



ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO
FACULTAD DE INFORMÁTICA Y ELECTRÓNICA
ESCUELA DE INGENIERÍA EN ELECTRÓNICA,
TELECOMUNICACIONES Y REDES

**“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DE COMUNICACIÓN
OMRON – CELULAR PARA MEDIR LA PRESIÓN ARTERIAL”**

TESIS DE GRADO

PREVIA A LA OBTENCIÓN DEL TÍTULO DE:

INGENIERO EN ELECTRÓNICA TELECOMUNICACIONES Y REDES

PRESENTADO POR:

ERNESTO LEONIDAS URGILES MOROCHO

DIEGO JAVIER COLCHA GUANANGA

RIOBAMBA – ECUADOR

2015

DEDICATORIA

El presente trabajo quiero dedicar en primer lugar a Dios por concederme el regalo más grande que es la vida, a mis padres que gracias a su esfuerzo y sacrificio diario me han permitido culminar una etapa más de mi vida estudiantil ya que siempre he recibido su apoyo incondicional sin limitaciones, además por inculcarme valores que sin duda alguna me han servido dentro de la sociedad en que me desenvuelvo en el diario vivir y a mis hermanas para que les pueda servir de ejemplo y luchan siempre por hacer realidad sus sueños logrando alcanzar sus metas.

A una persona que fue especial en mi vida y que hoy ya no se encuentra con nosotros pero de seguro siempre guiará e iluminará mis pasos desde el cielo.

Ernesto Leonidas Urgilés Morocho

DEDICATORIA

A Dios por darme la vida y brindarles salud a mis padres, permitiéndome llegar a esta etapa tan importante de mi formación profesional.

A mis grandes amigos mis padres que han sido el pilar fundamental para poder hacer realidad uno de mis sueños.

A mis hermanos que siempre están conmigo en las buenas y en las malas y familia en general que con sus consejos han hecho de mí el que soy.

A mis amigos más cercanos con los que compartí la etapa estudiantil, que entre risas y enojos supimos salir adelante en cada uno de los proyectos.

Diego Javier Colcha Guananga

AGRADECIMIENTO

A Dios por brindarles salud y vida a mis padres Ernesto y María Dolores quienes me dieron la oportunidad de tener unos estudios y lograr obtener un título como profesional.

A cada uno de los profesores de la ESPOCH pertenecientes a la Escuela de Ingeniería Electrónica, Telecomunicaciones y Redes con quienes tuve la oportunidad de poder adquirir conocimientos que me han permitido realizar este trabajo investigativo. A nuestro director de Tesis Ing. Wilson Baldeón quien nos brindó su apoyo necesario con sugerencias y comentarios que nos sirvieron para culminar nuestro proyecto.

A mi novia Karina por estar presente en cada momento compartido ya que con sus motivaciones me permitieron seguir siempre en lucha por conseguir lograr cumplir esta meta. Y a todos mis compañeros con los que pude compartir experiencias únicas e inolvidables en las aulas.

Ernesto Leonidas Urgilés Morocho

AGRADECIMIENTO

En primer lugar a Dios por guiarme, darme fuerzas para seguir adelante y sin desmayar sobrellevar las adversidades y salir victorioso de cada una de ellas.

A mis queridos padres por su apoyo, comprensión, consejos y ayuda en aquellos momentos difíciles, y que supieron darme los recursos necesarios para poder estudiar. Inculcándome valores, han formado mi carácter y mi perseverancia para alcanzar mis objetivos.

A cada uno de mis profesores de la facultad que compartieron sus conocimientos y hacen posible este trabajo investigativo, y a nuestro director de tesis el Ing. Wilson Baldón que con su ayuda y sugerencias ha hecho posible culminar este proyecto.

Diego Javier Colcha Guananga

FIRMAS DE RESPONSABLES Y NOTA

NOMBRE	FIRMA	FECHA
ING. NICOLAY SAMANIEGO		
DECANO DE LA FACULTAD DE INFORMÁTICA Y ELECTRÓNICA
ING. FRANKLIN MORENO		
DIRECTOR DE LA ESCUELA DE INGENIERÍA ELECTRÓNICA, TELECOMUNICACIONES Y REDES
ING. WILSON BALDEÓN LOPEZ		
DIRECTOR DE TESIS
ING. JORGE YUQUILEMA		
MIEMBRO DEL TRIBUNAL
DIRECTOR DPTO DOCUMENTACIÓN
NOTA DE LA TESIS	

DERECHOS DE AUTORÍA

Nosotros Ernesto Leonidas Urgiles Morocho y Diego Javier Colcha Guananga declaramos ser los autores del presente trabajo de Tesis “**DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DE COMUNICACIÓN OMRON – CELULAR PARA MEDIR LA PRESIÓN ARTERIAL**” previo a la obtención del título de INGENIEROS EN ELECTRÓNICA, TELECOMUNICACIONES Y REDES, haciéndonos responsables por las ideas, criterios, doctrinas y resultados expuestos en esta Tesis y declarando que el patrimonio de la misma pertenece a la Escuela Superior Politécnica de Chimborazo.

