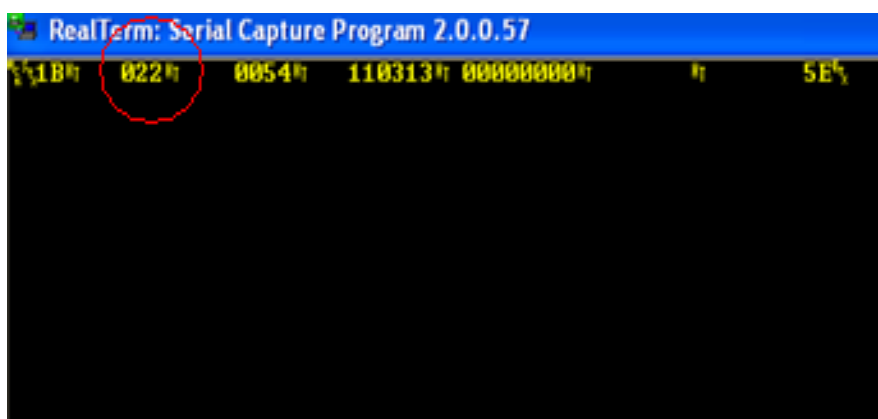


Figura 71: Última medición almacenada en glucómetro.



D. Termómetro

1. Diseño experimental

- a. Se adquirió el termómetro digital comercial Microlife MT3001 (Microlife Inc, 2010) para ser utilizado como referencia.
- b. Se estableció un material de buena conducción térmica capaz de proteger el termómetro digital DS18B20 del contacto directo con los fluidos salivales del paciente.
- c. Se aisló el termómetro digital de la estructura de sujeción del termómetro utilizando silicón.
- d. Se adhirió el termómetro digital DS18B20 a la estructura de sujeción del termómetro diseñado.
- e. Utilizando plástico termoencogible, se garantizó una buena sujeción del termómetro digital, así como un mejor aislamiento térmico del mismo.
- f. Se implementó un código en el microcontrolador PIC16F887, utilizando el compilador MikroC (Mikroelektronika Inc, 2011), para establecer comunicación con el sensor de temperatura de acuerdo al protocolo 1-Wire (Maxim Integrated Products, 2008). Así como para el envío, de manera serial, por medio de RS-232, de la medición obtenida a la computadora para su registro.

g. Se realizaron pruebas de medición de la temperatura oral del paciente utilizando como referencia el termómetro Microlife MT3001.

h. A partir de los anteriores resultados, se procedió a determinar la exactitud y tiempo de adquisición en la medición, del termómetro digital diseñado.

2. Resultados. Ya que el termómetro digital DS18B20 se encuentra en un empaquetado tipo TO-92, se decidió, para la implementación del termómetro oral, protegerlo mediante el recubrimiento del mismo con un material que posea buenas características térmicas, y a la vez ofrezca comodidad al paciente durante su uso. De esta manera, como se muestra en la Figura 73, se encontró que las terminales metálicas utilizadas en los fusibles tipo americano cumplen con el diámetro ideal para el alojamiento del termómetro digital, además de ofrecer una superficie cómoda para el paciente, así como una superficie protegida químicamente contra la corrosión.

Figura 72: Recubrimiento de termómetro digital DS18B20 en termómetro oral implementado.



Cabe resaltar que para mejorar el aislamiento térmico del termómetro, se colocó silicón líquido a la superficie expuesta del sensor, es decir, donde se encuentran las terminales del DS18B20. Finalmente, se soldaron a las terminales los cables necesarios para establecer la comunicación con el microcontrolador.

Para ofrecer una estructura sólida para la integración del sensor y la cubierta implementada, así como para ofrecer una superficie cómoda para el paciente, se utilizó la carcasa de un lapicero, como se observa en la Figura 74. Cabe resaltar que antes de la colocación del DS18B20, se aisló la tapadera superior del lapicero con silicón líquido para reducir el efecto de la temperatura ambiental en las mediciones, así como la colocación de plástico termoencogible para mejorar el soporte y aislamiento del sensor.

Figura 73: Termómetro oral implementado.



En cuanto al código se refiere, se implementó el protocolo de comunicación 1-Wire necesario para la comunicación con el DS18B20. De esta forma el algoritmo implementado, se muestra en la Figura 75, en donde se observa que se realiza una medición cada 10 segundos, siendo esta almacenada y comparada con la próxima medición, si esta no ha cambiado se concluye que la temperatura se ha equilibrado y la medición es mostrada en pantalla, como se observa en la Figura 75.

Figura 74: Algoritmo implementado en termómetro oral diseñado.

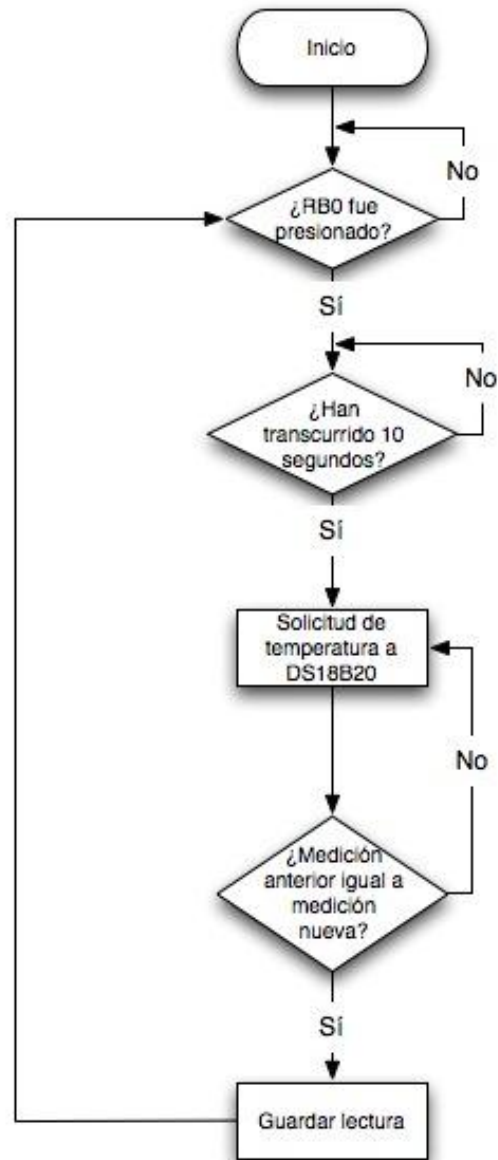
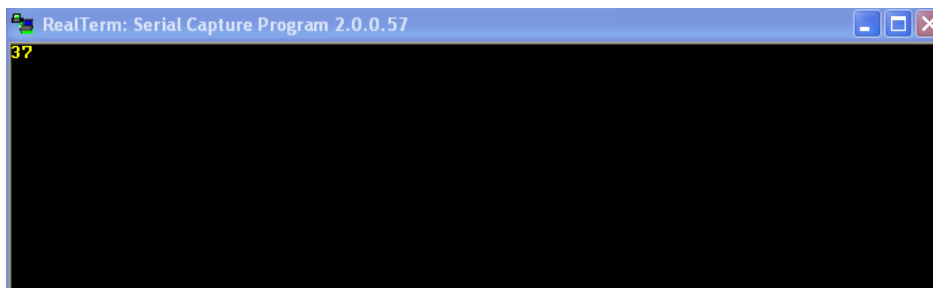


Figura 75: Temperatura adquirida con termómetro diseñado.



Por su parte, al comparar las mediciones realizadas con el termómetro implementado y las realizadas con el de referencia, se obtuvieron los resultados que se resumen en la Figura 77. En dicha tabla, se encuentra que el mayor porcentaje de error encontrado corresponde a 1.09%, lo cual equivale a 0.4 °C de diferencia. Debido a que durante las mediciones realizadas fue el único caso donde se presentó este porcentaje de error, dicho porcentaje de error se puede atribuir a la mala colocación del termómetro por parte del usuario, situación que, a pesar que se le instruyó previamente al usuario en el uso correcto del dispositivo, dependerá del mismo para ejecutar su uso adecuadamente, siendo imposible el control de esta variable en el diseño del dispositivo.

De igual manera, como se muestra en la Figura 78, se encontró que el termómetro oral implementado, presenta una efectividad del 93.3%, tomando en cuenta una tolerancia de $\pm 0.2^{\circ}\text{C}$ por cada medición, siendo este resultado satisfactorio de acuerdo a las expectativas del diseño.

Así mismo, la diferencia en los tiempos de adquisición para una misma temperatura se relaciona con el hecho que entre menos estable se mantenga el termómetro en la boca del usuario habrá pequeños cambios en la temperatura que harán más lento el proceso de alcanzar un punto de equilibrio. Teniendo en cuenta lo anterior, se puede observar a partir de la Figura 77, que el termómetro implementado es capaz de obtener la temperatura del paciente, de manera oral, en un tiempo promedio de 83 segundos.

Figura76: Comparación entre temperaturas registradas por termómetro MT3001 y termómetro diseñado.