FIRMAS:

Ernesto Leonidas Urgiles Morocho

CI. 060467158-6

Diego Javier Colcha Guananga

CI. 060380978-1

ÍNDICE DE ABREVIATURAS

OMS	Organización Mundial de la Salud.
GSM	Global System for Mobile Communication
ETSI	European Telecommunications Standards Institute.
SMS	Servicio de mensajes cortos.
TDMA	Acceso múltiple por división de tiempo.
UIT	Unión Internacional de Telecomunicaciones.
GPRS	General Packet Radio Service.
3GPP	3rd Generation Partnership Project.
GMSK	Gaussian Minimum Shift Keying.
CLIP	Calling Line Identification Presentation.
CLIR	Calling Line Identification Restriction.
TRX	Transceiver (Transrecibidor).
EIR	Equipment Identity Register (Registro de Identificación del Equipo).
MS	Mobile Station (Estación Móvil).
AC	Authentication Center (Central de Autenticación)
SIM	Subscriber Identity Module (Módulo de Identificación de Suscriptor)
HLR	Home Location Register (Registro de Localización de Llamada)
BTS	Base Transceiver Station (Estación Transrecibidora de Base)
BSC	Base Station Controller (Estación Base de Control).
MSC	Mobile services Switching Center.
VLR	Visitor Location Register (Registro de Localización del Visitante)
ISDN	Integrated Services Digital Network.
PSTN	Public Switched Telephone Network.
SMSC	Short Message System Center.
PIN	Personal Identification Number

IMEI	International Mobile Subscriber Identity
PDU	Protocol Description Unit.
PWM	Pulse-width modulation
FAC	Federación Argentina de Cardiología
TTL	Transistor-transistor logic.
UART	Universally asynchronous receiver/transmitter
USB	Universal serial port.

INDICE GENERAL

PORTADA

AGRADECIMIENTO

DEDICATORIA

FIRMAS DE RESPONSABILIDAD

RESPONSABILIDAD DE AUTORES

INDICE DE ABREVIATURAS

INDICE GENERAL

INDICE DE FIGURAS

INDICE DE TABLAS

INTRODUCCIÓN

CAPÍTULO I

MARCO REFERENCIAL.....	- 19 -
ANTECEDENTES	- 19 -
JUSTIFICACIÓN	- 20 -
OBJETIVOS	- 23 -
1.1.1 OBJETIVO GENERAL:	- 23 -
1.1.2 OBJETIVOS ESPECÍFICOS.....	- 23 -
HIPÓTESIS	- 23 -

CAPÍTULO II

TELEMEDICINA.....	- 24 -
2.1.1 Introducción.....	- 24 -
2.1.2 Historia de la Telemedicina	- 25 -
2.1.3 Definición de la Telemedicina	- 27 -
2.1.4 Clasificación de la Telemedicina	- 29 -
2.1.4.1 Por tipo de Servicio	- 29 -
2.1.4.2 Por Especialidad	- 30 -
2.1.5 La telemedicina en el Ecuador.....	- 31 -
2.1.6 Beneficios de la Telemedicina.....	- 33 -
Presión Arterial.....	- 33 -
2.1.7 Presión arterial sistólica	- 34 -
2.1.8 Presión arterial diastólica	- 34 -

2.1.9	Presión sanguínea diastólica y sistólica saludable	- 35 -
2.1.10	Métodos de Medición de la Presión Arterial.....	- 35 -
2.1.11	Qué Hacer Cuando se Mide La Presión Arterial	- 36 -
2.1.12	MÉTODO DE CAMBIO DE COLOR CAPILAR	- 36 -
2.1.13	MÉTODO DEL PULSO	- 37 -
2.1.14	MÉTODO AUSCULTATORIO	- 37 -
2.1.15	MÉTODO OSCILOMÉTRICO	- 38 -
2.1.16	MÉTODO ULTRASÓNICO / DOPPLER	- 38 -
2.1.17	MÉTODO DIRECTO O INTRARTERIAL	- 38 -
	Hipertensión Arterial.....	- 39 -
2.1.18	Por qué es peligrosa la hipertensión arterial.....	- 40 -
2.1.19	La hipertensión un problema frecuente.....	- 41 -
2.1.20	RIESGOS Y COMPLICACIONES DE LA HIPERTENSIÓN ARTERIAL.....	- 41 -
2.1.21	CAUSAS.....	- 42 -
2.1.22	SÍNTOMAS.....	- 43 -
2.1.23	DIAGNÓSTICO	- 43 -
2.1.24	TRATAMIENTO.....	- 43 -
2.1.25	Dieta:	- 44 -
2.1.26	Ejercicio:	- 45 -
2.1.27	Medicación:	- 45 -
2.1.28	TENSIÓN ARTERIAL.....	- 46 -
2.1.29	RIESGO DE TENER PRESIÓN ARTERIAL ALTA	- 46 -

CAPÍTULO III

	Sistema de Comunicación GSM	- 48 -
3.1.1	Tecnología GSM	- 48 -
3.1.2	Origen y Evolución de GSM.....	- 49 -
3.1.3	Arquitectura GSM.....	- 51 -
3.1.4	Características del GSM	- 54 -
3.1.5	Características técnicas	- 55 -
3.1.6	Ventajas de GSM	- 56 -
	SERVICIO DE MENSAJES CORTOS DE TEXTO SMS	- 56 -
3.1.7	Formato del SMS	- 57 -
3.1.8	Arquitectura del SMS	- 58 -
	COMANDOS AT.....	- 59 -
3.1.9	Estructura de los comandos AT	- 60 -

3.1.10	Comandos AT para modem GSM o teléfono celular	- 60 -
3.1.11	Comandos AT para la configuración del modem GSM.....	- 61 -
Estudio de la tecnología Arduino		- 62 -
3.1.12	Definición de Arduino	- 62 -
3.1.13	Características	- 63 -
3.1.14	Ventajas de Arduino	- 63 -
3.1.15	Hardware Arduino	- 64 -
3.1.15.1	Arduino Uno.....	- 65 -
3.1.15.2	Arduino Nano.....	- 65 -
3.1.15.3	Arduino Lilypad.....	- 66 -
3.1.15.4	Arduino Mega 2560	- 66 -
3.1.15.5	Netduino	- 67 -
3.1.15.6	Arduino Fio	- 67 -
3.1.16	Entorno Arduino	- 68 -
3.1.16.1	Barra de Herramientas	- 68 -
3.1.16.2	Menús.....	- 68 -
3.1.17	Comenzando con Arduino.....	- 70 -
3.1.18	Consideraciones antes de adquirir una placa Arduino	- 74 -
3.1.19	Características de modelos estándar de Arduino	- 75 -
3.1.20	Análisis para la adquisición de la plataforma Arduino	- 78 -
3.1.21	Descripción del Arduino a Utilizar en el Proyecto	- 80 -
Accesorios de Arduino (Shields)		- 81 -
3.1.22	Tipos de Shield.....	- 82 -
3.1.22.1	Shield de Ethernet.....	- 82 -
3.1.22.2	XBee.....	- 82 -
3.1.22.3	Motor	- 83 -
3.1.23	Descripción del EFcom GPRS/GSM Shield a utilizar en el proyecto.....	- 83 -
Shield GSM y Arduino		- 87 -
3.1.24	Creación de una configuración de prueba para Shield GPRS.....	- 87 -
3.1.25	Opciones de la Interfaz	- 89 -
3.1.26	Conociendo el software Arduino	- 89 -
3.1.27	Comunicación de Arduino con Puerto Serie	- 92 -
3.1.27.1	Puerto Serie.....	- 92 -
3.1.28	Arduino y el Puerto Serie	- 94 -
3.1.29	Conexión de Arduino con un Ordenador.....	- 94 -
Tecnología Omron.....		- 96 -

3.1.30	Historia del Medidor de Presión Arterial Omron.....	- 96 -
3.1.31	Listado Cronológico de Monitores de Presión Arterial.....	- 97 -
3.1.32	Características	- 99 -
3.1.33	Especificaciones del medidor de presión arterial Omron.....	- 99 -
3.1.34	Especificaciones técnicas	- 100 -
	Microcode Studio.....	- 101 -
	Isis Profesional.....	- 101 -
3.1.35	Área de Trabajo.....	- 102 -
3.1.35.1	Barra de título.....	- 103 -
3.1.35.2	Menús.....	- 103 -
3.1.35.3	Herramientas.....	- 103 -
3.1.35.4	Área de trabajo.....	- 104 -

CAPÍTULO IV

	DISEÑO E IMPLEMENTACION DEL SISTEMA DE COMUNICACIÓN	- 105 -
	OMROM – CELULAR PARA MEDIR LA PRESIÓN ARTERIAL.	- 105 -
	Software y Hardware Arduino	- 105 -
4.1.1	Instalando Arduino	- 105 -
4.1.2	Primera Prueba con Arduino.....	- 107 -
	Funcionamiento Específico del Omrom HEM – 7113.....	- 112 -
4.1.3	Recepción de la Señal del Omrom	- 114 -
4.1.4	Código que se utilizó para la transformación.	- 117 -
4.1.5	Captura de datos del Omron con proteus	- 123 -
	Envío de SMS por comandos AT.....	- 128 -
	Envío SMS a través de Arduino.....	- 133 -
	Lectura de SMS por comandos AT	- 135 -
	Lectura de SMS a través de Arduino	- 137 -
	Desarrollo de Aplicaciones y Evaluación del Sistema	- 142 -
	Código de la aplicación general para el sistema de control de presión arterial	- 143 -

CAPÍTULO V

PRUEBAS Y RESULTADOS - 153 -

Evaluación del sistema de gestión local - 153 -

Comparación del sistema de comunicación omrom – celular con un medidor de presión manual..... - 156 -

Comprobación de Hipótesis - 156 -

CONCLUSIONES

RECOMENDACIONES

RESÚMEN

SUMARY

BIBLIOGRAFIA

ANEXOS

INDICE DE FIGURAS

Figura 1 Cinco Principales causas de muerte Ecuador 2000 - 2011	- 22 -
Figura 2 Revista Radio News (abril de 1924)	- 26 -
Figura 3: Presión Sistólica y Diastólica	- 35 -
Figura 4: Arquitectura GSM	- 51 -
Figura 5: Proceso de envío de mensajes cortos	- 57 -
Figura 6: Estructura básica de la red SMS	- 58 -
Figura 7: Placa Arduino Uno	- 65 -
Figura 8: Placa Arduino Nano	- 65 -
Figura 9: Placa Arduino Lyilpad	- 66 -
Figura 10: Placa Arduino Mega 2560	- 66 -
Figura 11: Placa Netduino	- 67 -
Figura 12: Placa Arduino Fio	- 67 -
Figura 13: Placa arduino Uno y Placa Arduino Leonardo	- 74 -
Figura 14: Placa Arduino Mega y Placa Arduino DUE	- 74 -
Figura 15: Descripción Arduino Mega 2560	- 80 -
Figura 16: Shield de Ethernet	- 82 -
Figura 17: Shield Xbee	- 82 -
Figura 18: Shield de Motor	- 83 -
Figura 19: EFcom GPRS/GSM Shield	- 84 -
Figura 20: Cargador de EFcom GPRS/GSM Shield	- 85 -
Figura 21: Top - view	- 86 -
Figura 22: Bottom view	- 86 -
Figura 23: Dimensiones mecánicas de EFcom GPRS/GSM Shield	- 87 -
Figura 24: Interfaz de Arduino	- 89 -
Figura 25: Ejecución de Arduino	- 90 -
Figura 26: Selección de Puerto COM en arduino	- 90 -
Figura 27: Selección de placa a trabajar en arduino	- 91 -
Figura 28: Consola serial en Arduino	- 91 -
Figura 29: Comunicación Serie y Paralelo	- 93 -
Figura 30: Monitor serial Arduino	- 95 -
Figura 31: Zonas del monitor del puerto serie de Arduino	- 95 -
Figura 32: Omron Medidor de presión arterial	- 100 -
Figura 33: Area de trabajo de ISIS	- 102 -
Figura 34: Barra de título de ISIS	- 103 -
Figura 35: Menú de ISIS	- 103 -
Figura 36: Herramientas de ISIS	- 104 -
Figura 37: Area de trabajo de ISIS	- 104 -
Figura 38: Descargando Arduino	- 106 -
Figura 39: Menú herramientas de Arduino	- 106 -
Figura 40: Menú Archivo de Arduino	- 108 -
Figura 41: Opción Preferencias del menú archivo de Arduino	- 109 -
Figura 42: Compilación en Arduino	- 109 -
Figura 43: Selección de la Ubicación de carga del sketch en Arduino	- 110 -
Figura 44: Esquema de montaje en ISIS con placa Arduino	- 110 -
Figura 45: Esquema de montaje para hacer parpadear un LED con Arduino	- 111 -
Figura 46: Código generado en Arduino	- 112 -
Figura 47: Circuito de la Placa Omron	- 113 -