Individuo	Temperatura referencia [°C]	Temperatura diseño [°C]	Tiempo de adquisición [S]	% Error
1	36.4	36.3	90	0.27
2	37	37	70	0.00
3	36.5	36.4	80	0.27
4	37	36.8	70	0.54
5	36.4	36.4	100	0.27
6	37	37	90	0.00
7	36.8	36.4	70	1.09
8	36.8	36.9	90	0.27
9	36.6	36.4	70	0.54
10	37	37	100	0.00
11	36.7	36.6	70	0.27
12	37.0	36.9	120	0.27
13	36.4	36.4	70	0.00
14	36.4	36.5	80	0.27
15	36.8	36.6	70	0.54

Figura77: Efectividad de mediciones de temperatura

Equipo	Tolerancia [°C]	Efectividad
Termómetro digital	0.2	93.3%

E. Discusión

A partir de los objetivos planteados para este módulo, así como del seguimiento de la metodología planteada para su cumplimiento, fue posible implementar dispositivos que sean capaces realizar las mediciones de los signos vitales de concentración de glucosa en la sangre, presión arterial y temperatura corporal, de tal

manera que dicha información pueda ser enviada a la unidad Central de Información cuando esta lo requiera. Es de mencionar, que una de las principales limitaciones encontradas en el desarrollo de este trabajo de graduación, fueron las limitaciones tecnológicas con que se cuentan, resultando en la necesidad de adecuar instrumentos comerciales dedicados a la adquisición de estos signos vitales. Así mismo, aun cuando se hubiese contado con las herramientas tecnológicas necesarias, se considera que la tarea de diseñar un instrumento médico de este tipo, tomando en cuenta las implicaciones que un mal diseño podrían tener para la salud del paciente, es una actividad ardua que requiere una amplia experiencia en el campo de la instrumentación electrónica. No obstante lo anterior, se considera un importante logro abrir una nueva rama de investigación en Guatemala, de tal forma que en futuros proyectos se retome el concepto de telemedicina para ampliar sus alcances.

En cuanto a los resultados obtenidos, se encontró que el circuito diseñado para la adquisición de los datos almacenados en la memoria del esfigmomanómetro no afectó el funcionamiento normal del instrumento, encontrándose como máximo, con respecto al esfigmomanómetro de referencia, un porcentaje de error del 8.58% en la medición del pulso cardíaco, 2.8% en la medición de la presión sistólica, mientras que un 5.48% en la diastólica. Es de resaltar, que se decidió utilizar el esfigmomanómetro de muñeca LifeSource UB-515 ya que no se dispuso de uno del mismo modelo que el intervenido, no obstante a partir de los anteriores resultados descritos se demostró que se obtienen mediciones similares. Cabe mencionar, que dado que el principal objetivo de este circuito era obtener las mediciones realizadas por el instrumento comercial sin afectar su funcionamiento, solo se realizaron tres mediciones, de tal forma que se comprobara que el dispositivo sigue brindando mediciones exactas. Así mismo, ya que la presión arterial es un proceso dinámico, se decidió espaciar las mediciones en 5 minutos en cada paciente, de tal manera que la presión se restableciera tras la medición.

Por su parte, en la implementación del circuito para obtener la concentración de glucosa en la sangre, este presentó la dificultad de implementar un protocolo según el estándar IrDA, un protocolo de comunicación con el cual no se había trabajado previamente, lo cual implicó una exhaustiva consulta de referencias, tomando en cuenta que no se disponía con detalles específicos del protocolo implementado por el

glucómetro Accu-Chek Aviva. De esta forma, el concepto de un analizador de paquetes fue fundamental para la decodificación del protocolo, habiendo, claro está, realizado varias pruebas en los diferentes modos que el protocolo IrDA implementa. Es así como a partir de la secuencia de datos recabada con el analizador de paquetes se encontró que el protocolo IrDA es en el modo RAW. De igual manera, el papel que, tanto el microcontrolador PIC24FJ64GB002 como el TFDU 4101 juegan en la implementación de la capa física del protocolo IrDA simplifica en gran manera la implementación del protocolo de comunicación ya mencionado. Por otro lado, otra de las dificultades presentadas en la implementación de este circuito, fue la decodificación de las instrucciones a partir de la trama de valores hexadecimales obtenidos, proceso que implicó la búsqueda de patrones, siendo la tabla de códigos ASCII de gran ayuda para dicha tarea. Finalmente, la obtención del dato almacenado en memoria representó menor complejidad, ya que a partir de los bytes obtenidos solo fue necesario ubicar, de acuerdo a su representación ASCII, los datos que coincidieran con la última medición registrada en el dispositivo, así como la hora y fecha de la medición. Uno de los aspectos importantes a destacar, es el hecho que al ser el protocolo de comunicación IrDA RAW, no es posible determinar cuándo se ha interrumpido la comunicación. No obstante, dado que el circuito que se ha integrado al dispositivo se encuentra fijado a éste, se garantizará en todo momento que nada bloquee la línea de visión entre el transceptor IrDA del glucómetro y el del circuito diseñado.

Es de notar que en el desarrollo de este objetivo, no fue necesario realizar una medición de la efectividad del circuito, ya que únicamente se debía determinar la capacidad del mismo de obtener la información. En cuanto al tiempo de adquisición de la información, se encontró que este no es crítico, cumpliéndose en un tiempo mucho menor que los 30 segundos máximos propuestos en los objetivos, dependiendo prácticamente del tiempo que el glucómetro tarda en realizar la muestra luego que la muestra de sangre se ha colocado en la tira.

Por otro lado, en el diseño del termómetro para la adquisición de la temperatura corporal de manera oral, el termómetro digital DS18B20 resultó de gran utilidad en la implementación, principalmente porque este brinda la medición de manera digital, además que este se encuentra calibrado por el fabricante para ofrecer mediciones

precisas. De esta forma, al utilizar este dispositivo se evitó la necesidad de implementar un circuito quemidiera la temperatura de manera analógica, lo que hubiese implicado la caracterización del mismo, así como el diseño de circuitos para acomodar las señales y luego digitalizarlas. De esta forma, la tarea de la implementación del termómetro se resumió a la configuración del dispositivo de tal manera que se pudiera obtener la máxima resolución del dispositivo, es decir, 12 bits, y con los cuales se obtuvo una resolución de 0.1°C en la lectura de la temperatura. No obstante lo anterior, la configuración del DS18B20 con 12 bits de resolución, implica un aumento en el tiempo de conversión del dispositivo, siendo este de 750 milisegundos por lectura, lo cual, por el algoritmo implementado en él termómetro diseñado, no representa un problema como se describirá a continuación.

De esta forma, el algoritmo implementado consiste en la acomodación, vía software, del formato en la cual el DS18B20 devuelve la lectura en una cadena de caracteres que representa la temperatura medida. Así mismo, en este algoritmo se procede a muestrear la temperatura cada 10 segundos, razón por la cual, el retraso en tiempo de conversión para la configuración de 12 bits, mencionado previamente, no afecta en el resultado final. Durante cada muestreo la temperatura es comparada con la anterior medida, de tal manera que cuando estas coinciden, se ha llegado a un equilibrio térmico entre la temperatura del paciente y la del termómetro. Cabe mencionar, que la exactitud depende enteramente de este tiempo, ya que si se aumenta el tiempo en el muestreo, se podrá mejorar la exactitud alcanzada, sin embargo, el termómetro será más lento en la adquisición de la temperatura del paciente. Así mismo, este tiempo de adquisición de la temperatura depende de la transferencia de calor del material que recubre el termómetro DS18B20. Aun cuando se buscó un material con buena conducción térmica, el tiempo en que el cambio de temperatura afecta al DS18B20 retrasa la medición.

En cuanto a los resultados obtenidos a partir de las pruebas realizadas con el termómetro diseñado, se encontró que este cumple con los $\pm 0.2^{\circ}\text{C}$ de exactitud propuestos para el dispositivo. De las 15 muestras realizadas, se encontró que éste presenta un 93.3% de efectividad. De esta forma, se encontró que el porcentaje de error máximo encontrado con respecto al termómetro de referencia corresponde a 1.09%, lo

cual equivale a 0.4°C de diferencia. Es de mencionar que la efectividad del termómetro diseñado depende en gran manera del correcto uso del termómetro, ya que si el usuario no mantiene estable el termómetro, o la punta, donde se encuentra incorporado el DS18B20, no tiene contacto con el tejido debajo de la lengua, las mediciones serán erróneas, situación que no es posible controlar, pudiéndose únicamente instruir previamente al usuario en su utilización. Finalmente, en la determinación del tiempo de adquisición de la temperatura del paciente, se encontró que en promedio el dispositivo tarda 83 segundos, como se observa en la Figura 77.

XX. APÉNDICE D. Sistema de Medición de Frecuencia Respiratoria y Peso

A. Metodología general

Para obtener las señales de procesamiento necesarias se llevó a cabo lo siguiente:

1. Se investigó en el mercado local de Guatemala y en referencias en línea las opciones factibles para la medición de frecuencia respiratoria y peso.
2. Se determinó una opción factible para la medición de peso en el mercado local.
3. En el caso de la balanza digital se procedió a remover los circuitos integrados que ésta posee para dejar disponible el trabajo con los sensores de carga que ésta posee.
4. Se diseño y construyó un circuito que funcione como acondicionador de la señal a partir de la respuesta que mostraba cada sensor.
5. Se conectó el circuito diseñado a los sensores de carga y al microcontrolador en el cual se realizó el procesamiento de la señal.
6. Se desarrolló un algoritmo en lenguaje de alto nivel para el microcontrolador que cumple con la implementación de un filtro sin realimentación y el envío del resultado final por medio de protocolo RS-232 o serial.
7. Se realizó pruebas de medición de peso muerto en los sensores, ajustando las ganancias de todos los amplificadores instrumentales. Se realizó la misma prueba con el peso de algunos usuarios. Estos datos de medición fueron tabulados y comparados con su valor respectivo a partir de una balanza mecánica de referencia.
8. Una vez tabulados estos datos se procedió a determinar la función de caracterización adecuada que mostraba la respuesta del circuito.
9. En lo referente a la medición de frecuencia respiratoria, se comenzó por el diseño y armado de un cinturón con galgas extensiométricas que funcionan como transductores de la expansión de la caja torácica del usuario en señales de voltaje.
10. Luego, se hizo un diagrama esquemático del circuito a implementar así como las mejoras encontradas para habilitar el ajuste automático del equipo.

11. Al igual que en la balanza, también se desarrolló un algoritmo que utiliza un filtro digital no realimentado utilizado para la adquisición de datos del usuario.

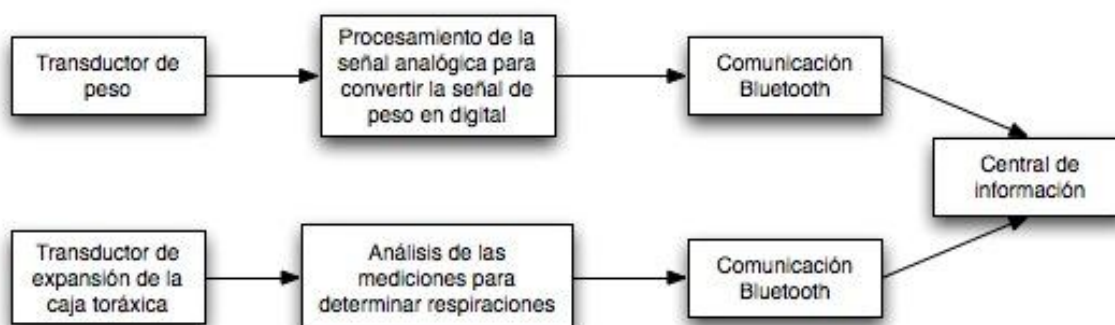
12. Se realizaron las mejoras y cambios pertinentes en el algoritmo para presentar adecuadamente los valores.

13. Se procedió a hacer mediciones con el cinturón utilizando el algoritmo desarrollado. Se tabularon estas mediciones con un valor de referencia del método convencional por observación del paciente.

14. Con las mediciones se determinó la adecuada caracterización del equipo.

15. Se integraron los dos sistemas completos a la unidad de procesamiento central por medio de interrupciones para no alterar el proceso de medición. El diagrama general del módulo se puede observar en la Figura 79.

Figura 78: División del Sistema de Medición de Frecuencia Respiratoria y Peso.

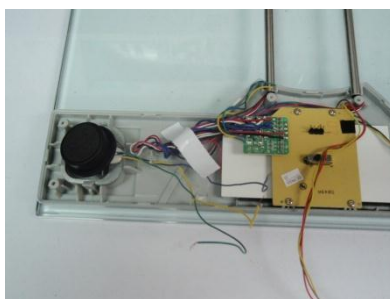


B. Medición de peso

1. Diseño experimental

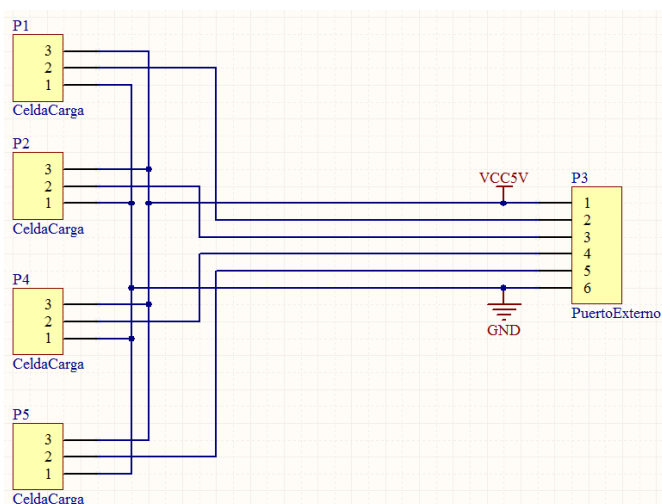
a. Se buscó identificar cada uno de los componentes internos en la balanza Tecnipesa EB9008H para comprender su funcionamiento.

Figura 79: Tarjeta interna y sensor de carga de la balanza



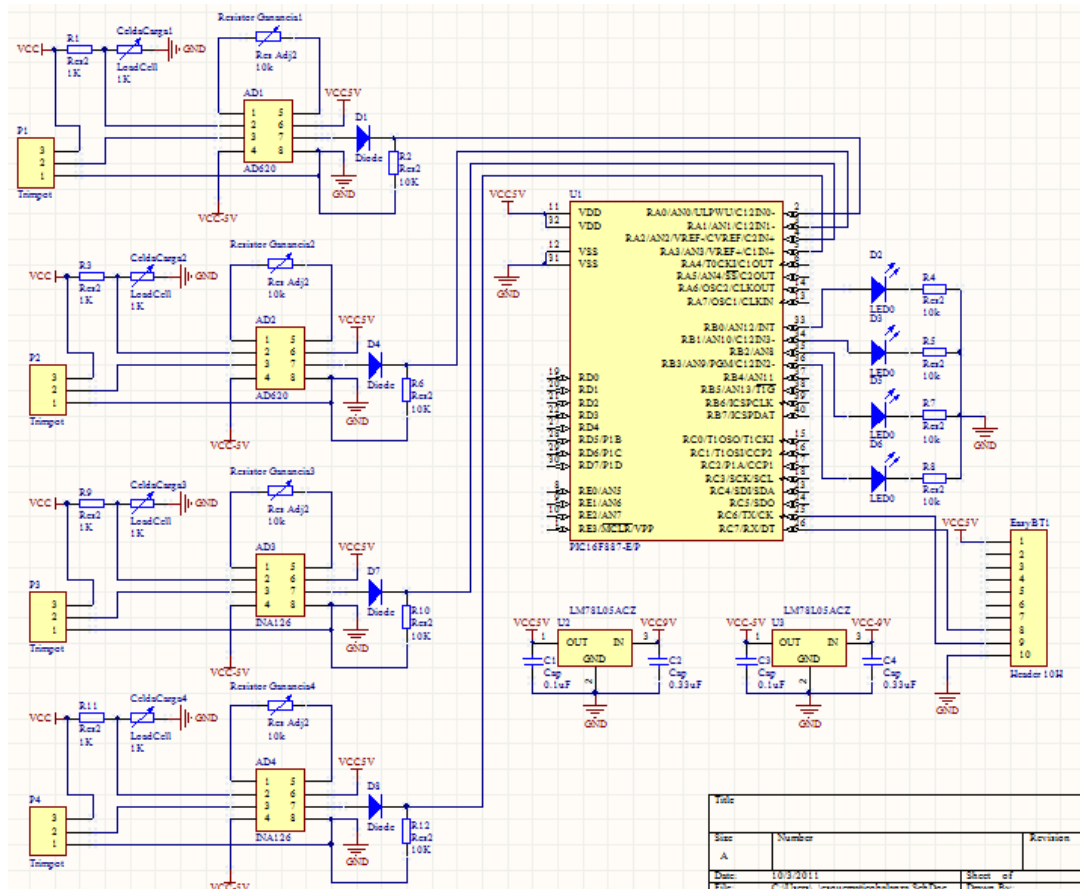
- b. Se quitó la tarjeta con el circuito impreso que trae la balanza así como la pantalla de despliegue de información, dejando los sensores dentro del equipo.
- c. Se realizaron mediciones de resistencia en las terminales de las 4 celdas de carga de la balanza para determinar su conexión interna, colocando diferentes pesos sobre las mismas.
- d. Se conectaron las 4 celdas de carga a una placa ubicada dentro de la balanza siguiendo el siguiente esquema:

Figura 80: Conexión de celdas de carga dentro de la balanza.



- e. Se armó e implementó un circuito amplificador diferencial para cada uno de los sensores utilizando los circuitos integrados INA126 y AD620, interconectando de esta manera la salida de las celdas de carga con las entradas del ADC del microcontrolador.

Figura 81: Circuito de implementación de los sensores de la balanza.