Figura 48: Selección de Serial Communicator en la opción view de microcode	- 114 -
Figura 49: Valores de Configuración de Serial Communicator	- 115 -
Figura 50: Código 1 extraído del Omron	- 115 -
Figura 51: Código 2 extraído del Omron	- 116 -
Figura 52: Código 4 extraído del Omron	- 116 -
Figura 53: Resultado 1 de la transformación de los valores obtenidos con el Omron en programación Arduino	- 122 -
Figura 54: Resultado 2 de la transformación de los valores obtenidos con el Omron en Programación Arduino	- 123 -
Figura 55: Resultado 3 de la transformación de los valores obtenidos con el Omron en programación Arduino	- 123 -
Figura 56: Montaje del circuito en Proteus para capturar los datos del Omron	- 124 -
Figura 57: Valores de Configuración de puerto serial del arduino en Proteus	- 125 -
Figura 58: Valores de configuración de velocidad de transmisión de arduino en Proteus	- 125 -
Figura 59: Conexión del omron de cable serial a usb	- 126 -
Figura 60: Resultado de los datos obtenidos del Omron en Proteus	- 127 -
Figura 61: Conexión del Omron al Computador con cable serial a usb	- 128 -
Figura 62: Conexión de jumpers del Shield	- 129 -
Figura 63: Puerto COM a utilizar	- 129 -
Figura 64: Entorno hyperterminal	- 130 -
Figura 65: Selección del puerto COM en hyperterminal	- 131 -
Figura 66: Valores de configuración del puerto en hyperteminal	- 131 -
Figura 67: Prueba de recepción de datos con comandos AT	- 132 -
Figura 68: Recepción del SMS en el teléfono móvil	- 135 -
Figura 69: Ejecución de comandos AT	- 137 -
Figura 70: Montaje del circuito para encender 4 leds	- 138 -
Figura 71: Arduino Uno y Mega	- 144 -

ÍNDICE DE TABLAS

Tabla I: Características de los modelos mas estándar de arduinos	- 75 -
Tabla II: Características de Arduino Mega	- 81 -
Tabla III: Luz de estado de EFcom GPRS/GSM Shield.....	- 87 -
Tabla IV: Comparacion entre Arduino Uno y Mega	- 144 -
Tabla V: Resultados de mediciones realizadas a diferentes personas con el sistema implementado.....	- 155 -
Tabla VI: Comparación del sistema implementado con un medidor de presión manual.....	- 156 -

INTRODUCCIÓN

El presente trabajo de investigación tiene como propósito presentar a la sociedad una innovación tecnológica adaptada al mundo de la medicina, permitiendo contribuir en el mejoramiento de la vida de los seres humanos. En la actualidad la medición de la presión arterial es indispensable en pacientes que se acercan a consulta en cualquier centro médico ya que permite conocer el comportamiento que presenta el sistema cardiovascular de cada persona.

El proyecto describe el diseño e implementación de un medidor de presión arterial que permita al paciente obtener los valores de la presión diastólica, sistólica y pulso, logrando visualizar los datos en un teléfono celular a través de un mensaje de texto en un número previamente establecido.

Se puede detectar la hipertensión en sus inicios únicamente a través de revisiones periódicas, por lo que construir un sistema de comunicación para medir la presión arterial permite a las personas en especial adultos mayores que sufren este tipo de enfermedad tomar las medidas necesarias para poder realizar a tiempo un tratamiento. Se conoce como hipertensión arterial cuando la presión arterial disminuye demasiado provocando síntomas como: desmayos, mareos o en casos extremos shock en el paciente., incluso puede provocar complicaciones severas como un infarto de miocardio, una hemorragia o incluso trombosis cerebral.

El sistema consta de un medidor de presión arterial omron que es el encargado de obtener los datos de la presión diastólica, sistólica y pulso, de una placa arduino mega que cargado con un sketch permite la transformación de los datos que arroja el omron y de un shield GSM SIM900 mediante el cual se puede convertir un dispositivo en un terminal móvil con la capacidad de poder insertar una tarjeta SIM lo que permite establecer una comunicación con otros dispositivos móviles mediante las tecnologías GSM.

Con lo mencionado se pretende brindar un aporte significativo a los problemas de salud mediante la utilización de tecnología mejorando la calidad de vida de las personas.

CAPÍTULO I

MARCO REFERENCIAL

ANTECEDENTES

La telemedicina no es una disciplina reciente, su historia es anterior a la aparición de la comunicación electrónica remontando a la aparición del telégrafo, ya que las primeras consultas a distancia se realizaban a través de cartas, siendo el mensajero uno de los personajes más antiguos ya que la transmisión de noticias entre distintos grupos humanos es tan remota como su propia historia, con el pasar de los años empezó a efectuarse la prestación de servicios de telemedicina por radio en donde varios países ofrecían asesoramiento médico desde los hospitales a su flota de buques mercantes utilizando el código Morse. Hoy en día relacionando con la globalización de las comunicaciones y la presencia de internet resulta sencillo entender la prestación de servicios de medicina a distancia la cual muchos países la están aplicando y no necesariamente países industrializados.