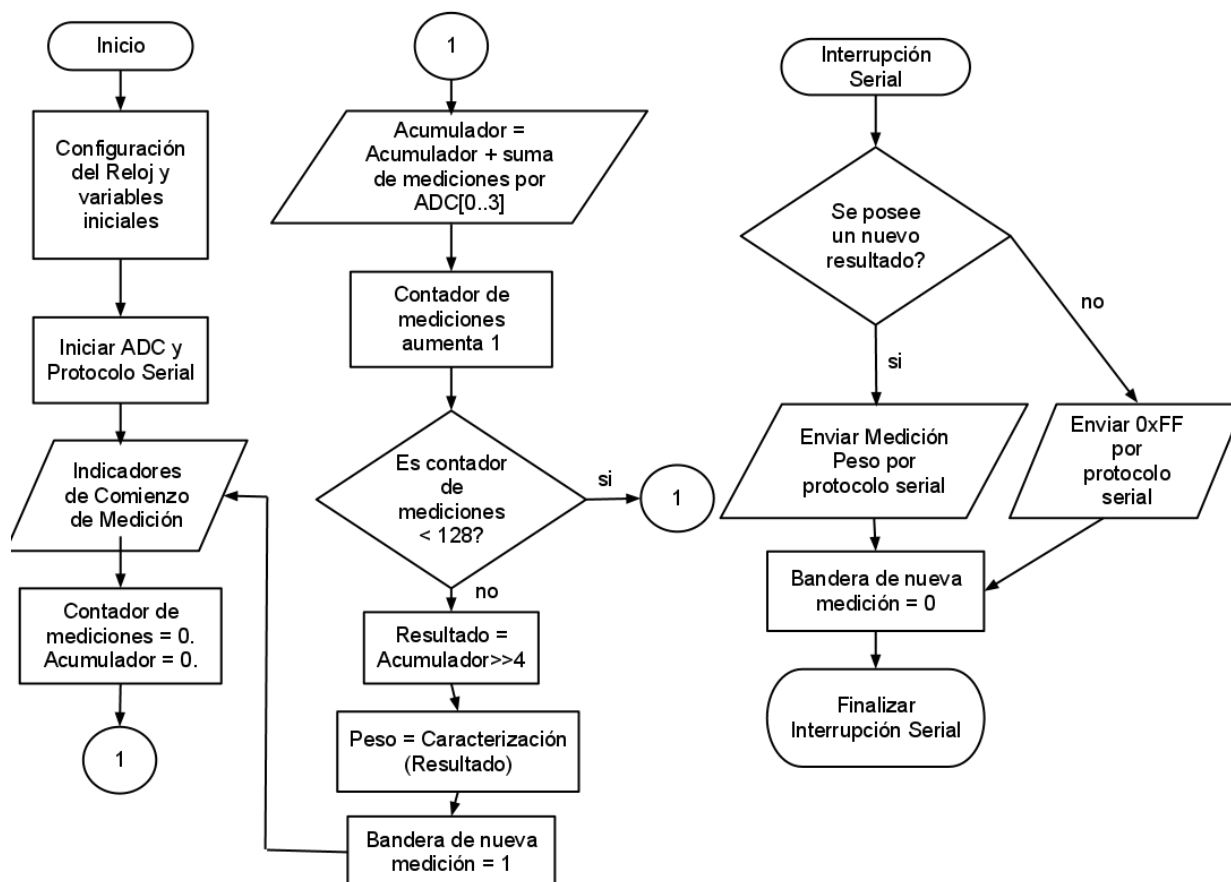


f. Se ajustó el voltaje de referencia de cada uno de los sensores a manera de obtener una diferencia de 0V en la salida del amplificador instrumental. Además se ajustó la ganancia de cada sensor para tener resultados similares en la amplitud de cada medición.

g. Se utilizó una balanza mecánica Tecnipesa(Tecnipesa, 2011) con una incertidumbre de ± 2 libras como valor de referencia para la caracterización de la balanza.

h. Se programó en el microcontrolador 16F887, un algoritmo que considerara las diferentes mediciones de peso del usuario para traducirlo en valores de variables enteras en el microcontrolador.

Figura 82 Algoritmo para medición de peso



El algoritmo descrito en la Figura 83, consiste en primero configurar el reloj interno, ADC y protocolo serial. Luego comienza a realizar las mediciones de cada sensor individualmente y sumarlos en un acumulador. Después se divide esta suma de las mediciones entre 16 para reducir componentes de ruido innecesario. Por último se aplica una fórmula lineal a este dato para obtener el peso en libras a partir de la medición que se tiene de todos los ADC del microcontrolador. Para el envío de la información, se habilitó el envío por medio de la interrupción serial, por la cual se envía el peso cuando existe una medición nueva y 0xFF cuando no se cuenta con una medición reciente.

- i. Se tabularon los datos con diferentes pesos muertos, o masas fijas, incluyendo algunas personas para realizar la caracterización de la balanza.
- j. Se determinó la fórmula de caracterización lineal que se debe aplicar a la medición obtenida para desplegar el resultado.
- k. Se tabularon los datos y se concluyó respecto a la efectividad de la balanza.

1. Se integró el sistema implementado a la unidad central de monitoreo de la red de sensorial de signos vitales Konócete.

2. Resultados

Figura83 Resultado obtenido con base en el peso de referencia en libras

Peso de referencia [libras]	Resultado obtenido en variable entera	Peso de referencia [libras]	Resultado obtenido en variable entera	Peso de referencia [libras]	Resultado obtenido en variable entera
135	1037	50	383	156	1292
135	1031	50	382	156	1285
135	1050	50	382	156	1288
135	1062	78	643	156	1289
135	1061	78	644	156	1285
50	381	78	642	156	1242
50	382	78	642	156	1281
50	381	78	642	156	1289
50	383	78	642	156	1288
50	383	78	641	156	1280
50	382	78	642	156	1256
50	381	78	644	156	1267
50	382	78	643		

En la Figura 84, se pueden observar los valores que obtuvo el acumulador a partir de la suma de todos los ADC utilizados en el microcontrolador. Estos fueron los valores utilizados para encontrar el modelo lineal para caracterizar la balanza.

Figura84: Resultados de la caracterización de la balanza

Peso de Referencia [libras]	Valor obtenido por el sistema [libras]	Desviación de la referencia [libras]	Peso de Referencia [libras]	Valor obtenido por el sistema [libras]	Desviación de la referencia [libras]
180	179.46	0.538	166	168.15	2.15
180	179.69	0.314	166	169.38	3.382
156	156.22	0.222	166	170.89	4.894
156	157.34	1.342	166	170.33	4.334
156	157.06	1.062	185	186.07	1.07
156	157.51	1.51	185	186.63	1.63
130	131.13	1.134	185	185.57	0.566
115	114.84	0.162	180	181.20	1.198
115	114.17	0.834	180	181.93	1.926
115	113.72	1.282	180	182.54	2.542
115	114.05	0.946			

En la Figura 85, se pueden observar los resultados de las mediciones aplicando la fórmula de caracterización lineal implementada. Encontrando que la mayor desviación encontrada fue de 4.894 libras del peso de referencia.

Figura85: Errores cuadrados de la caracterización de la balanza.

Valor de referencia mínimo [libras]	Pendiente lineal de la caracterización	Intercepto [libras]	Mayor desviación del peso de referencia [libras]	R ² del modelo lineal
0	0.0571	17.746	4.894	0.9984
25	0.0569	18.266	4.894	0.9995
50	0.056	20.143	4.894	0.9997
115	0.0554	21.708	4.894	0.9987

La Figura 86, muestra los elementos de la fórmula de caracterización de la balanza considerando diferentes rangos de medición. Siento el menor valor de 0, 25, 50 115 libras hasta las 180 libras como máximo. La pendiente y el intercepto se refieren al los valores de una ecuación lineal de la forma:

$$Peso_{[enlibras]} = pendiente * acumulador + intercepto$$

De esta manera se puede observar en la tabla 4 que el intercepto cambia dependiendo el rango de mediciones que se tomaron en cuenta. El aspecto más importante de esta tabla es la última columna, donde se muestra el R^2 del modelo lineal, en el cual el mayor es 0.9997. Tomando en cuenta que todos los valores de R^2 son mayores a 0.998 se concluye que es aceptable considerar las mediciones obtenidas por el sistema y transformarlas por medio de un modelo lineal para obtener el peso en libras. Además considerando que el mayor R^2 se obtuvo considerando las mediciones entre 50 y 180 libras, se optó por utilizar este rango como válido para realizar las mediciones en usuarios.

3. Discusión. Cabe destacar que en este trabajo sólo se están compartiendo los resultados finales de la medición efectiva del peso, sin embargo se puede mencionar algunos de los problemas obtenidos en el inicio de las mediciones en esta sección, así como la solución que se le dio a los mismos.

En un principio se buscó invadir la memoria de la balanza Tecnipesa Modelo EB9008H, debido a que esta funcionaba con protocolo I2C. Los problemas que se encontraron con este procedimiento fueron específicamente la dificultad para determinar dónde se guardaba la última medición realizada, ya que esta balanza tenía la capacidad de informar al usuario sobre el cambio relativo de peso entre la medición actual y la anterior. Para realizar este proceso, la balanza recorría su memoria interna con los valores y aquel que esté en un rango menor a 10 libras, se consideraba como el valor anterior a la medición actual de ese usuario. Además, considerando la falta de información que se tenía sobre el funcionamiento interno de la balanza y el hecho de que no se pudo reconocer el

algoritmo de guardado de la memoria se recurrió a implementar directamente los transductores de la balanza.