En la actualidad es indispensable en países europeos la telemedicina tal es el caso del hospital Ramón y Cajal de Madrid que para el año 2013 ahorró un 40% de tiempo en el departamento de consultas dermatológicas, en el cual los médicos de atención primaria envían imágenes y primeros diagnósticos dermatológicos al departamento correspondiente del hospital. “Le evitamos al paciente viajes inútiles al hospital, pero además conseguimos detectar antes las dolencias graves”, explica Sergio Vañó, médico del departamento y creador de DermoMap, una aplicación para el móvil.

Los sistemas de telemedicina y de monitorización remota, han evolucionado considerablemente a lo largo de los últimos años, incrementando entre otros aspectos su autonomía, portabilidad y funcionalidad. Debido al desarrollo de nuevas tecnologías en las redes celulares y su adopción en escenarios relativos al cuidado de salud, las aplicaciones de telemedicina están continuamente experimentando avances sustanciales para la prestación de mejores servicios.

En países como el Ecuador, son relativamente escasos y caros los recursos necesarios para su aplicación, además tienen estos que competir con otras necesidades urgentes de los servicios de salud, por otro lado está bien el hecho de que quienes viven en lugares cuya geografía dificulta el acceso a una atención de salud de calidad y que pertenecen a sectores con menores ingresos los mismos que cuentan con una menor dotación de infraestructura de comunicaciones.

Implementar un sistema de comunicación en la medicina permitirá lograr un impacto social y tecnológico ya que el lugar en el que se encuentre el paciente no será un impedimento, lo importante será que exista la tecnología mínima para poder realizar el enlace a un tele consultorio. Una re-ingeniería de la atención sanitaria permitirá reducir las listas de espera y a la vez mejorar la calidad del servicio

JUSTIFICACIÓN

Este proyecto está enmarcado en el área de la electrónica y las telecomunicaciones aplicadas a la salud.

Siendo la calidad de vida un eje primordial del gobierno, este proyecto va a ayudar a mejorar la salud de las personas que sufren hipertensión arterial, así como también

puede servir como un mecanismo que alerte de esta enfermedad a personas que sean propensas a ella genéticamente o debido a su calidad de vida.

Por lo tanto la justificación de este proyecto está dada por la constitución, el objetivo 3 y el objetivo 11 del plan nacional de buen vivir periodo 2013-2017, así como también por las líneas de investigación declaradas por la FIE-ESPOCH.

Respecto de la constitución, el artículo 66, establece que “el derecho a una vida digna, que asegure la salud, alimentación y nutrición, agua potable, vivienda, saneamiento ambiental, educación, trabajo, empleo, descanso y ocio, cultura física, vestido, seguridad social y otros servicios sociales necesarios”. Por ello, mejorar la calidad de vida de la población es un proceso multidimensional y complejo.

Respecto de los objetivos del buen vivir se tiene:

Objetivo 3 mejorar la calidad de vida de la población” y menciona lo siguiente: entre los derechos para mejorar la calidad de vida se incluyen entre otros el acceso a la salud (art. 32). La calidad de vida se enmarca en el régimen del buen vivir, establecido en la constitución, dentro del sistema nacional de inclusión y equidad social (art. 340), para la garantía de servicios sociales de calidad en los ámbitos de salud, cultura física y tiempo libre, hábitat y vivienda, transporte y gestión de riesgos.

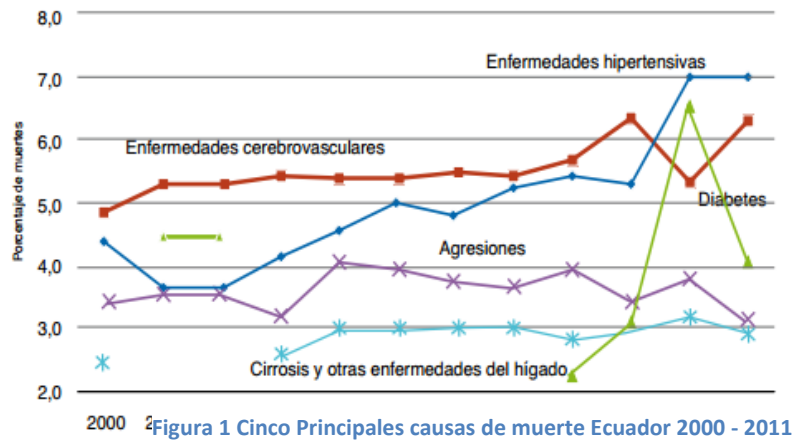
Objetivo 11: asegurar la soberanía y eficiencia de los sectores estratégicos

Para la transformación industrial y tecnológica.

11.3 democratizar la prestación de servicios públicos de telecomunicaciones y de tecnologías de información y comunicación (tic), incluyendo radiodifusión, televisión y espectro radioeléctrico, y profundizar su uso y acceso universal.

Este proyecto en función de lo indicado arriba ayudará a mejorar la calidad de vida de las personas hipertensas brindándoles un sistema de comunicaciones de tele monitoreo, de una forma eficiente sin la necesidad de que los pacientes propensos a tener una presión arterial alta se trasladen a sitios de consulta médica innecesariamente, además el medico dispondrá de estas mediciones en su celular inmediatamente de manera que pueda tomar las mejores decisiones para la buena salud del paciente. Este dispositivo se puede utilizar como un instrumento de ayuda para diagnosticar y tratar esta enfermedad.

Según las estadísticas del inen la hipertension es una de las 5 causas de muerte en el ecuador esto se indica en el gráfico. En el Ecuador, tres de cada 10 personas son hipertensas.