Los transductores que este equipo posee son celdas de Carga en cuyas características está que funcionan linealmente en rangos de peso entre 40Kg y 50Kg^v. Estos rangos equivalen entre 88 y 110libras. La balanza al poseer 4 de estos sensores se cree poder soportar pesos por encima de las 400 libras y medirlos apropiadamente, sin embargo, el sensor está caracterizado para funcionar apropiadamente hasta las 330 libras. Es por esta razón que se considero que el sistema completo debía determinar dos aspectos importantes en su medición, primero la incertidumbre de su medición y segundo, el rango apropiado de operación.

Al momento de implementar las celdas de carga se optó por utilizar el puente de Wheatstone ya que es ampliamente utilizado para la instrumentación electrónica. El primer inconveniente encontrado, fue el hecho de que los sensores colocados en un puente de Wheatstone poseen únicamente dos terminales para cada uno de los elementos que se utilizan, mientras las celdas de carga encontradas en la balanza tienen 3 terminales. Mediante la observación de los cambios en los valores de resistencia de los sensores en cada par de terminales se concluyó que estos sensores se pueden implementar en una configuración de un cuarto de puente de Wheatstone. Esto se debe a que en dos pares de terminales, se encontraba un valor constante de medición y en la última combinación mostraba una variación de aproximadamente 1Ω respecto a su valor nominal de $1K\Omega$.

El cambio en el valor de resistencia del sensor era muy pequeño como para detectar un cambio de voltaje significativo con el ADC utilizado del microcontrolador 16F887. Fue por ello que se procedió a configurar un circuito acondicionador de la señal. Los resultados obtenidos de estas mediciones fueron aceptables, sin embargo en búsqueda de mejorar la precisión del circuito se procedió a utilizar un amplificador instrumental.

^vInformación encontrada en el sitio de www.sparkfun.com

Considerando que cada sensor posee dos resistores internos, de los cuales uno es variable se conectó cada uno en configuración de un cuarto de puente, considerando la disposición interna de la resistencia variable y la fija del sensor. De manera que el sensor conformaba las dos resistencias de un lado del puente de Wheatstone. La otra mitad se armó con una resistencia variable como se puede observar en la figura 20. Se determinó por medio de experimentación que la manera más apropiada para implementar los sensores es utilizando cada sensor por separado con su respectivo valor de referencia individual y dedicado, debido a que las señales no se comportan con la misma amplitud al interconectar todos los sensores. Por ello al separarlos individualmente, se pudo acondicionar cada sensor para obtener los valores deseados. De esa manera se eliminó el ruido que tenía una amplitud de 1V respecto al cambio de la señal de 2V, cuando se conectaban todos los sensores juntos.

En cada medición realizada se trata de que cada medición de los sensores presente valores de ganancia similares para asegurar la apropiada suma de los resultados. Cabe destacar que se redujo la cantidad de bits de precisión del ADC interno del microcontrolador debido a variaciones de ruido encontradas mientras se realizaba la medición de voltajes constantes. Esto sin afectar el resultado final por la manera en que se procesan los datos al finalizar cada medición.

Realizando las pruebas en personas y pesos muertos, se analizó cada una de las correlaciones que existían entre las mismas. Al finalizar el proceso se determinó que el error cuadrado de la linealización del sensor presenta mejoras notables cuando el mínimo peso considerado a pesar es de 50 libras. Tomando en cuenta que el sistema pesa a personas mayores a 18 años, se excluye la medición de niños y son casos poco comunes según las mediciones realizadas en el laboratorio, de personas con esa edad que pesen menos de 50 libras. La precisión del sensor indica que puede detectar pesos de 25 libras, sin embargo se sugiere utilizarlo para pesar personas sobre 50 libras para no afectar desfavorablemente su precisión.

Actualmente la balanza implementada posee una efectividad del 85% referente a un porcentaje de error menor al 2% del peso de la persona. El porcentaje de efectividad fue determinado con la cantidad de mediciones que están dentro de la incertidumbre permitida de 3 libras. Estos resultados se pueden observar en la tabla 3 y 4. Las pruebas realizadas muestran que el mayor error es de 4.88 libras, lo cual resulta en una alteración del Índice de Masa Corporal de la persona. Es por ello que para futuras etapas de investigación se recomienda utilizar otro método para obtener los valores de referencia con menor incertidumbre y corroborar la linealización de la persona en rangos similares de peso. También cabe destacar, que al igual que las balanzas comerciales la forma de pararse sobre el equipo que opte el usuario afecta considerablemente el resultado a medir, por lo que al utilizar este equipo se debe recordar que el paciente debe pararse de manera erguida sobre el equipo, distribuyendo el peso de manera uniforme entre los 4 sensores de carga y evitando moverse bruscamente.

El sistema de medición de peso obtiene un resultado cada 8 segundos, el cual guarda en una variable temporal. De esta manera, cuando la unidad central de monitoreo de la red de sensores solicite dicha información, el programa entra en la interrupción serial y envía la respuesta en el protocolo y tiempo requerido.

C. Medición de frecuencia respiratoria

1. Diseño experimental

a. Se buscó identificar los métodos convencionales de medición de frecuencia respiratoria.

b. Se investigó sobre los transductores conocidos como galgas extensiométricas para conocer su funcionamiento y la viabilidad de su implementación para medir la expansión de la caja torácica cuando un usuario respira.

c. Se diseñó un cinturón que mantuviera las galgas extensiométricas en un lugar apropiado para la medición de respiraciones en una persona.

Figura 86: Modelo de cinturón en inhalación y exhalación

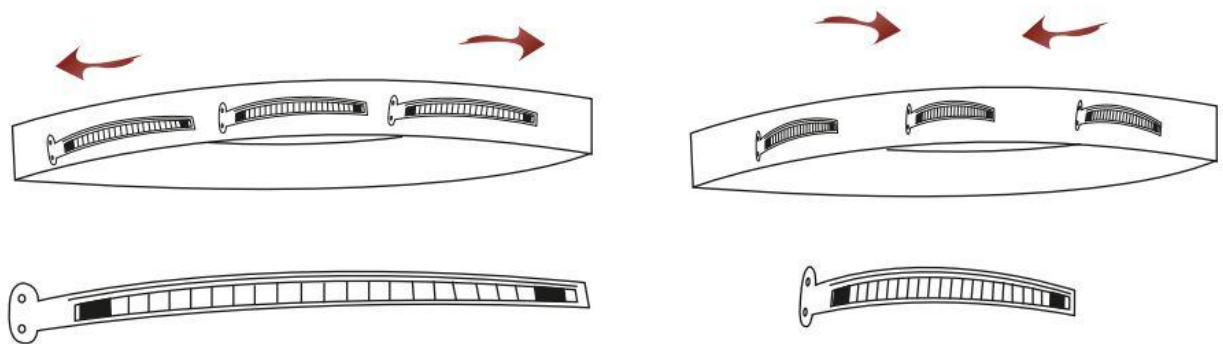


Figura 87: Cinturón para Medición de Frecuencia Respiratoria.



d. Se realizaron mediciones para verificar la correspondencia entre la variación del valor de resistencia y voltaje en la salida, respecto a las inspiraciones y espiraciones del usuario medido.

e. Se construyó una segunda versión del cinturón, la cual buscaba mejorar la capacidad de mantener a los sensores en su lugar cuando se cambiaba entre usuario.

Figura 88: Segunda versión del cinturón para Medición de Frecuencia Respiratoria.

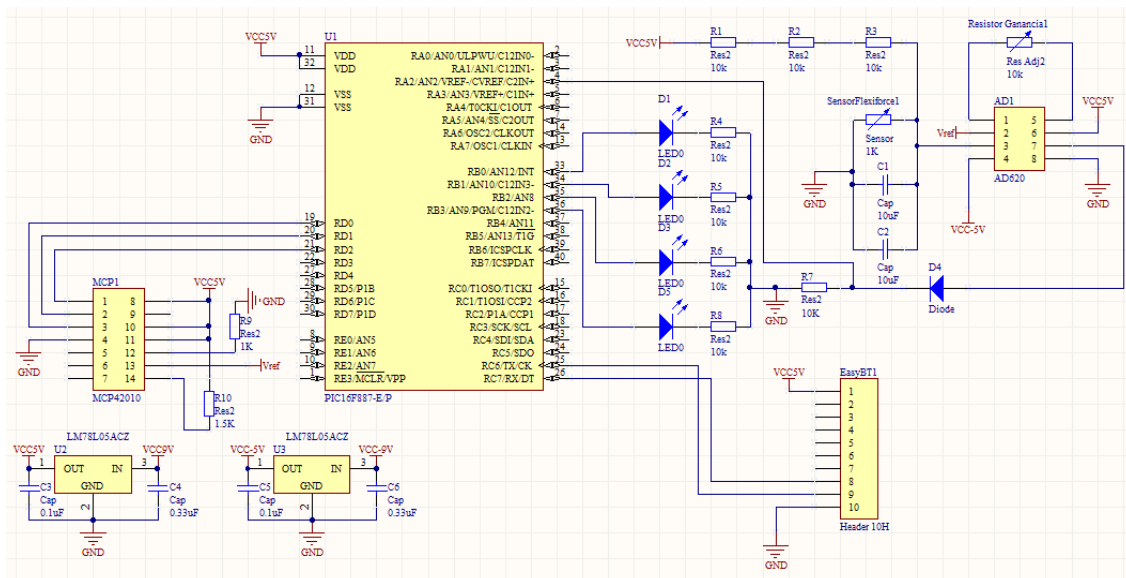


Figura 89: Usuario utilizando la segunda versión del cinturón.