De acuerdo al último informe estadístico de la OMS en todo el mundo uno de cada tres adultos mayores de 25 años sufre de hipertensión arterial mientras tanto en el ecuador 3 de cada 10 personas padece de esta afección, cifras alarmantes de una patología que aunque no presenta síntomas, en algunos casos el individuo sufre dolores de cabeza, sudoración excesiva y taquicardia provocando cardiopatías y accidentes cardiovasculares, llevándolos a la muerte prematura y discapacidades.

Alrededor del mundo 9,4 millones de ciudadanos mueren anualmente como consecuencia de enfermedades del corazón donde la hipertensión arterial juega un rol preponderante. En el ecuador según reportes del INEC, en el 2010, de todas las causas de muerte registradas el 5% se deben a accidentes de tránsito, mientras que el 7% de los ecuatorianos mueren por hipertensión, con el fin de reducir el alto índice de enfermos y el descenso prematuro el MSP creo el plan estratégico Nacional de Prevención y Control de ECNT, con esta iniciativa las unidades operativas concientizan a la ciudadanía mediante casas abiertas, charlas, controles y eventos masivos sobre cómo evitar estos males.

OBJETIVOS

1.1.1 OBJETIVO GENERAL:

“DISEÑAR E IMPLEMENTAR UN SISTEMA DE COMUNICACIÓN PARA TRANSMITIR VIA MENSAJE A TRAVES DE LA RED CELULAR LOS VALORES DE LA PRESIÓN ARTERIAL MEDIDOS MEDIANTE UN DISPOSITIVO OMRON”

1.1.2 OBJETIVOS ESPECÍFICOS

- Analizar los protocolos de comunicación para la construcción del dispositivo de interfaz entre el omron y la red celular.
- Interpretar las señales del OMRON para acoplar al dispositivo GSM.
- Diseñar el sistema integrado de comunicación GSM para acoplarlo al medidor de presión.
- Verificar el envío de los datos en tiempo real.

HIPÓTESIS

El diseño e implementación de un sistema de comunicación omrom - celular brindara confiabilidad, seguridad y eficiencia en la recepción de datos.

CAPÍTULO II

TELEMEDICINA

2.1.1 Introducción

El constante desarrollo de las Tecnologías de la Información y Comunicaciones (TIC's) ha permitido incorporar nuevos dispositivos en el cuidado de la salud de los pacientes los mismos que permitan aprovechar la tecnología disponible usando tácticas que para lograr realizar el monitoreo de la salud de múltiples pacientes así también como dar respuesta inmediata a determinadas alertas. Estas estrategias integrales ofrecidas a través de las TIC's forman parte del mundo actual de la telemedicina.¹

¹ *Telemedicina en el Ecuador: un mundo de desafíos y oportunidades*. Bolaños, d. M. (2010 de Diciembre de 10). Consultado el 22 de Mayo de 2014, de <http://lagranja.ups.edu.ec/documents/1317427/1371462/05Telemedicina.pdf>

El intercambio de información médica entre los lugares remotos y los hospitales de especialidades, es la aplicación que más se ha desarrollado a nivel mundial, permitiendo a la telemedicina proporcionar servicios de salud a largas distancias. Estos avances tecnológicos han permitido que cada día sea más común encontrar en hospitales y centros de salud elementos como PCs, redes de computadoras, sistemas de comunicación, etc.

La telemedicina brinda la posibilidad de prevenir anomalías en la salud de las personas, diagnosticar de manera rápida enfermedades para poder orientar de una forma adecuada al paciente con su tratamiento y seguimiento de la enfermedad. Por lo tanto es importante crear modelos de atención médica que permitan conectar a los pacientes con los médicos y en general con los diferentes servicios de salud, logrando llevar la atención médica más allá de los centros de salud o de los hospitales, teniendo siempre disponible a un médico y así conseguir descongestionar siempre que sea posible las consultas físicas.

2.1.2 Historia de la Telemedicina

Se desconoce una fecha exacta en que apareció la telemedicina, pero desde la edad media se ha venido practicando cuando las familias opulentas enviaban muestras de orina a su médico y a su vez el médico realizaba un bosquejo gráfico para el diagnóstico lo que podría considerarse un ejemplo claro de la telepatología.

Luego con el desarrollo de los correos nacionales por el siglo XVIII, los enfermos escribían una carta a un médico eminente, en donde especificaban los signos y síntomas de su enfermedad, respondiendo éste el diagnóstico, recomendaciones e indicaciones a seguir y la prescripción que era realizada por el boticario local.

La telemedicina tiene dos fases diferenciadas: la analógica y la digital.

En la guerra civil norteamericana ya se daban informaciones médicas a través del telégrafo. Los médicos en 1900 fueron los primeros en utilizar los teléfonos convirtiéndose en un pilar fundamental de las comunicaciones médicas que se mantiene hasta la actualidad.

El objetivo que tenía el primer experimento con el teléfono era transmitir ruidos cardíacos y pulmonares a un especialista. En 1906 el inventor de la

electrocardiografía, Wilhelm Einthoven realizó una teleconsulta a través de la red telefónica.

A fines del siglo XIX, se hizo posible la comunicación por radio, a través del código Morse en sus inicios y posteriormente con la voz, tomando los beneficios de la comunicación por radio, se la utilizó para ayuda médica a marineros que se encontraban en buques en alta mar en donde no existía ningún médico a bordo.

La revista Radio News (Figura 2) en abril de 1924 publicó un editorial que literalmente traduce “Doctor Radio – Quizás”; la gráfica fue diseñada por el artista Howard V. Browne que hacía un diagnóstico remoto con televisión y audio, entrevistando de esta forma el doctor al paciente imprimiéndose automáticamente la prescripción.



Figura 2 Revista Radio News (abril de 1924)

Posteriormente en el año de 1930 en Alaska y Australia Ya se usaba en áreas remotas para transmitir información médica.