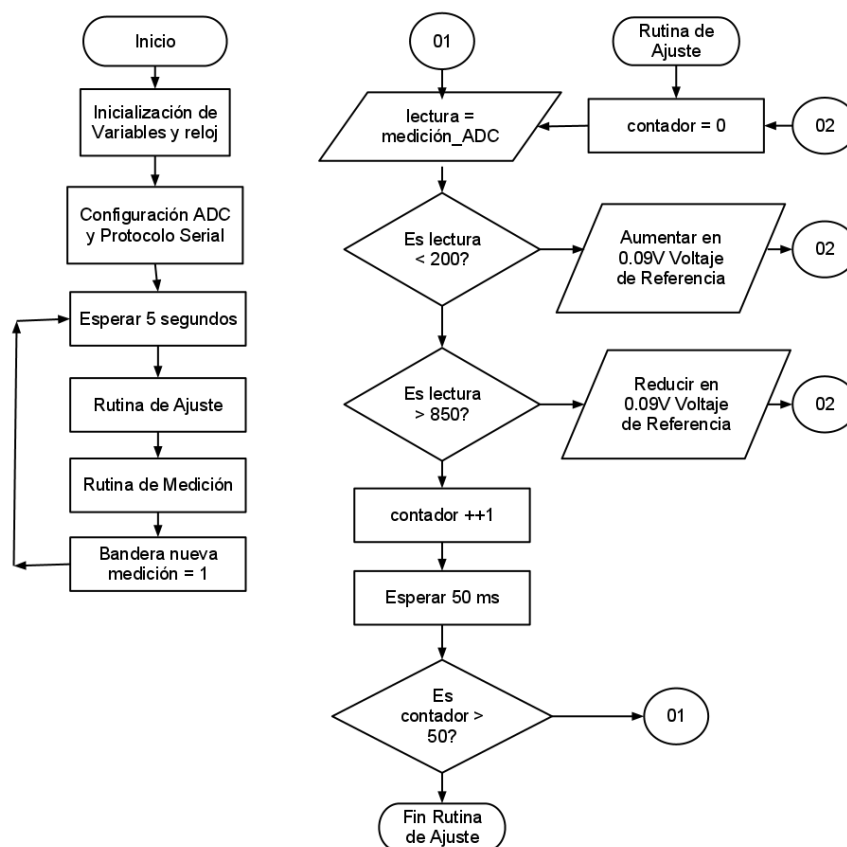
f. Se procedió a implementar un circuito capaz de ajustarse por medio de programación para la detección de la frecuencia respiratoria. Donde el sensor consiste en 3 galgas extensiométricas conocidas como Flexiforce (Tekscan, 2011). La capacidad de ajustarse se refiere a alterar el voltaje de referencia con el cual se compara la entrada del sensor de frecuencia respiratoria. El circuito se muestra en la Figura 91.

Figura 90: Circuito implementado para la Medición de Frecuencia Respiratoria.



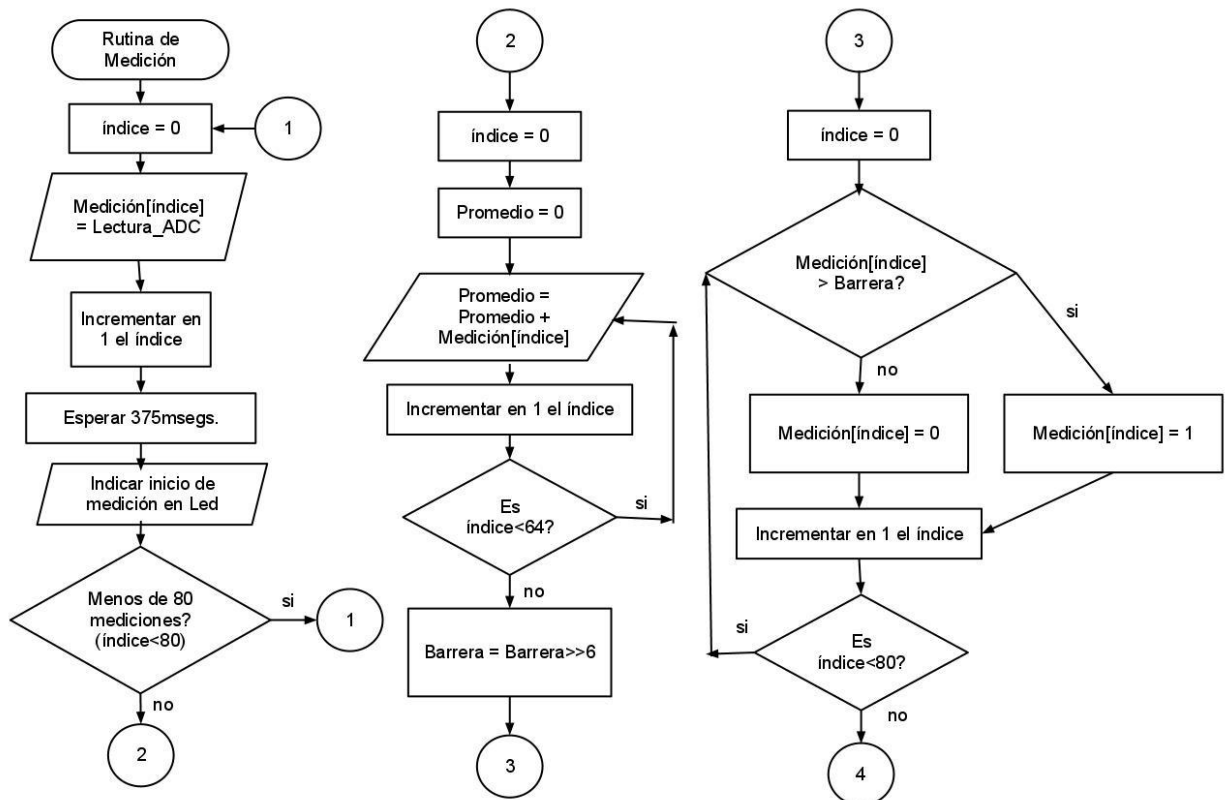
g. Se programó un algoritmo en el microcontrolador 16F887 que realiza el ajuste del voltaje de referencia y luego analiza los datos para enviar los resultados por protocolo Bluetooth.

Figura 91: Algoritmo para medir frecuencia respiratoria sección A



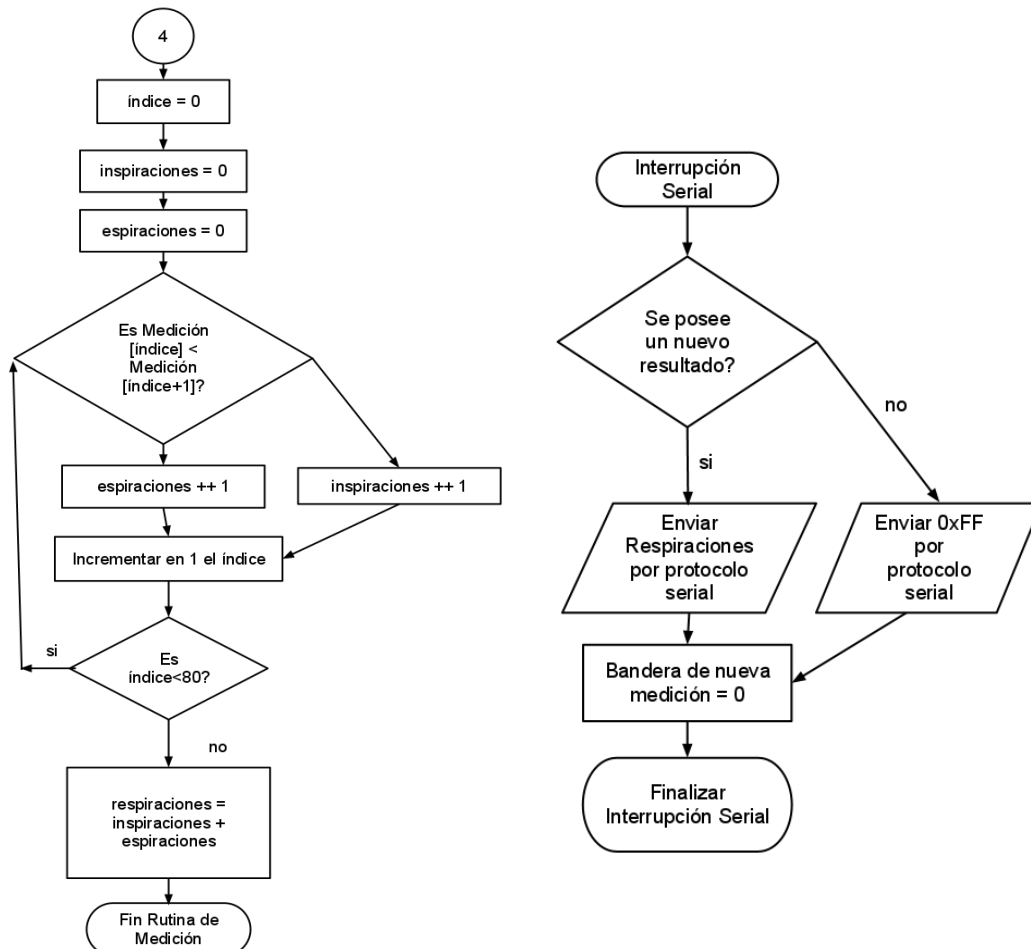
En este algoritmo se muestra cómo se realiza la configuración inicial y donde el ciclo principal del programa incluye una rutina de ajuste antes de una de medición. El objetivo de realizar este ajuste es para acondicionar la señal de cada usuario al sistema de forma automática.

Figura 92 Algoritmo para Medición de Frecuencia Respiratoria sección B



En este punto se describe el cálculo del promedio de las mediciones, que luego funcionará como punto para definir el umbral de cuando una medición está en inspiración o espiración. Dejando el registro de las mediciones como 0 o 1 cuando es espiración o inspiración respectivamente.

Figura 93 Algoritmo para Medición de Frecuencia Respiratoria sección C



h. Se tabularon los datos de respiraciones obtenidas con el equipo y se compararon dichos resultados con las mediciones por el método convencional.

i. Se tabularon los datos y se concluyó respecto a la efectividad del sistema.

j. Se integró el sistema implementado a la unidad central de monitoreo de la red de signos vitales Konócete.

2. Resultados. En la Figura 95 se muestran los resultados de la medición por parte del ADC de la respiración de un usuario. Los máximos se refieren a espiraciones y los mínimos a inspiraciones.

Figura 94: Mediciones de la primera versión del cinturón tomadas cada 400ms sin aplicar filtros

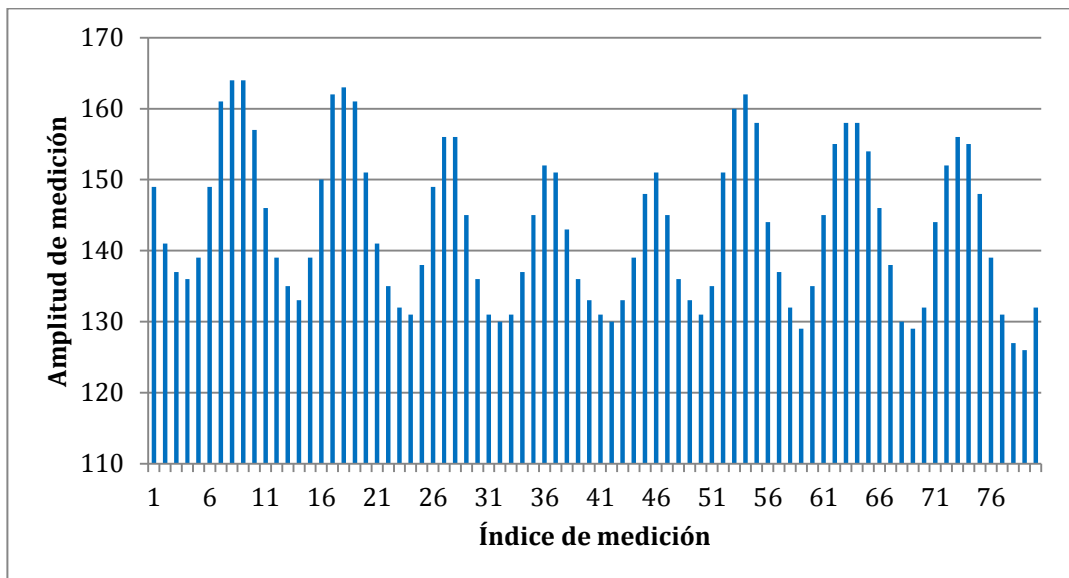
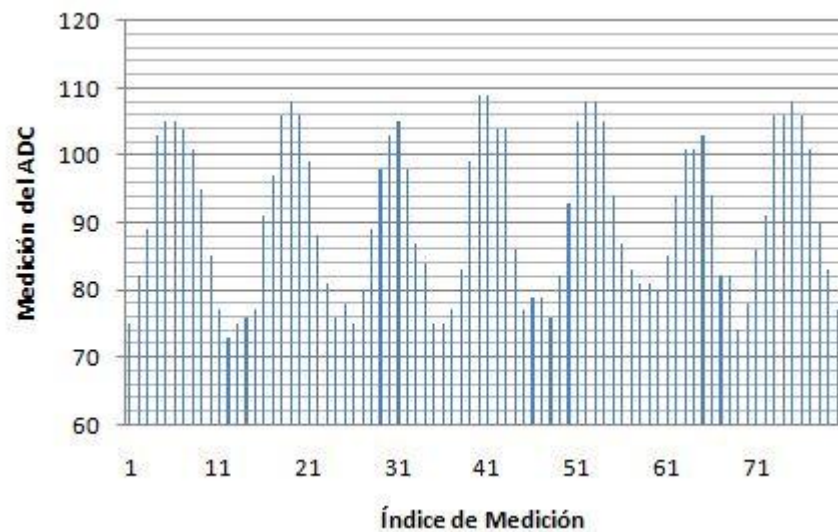


Figura95 Mediciones con alteraciones en la primera versión del cinturón



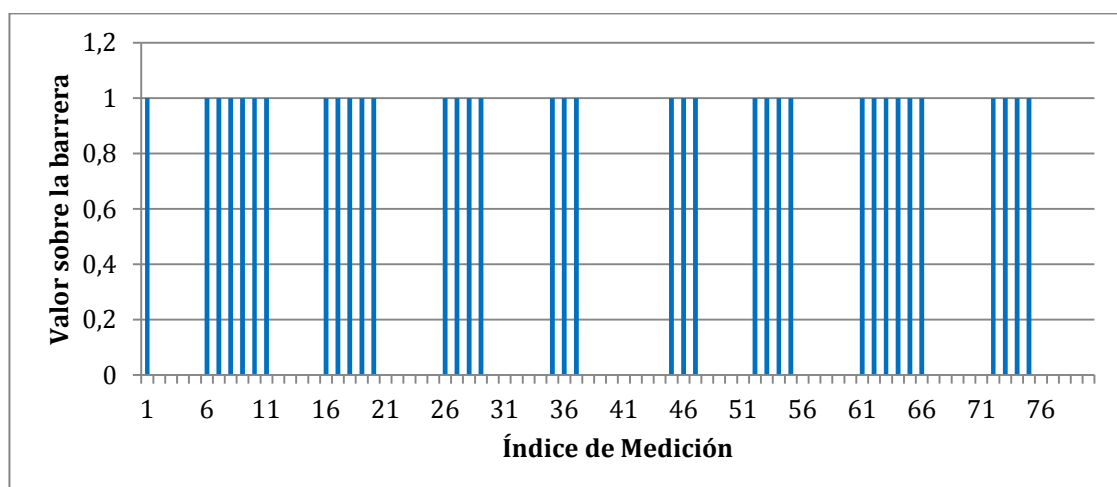
En la primera versión del cinturón se pueden observar alteraciones en las mediciones que son resultado de la forma en que respira el usuario. Es por estas alteraciones que un algoritmo para buscar crestas no funcionó ya que se perdían algunos datos importantes y no se contaba con el índice exacto donde se encontraba la cresta como para determinar cuánto es el tiempo que existe entre cada medición.

Figura96: Promedio y desviación para estimar la barrera de medición

Promedio entre el rango de mediciones de 1 a 64	Promedio entre el rango de mediciones de 1 a 80	Desviación
144.062	143.237	0.825

La Figura 97 ilustra que el promedio de las primeras 64 mediciones con respecto a las 80 del registro, tiene una desviación de 0.825. Por lo que el promedio se puede obtener de las primeras 64 mediciones y evitar hacer una división entre 80 para obtener el promedio, cambiándolo por realizar un corrimiento de 6 bits correspondiente a una división entre 64.

Figura 97 Detección de cambios de inspiración y espiración en un usuario



En la Figura 98, se puede observar cómo queda el registro de mediciones después de comparar cada medición con el promedio calculado. A partir de este número de cambios es que se cuentan la cantidad de espiraciones e inspiraciones que tuvo el usuario durante el tiempo de medición.

Figura98 Resultados de Medición de Frecuencia Respiratoria en distintos usuarios

Usuario	Frecuencia respiratoria detectada por el sensor [rpm]^{vi}	Medición convencional durante 30 segundos [rpm]	Medición convencional durante 60 segundos [rpm]	Desviación sobre la medición de 30 segundos [rpm]	Desviación sobre la medición de 60 segundos [rpm]
1	18	9	19	0	1
2	16	8	17	0	1
3	20	11	22	2	2
4	14	12	20	10	6
5	9	8	15	7	6
6	9	9	17	9	8
7	15	9	15	3	0
8	15	8	14	1	1
9	26	7	14	12	12
10	20	10	21	0	1
11	24	13	26	2	2
12	23	12	27	1	4
13	21	14	25	7	4
14	19	10	21	1	2
15	16	12	20	8	4
16	24	12	24	0	0
17	12	8	16	4	4
18	12	6	11	0	1
19	9	5	10	1	1
20	15	11	22	7	7
21	16	10	21	4	5

^{vi} En este documento la dimensional de *rpm* se refiere a respiraciones por minuto, no a revoluciones por minuto como generalmente se utiliza en ingeniería.