Por el año 1950 la telemedicina se empleó en los circuitos cerrados de televisión en los congresos de medicina. A finales, el Instituto Psiquiátrico de Nebraska, realizó un enlace de microondas de video desde el hospital de Omaha entre el departamento de Psiquiatría de la Universidad de Nebraska y el Hospital Estatal Mental en Norfolk.

Al final de los 60 y principios de la década de los 70 el gobierno de Estados Unidos de Norteamérica financió 7 proyectos de investigación-demostración en telemedicina con la esperanza que con estos proyectos se pudiera resolver problemas de atención médica a distancia.

A pesar del gran número de proyectos no se obtuvieron respuestas definitivas, se logró demostrar que las telecomunicaciones se pueden utilizar como un sustituto del desplazamiento de pacientes para recibir atención médica y para establecer un link vital en situaciones de emergencia donde el acceso de un médico se dificulte.

Es así que en realidad casi ningún programa de las décadas de los 60, 70 consiguieron mantenerse por sí solos. En 1984 se llevó a cabo un proyecto de telemedicina por satélite, el QNetwork demostrando la reducción de costos económicos y traslados de urgencias.

A finales de los 80 la NASA (National Aeronautics and Space Administration) con el programa Space Brigge la comunicación se hizo a través de video en una dirección de voz y fax bidireccionales entre cuatro hospitales de Estados Unidos, Armenia y el Centro Médico de Yerevan. El avance significativo que fue tomando las telecomunicaciones desde la década de 1960, ha ayudado a la revolución de la telemedicina, comenzando con la telefonía electromecánica hasta los tendidos de fibra óptica de alta velocidad de hoy en día.

La mayor revolución en la telemedicina se dio con la aparición de la computación, que posibilito el almacenamiento de grandes cantidades de datos médicos con la posibilidad de ser enviados a otros lugares para ser revisados y consultados. También se logró acceder a grandes bases de datos e Historias Clínicas y poder brindar educación médica a distancia.²

2.1.3 Definición de la Telemedicina

La palabra literalmente significa “medicina a distancia” ya que procede del griego “tele” que quiere decir distancia y medicina. Siendo así la telemedicina la práctica médica que brinda salud a pacientes distantes geográficamente pero conectados por sistemas de tecnología y comunicación.

El concepto de Telemedicina se remonta a los años sesenta, cuando se transmitían datos en las primeras misiones espaciales estadounidenses, además de la transmisión de videos e información acerca de la regulación del estado anatómico de los astronautas que tripulaban la nave en el espacio. La información que se enviaba desde la nave espacial y recibida en la base terrestre, permitía conocer el

² *Historia de la telemedicina*. Tigre, D. S. (2011). Consultado el 25 de Mayo de 2014, de <http://dspace.ups.edu.ec/bitstream/123456789/1052/13/UPS-CT002035.pdf>

estado de salud de los tripulantes, relacionada con su sistema cardíaco, nervioso y respiratorio, entre otros.³

Según la Organización Mundial de la Salud (OMS), se define la telemedicina: “El suministro de servicios de atención sanitaria, en los que la distancia constituye un factor crítico, por profesionales que apelan a las tecnologías de la información y de la comunicación con objeto de intercambiar datos para hacer diagnósticos, preconizar tratamientos y prevenir enfermedades y heridas, así como para la formación permanente de los profesionales de atención de salud y en actividades de investigación y evaluación, con el fin de mejorar la salud de las personas y de las comunidades en que viven”.

El INSALUD en el documento “Marco de la Telemedicina en el INSALUD” elaborado en 1998, considera que Telemedicina es: “La utilización de las tecnologías de la información y de las comunicaciones como un medio de proveer servicios médicos, independientemente de la localización tanto de los que ofrecen el servicio, los pacientes que lo reciben, y la información necesaria para la actividad asistencial”.

La Comisión Europea define a la Telemedicina como “la prestación de servicios de salud, a través del uso de las TIC, en situaciones en las que el profesional de salud y el paciente no están en la misma ubicación. Se trata de la transmisión segura de datos médicos e información, a través de texto, sonido, imágenes u otras formas necesarias para la prevención, diagnóstico, tratamiento y seguimiento de pacientes”.

El Ministerio de Salud de Francia define a la telemedicina como “la telemedicina es una práctica médica a distancia utilizando tecnologías de la información y la comunicación”.

Según la Asociación Americana de Telemedicina, ATA “la telemedicina es el intercambio de información médica de un lugar a otro, utilizando las vías de comunicación electrónica, para la salud y educación del paciente o el proveedor de los servicios sanitarios y con el objetivo de mejorar la asistencia del paciente”⁴

³ Guerrero, J. A. (Abril de 2011). *Definición de telemedicina*. Consultado el 26 de Mayo del 2014 , de http://biblioteca.usac.edu.gt/tesis/08/08_0513_CS.pdf

⁴ Ramos, C. M. (2009). *Telemedicina, Aspectos Generales*. Consultado el 26 de Mayo de 2014, de <http://www.revistareduca.es/index.php/reduca/article/viewFile/7/4>

2.1.4 Clasificación de la Telemedicina

Gracias a las tecnologías de la Información y la comunicación (TIC), los médicos y otros profesionales pueden consultarse en tiempo real por teléfono o videoconferencia, la transmisión de imágenes cumple un papel importante en las consultas a distancia. Basados en el tipo de aplicación de la telemedicina, y del servicio o de especialidad médica seleccionada, se la clasifica de la siguiente manera:

2.1.4.1 Por tipo de Servicio

Dependiendo del servicio a ofrecer, la telemedicina la podemos clasificar en:

Teleconsulta.- Es la consulta a través de sistemas de videoconferencia con un médico de atención primaria o de medicina especializada que pueden estar ayudados o no de pruebas complementarias.