A partir de los datos presentados en la Figura 99, se concluye que a pesar de las grandes desviaciones presentadas, se cuenta con mayor cercanía cuando se mide al paciente durante 30 segundos y no durante 1 minuto.

Figura99: Porcentajes de mediciones respecto a la desviación de 1 o 2 rpm

Método de comparación	Mediciones con menos de 2 rpm de desviación	Mediciones con menos de 3 rpm de desviación	Porcentaje de mediciones con menos de 2 rpm de desviación	Porcentaje de mediciones con menos de 3 rpm de desviación
Medición convencional por 30 segundos	13	17	54.2%	70.8%
Medición convencional por 60 segundos	13	19	54.2%	79.2%

En la Figura 100 podemos observar el porcentaje de mediciones del sensor que se encuentran dentro de una desviación de 3 rpm, acercándose al 80% comparada con la medición convencional de 60 segundos. Por lo cual se procedió a realizar las mediciones de autoajuste para diferentes usuarios utilizando las mediciones realizadas durante 30 segundos con el sensor.

Figura100: Mediciones realizadas con el cinturón completo

Prueba	Género	Teórico [rpm]	Medido [rpm]	Desviación del Valor Teórico [rpm]
1	m	14	14	0
2	f	14	14	0
3	f	20	19	1
4	m	14	14	0
5	m	10	10	0
6	m	20	18	2
7	f	8	7	1
8	m	14	15	1
9	m	14	13	1
10	m	18	20	2
11	m	16	15	1
12	m	10	9	1
13	f	16	19	3
14	f	14	14	0
15	m	14	13	1
16	m	16	16	0
17	f	4	4	0
18	f	24	24	0

En la tabla 101, como mediciones finales del sensor podemos observar que el 100% de las mediciones cuentan con una desviación de 3 respiraciones por minuto o menor. Lo cual es un valor aceptable considerando que 4 es la incertidumbre del método convencional de medición, al contar la cantidad de ciclos de respiración por minuto que realiza el paciente durante 15 segundos.

3. Discusión. Aquí se presentan los resultados de los procedimientos relevantes para el proceso de adquisición de datos del equipo. Entre los procedimientos que no cuentan con sus resultados listados podemos hacer mención de que en una etapa preliminar para el primer diseño del cinturón que mide frecuencia respiratoria, se pensó utilizar solamente un sensor Flexiforce (galga extensiométrica). Uno de los inconvenientes encontrados de este modelo fue el hecho de la poca repetitividad de la respuesta del sistema. Además, el área que un sensor de estos puede cubrir, mientras trabaja adecuadamente, es de aproximadamente 5 pulgadas. Considerando un tórax que mide en promedio 45 pulgadas, el área que puede cubrir un sensor no es significativa.

Con base en estas observaciones se decidió por utilizar más galgas extensiométricas conectadas en serie para ser más sensible a los cambios en la expansión de la caja torácica, encontrando que evidentemente se encontró con un instrumento de medición más sensible al utilizar 3 galgas extensiométricas conectadas en serie que solamente una.

La primera versión del cinturón presentó inconvenientes en su repetitividad de medición debido a que con cada colocación del equipo, se movían los sensores de su lugar. Sin embargo en esta versión ya se contaban con valores aceptables en sus mediciones. En la segunda versión del cinturón como se puede observar en los figuras 89 y 90, se redujo la variación del lugar de los sensores al ajustarlos con hilo al cinturón. El sistema de sujeción del equipo para esta edición fue utilizando broches a diferencia del velcro utilizado en la primera medición. Esto se realizó de esta manera por la facilidad de colocación y el hecho de que el velcro bajo mucho tiempo de uso pierde estabilidad y fuerza en la sujeción.

Para el análisis de la señal adquirida por el ADC del microcontrolador, se optó en primera instancia por filtrar la señal utilizando un filtro no recursivo, sumando los primeros 8 términos de la señal. Sin embargo aunque el ruido se redujo al utilizar este algoritmo, no se pudo determinar apropiadamente la concavidad de la señal para determinar los mínimos y máximos medidos sin alterar la capacidad de memoria y rango de tiempo de medición.

Luego se optó por utilizar un algoritmo, luego de haber amplificado la señal de entrada, que consiste en estimar un promedio de todas las mediciones realizadas por un tiempo de 30 segundos. Los puntos donde la medición está por encima de esta barrera y cuando está por debajo se corresponden a la cantidad de espiraciones e inspiraciones respectivamente. De manera tal que se cuenta la cantidad de cambios sobre el promedio de las mediciones para determinar cuántas respiraciones se hicieron por esos 30 segundos, lo que se obtiene fácilmente para un minuto completo considerando periodicidad en el ritmo de respiración.

Se consideró la idea de realizar una transformada rápida de Fourier para encontrar la frecuencia de las mediciones esperando encontrar componentes de alta amplitud en las correspondientes a la frecuencia respiratoria del instrumento. Sin embargo, el microcontrolador utilizado no posee la capacidad para realizar una integral o una raíz cuadrada, por lo cual se optó por otras alternativas. Además la memoria disponible del microcontrolador no tenía la capacidad para seguir acumulando más datos por la forma en que se compilaba la información en el lenguaje de programación utilizado.

Tomando en cuenta que el sensor está midiendo constantemente mientras está ubicado en el paciente, se optó por implementar una interrupción activada cuando recibe una solicitud de la unidad central de monitoreo. De esta manera cada vez que exista un nuevo dato registrado se enviará esa información a la unidad central sin interferir por un tiempo significativo la medición que se esté realizando.

La capacidad de autoajuste del sistema se realizó en las etapas finales del desarrollo del equipo, al encontrar que no existe una forma estándar de colocación del instrumento sin afectar la precisión al momento de adquirir los datos. En otras palabras, se tenía que ajustar el valor de referencia y ganancia para que funcionara adecuadamente con cada usuario diferente. A cuenta de esto, el autoajuste consiste en un algoritmo que se realiza antes de hacer la medición en el cual se modifica el valor del voltaje de referencia con el cual se compara la señal a manera de encontrar un rango estable donde el sensor funciona adecuadamente para cada

usuario. Este proceso varía dependiendo de la forma en que la persona respira, sin embargo se diseñó de tal manera que no se tarde más de 10 segundos antes de informar que hubo error en la etapa de ajuste. Este proceso de autoajuste se realiza al momento de colocar el cinturón en el usuario y encender el equipo. El algoritmo modificará el voltaje de referencia que utiliza el amplificador instrumental para mantener la señal amplificada en valores mayores a 1.3 voltios y menores a 4.5 voltios. Si el proceso de autoajuste no logra realizarse efectivamente, o sea el usuario muestra respiraciones que no están dentro del intervalo deseado, se muestra una señal de error para que se repita el proceso de colocar el sensor, apagando el sistema y encenderlo luego de cambiar la posición en que se asegura el aparato.

En la Figura 101 se pueden observar los resultados de 18 usuarios diferentes utilizando el método de autoajuste. Como se muestra en la última columna de esta tabla, se realizaron pruebas en 11 hombres y 7 mujeres. Cabe destacar que en 9 usuarios se tuvo que ajustar el equipo dos veces más, ya que el sistema notificaba de errores en las mediciones. Como se puede observar en los resultados, el cinturón en su segunda versión, muestra una desviación máxima de 3 respiraciones por minuto respecto al valor obtenido con el método convencional. Esto se traduce en un error del 18.75%. Sin embargo, el método convencional posee una incertidumbre de 4 respiraciones por minuto, así que se podría considerar que el cinturón cumplió con mantenerse dentro del margen aceptable en todas sus mediciones. El objetivo planteado del cinturón es tener una incertidumbre de 2 respiraciones por minuto, cumplida en el 94.4% de los casos.

XXI. GLOSARIO

SMS: SMS por sus siglas en ingles Short MessageService (SMS). Este es un servicio que permite enviar y recibir mensajes de texto entre teléfonos celulares atreves de las redes de telefonía celular. (Haseman, 2010)

Tarjeta de memoria Micro SD : Tarjeta de memoria común mente utilizada en celulares y otros dispositivos móviles para incrementar capacidad de almacenamiento. Se puede utilizar para almacenar texto, imágenes música y video. (Crowder, 2010)

Smartphones: Dispositivo que permite realizar llamadas telefónicas pero agrega funcionalidades que antes solo se podían encontrar en una computadora. Por ejemplo la funcionalidad de enviar y recibir correos electrónicos. (Mitchell, 2010)

rpm: La dimensional de rpm se refiere a respiraciones por minuto.

mmHg: La dimensional mmHg se refiere a milímetros de mercurio la cual es una dimensional de presión.

Transistor 2N3904:Transistor NPN de propósito general.

mg/d:Miligramos por decilitro. Dimensión de peso utilizada para medir concentración de glucosa en la sangre. (Society for Biomedical Diabetes Research, 2010)

SMS1	mmHg19
Tarjeta de memoria SD1	Transistor 2N390424
Smartphones11	mg/d24
rpm18	