Telediagnóstico.- Los diagnósticos pueden ser los resultantes de una primera consulta o de una sucesiva, por el mismo médico, interconsulta o segunda opinión.

Telecuidado o Teleatención.- Es el cuidado de pacientes en casa asistidos por enfermeras remotas, monitorizados a distancia por el profesional y con respuesta – botón de pánico en caso de alerta en el estado del paciente, que son empleados con frecuencias en personas con impedimentos sensoriales y ayuda a personas mayores.

Teleurgencias.- Es la transmisión de signos vitales en línea desde el medio de transporte del paciente (ambulancia, helicóptero).

Telemetría o Telemedida.- permite el monitoreo de signos vitales: electrocardiograma (ECG), electroencefalografía (EEG), electromiografía (EMG), presión arterial (PA), temperatura, pulso – oximetría, espirometrías y exámenes de laboratorio mediante punción digital para el control de enfermedades metabólicas que requieren monitoreo frecuente.

Teletutoría.- Es la dirección de actos médicos y quirúrgicos realizados por especialistas.

Teleeducación.- Es el entrenamiento y educación médica continuada a profesionales, estudiantes, población en general o a pacientes distantes geográficamente, esto se lo puede realizar en tiempo real o diferido.

Teleadministración.- se aplica a los sistemas de gestión de salud, para realizar a distancia la administración de procesos tales como el control de citas, remisiones, derivaciones, facturación, control de cartera, inventarios, planeación estratégica y orientación al usuario con el propósito de mejorar la calidad.

Teleterapia.- Es posible realizar consulta y tratamiento de pacientes empleando la videoconferencia para: Telepsiquiatría, telefisioterapia, teleoncología, teleprescripción.

Telefarmacia.- por medio de diversos sistemas se pueden realizar procesos de prescripción, dispensación, facturación y seguimiento de tratamientos pautados a pacientes, evitando su desplazamiento.

Telecirugía.- Permite la realización de intervenciones quirúrgicas por medio de robots manejados de forma remota.

Teleinterconsulta.- Es la consulta, orientación e intercambio de criterios entre especialistas.

2.1.4.2 Por Especialidad

Dependiendo la especialidad que se desea emplear, la telemedicina se la divide en:

Teleoncología.- Es la aplicación de las telecomunicaciones para proporcionar servicios oncológicos.

Teledermatología.- No es una tecnología, consiste en el diagnóstico y tratamiento clínico de problemas dermatológicos a distancia.

Teleradiología.- Es una de las especialidades que más se utilizan en la telemedicina. Consiste en la transmisión de imágenes radiológicas a través de redes de comunicación.

Esto se debe a que generalmente un radiólogo no tiene contacto con el paciente, lo que la convierte en la más propicia, adicionando a que hoy en día la mayoría de la captura de imágenes para analizarla está en formato digital.

Telepatología.- Se trabaja a partir de imágenes digitales o de video, obtenidas directamente desde el ocular del microscopio.

Telecardiología.- Con el desarrollo de las nuevas TIC, se pueden realizar a distancia ECG, ecocardiogramas, angiografías, ruidos cardíacos, entre otros.

Teleoftalmología.- Se puede realizar en parte a través de oftalmoscopios conectados a un sistema de videoconferencia que digitalice las imágenes de video para el diagnóstico en el fondo del ojo.

TeleORL.- Se pueden realizar exámenes de oídos, nariz y garganta, a través de sistemas de endoscopia de fibra óptica que digitalice las imágenes de video para el diagnóstico.⁵

2.1.5 La telemedicina en el Ecuador

Ecuador al igual que otros países tiene problemas en la dotación de servicios médicos, particularmente en unidades de salud distantes a los centros de tercer nivel, ya sea por la escasez de recursos, insuficiencia de especialistas, el aislamiento de unidades de salud remota y las dificultades de comunicación interna por la gran diversidad geográfica.

La telemedicina se plantea como una solución a cada uno de estos inconvenientes. En la actualidad representa una herramienta tecnológica de gran valor para vencer los obstáculos que impiden el acceso a los servicios de salud especializados, especialmente en zonas alejadas a centros de salud u hospitales. Su implementación permitirá que cualquier ecuatoriano por distante y dispersa que se encuentre su comunidad, puede tener acceso a nuevas tecnologías con eficiencia, equidad y la calidad deseada sin que la distancia sea un impedimento.

⁵ Kuzmar, I. (2013). *Cómo crear un servicio de telemedicina*. España, España. Consultado el 28 de Mayo de 2014

Hoy en día mucha información médica puede ser transmitida desde el nivel local hacia un centro médico principal para su interpretación y asesoría sobre el tratamiento que necesite el paciente. Esto se ha visto favorecido en gran medida por las nuevas tecnologías de información y comunicación entre las cuales el internet ha sido un gran apoyo convirtiéndose en el principal medio de telecomunicación global.

Las perspectivas de crecimiento de esta área tecnológica de la medicina son ilimitadas y van a la par con el desarrollo de las telecomunicaciones e informática. En Ecuador se debe competir con otras necesidades urgentes en los servicios de salud y los recursos con los que cuentan para su aplicación son escasos.

A escala global existe un impacto más profundo en naciones pobres, ya que albergan a gran cantidad de poblaciones marginales que se favorecerían con el servicio, ya que para empezar no hace falta grandes recursos, la estructura más sencilla para organizar un tele consultorio requiere conexión a una red desde el lugar donde notifica un paciente que acude a consulta, una central desde donde un médico atiende a dicho paciente y lo más importante la voluntad de servir de médicos y enfermeras.

Se debe tomar en cuenta que no siempre las condiciones son las más favorables, a veces existen inconvenientes que se deben tomar en cuenta como: falta de conocimiento en el uso de nuevas tecnologías, carencia de infraestructura, altos costos en los servicios de comunicación, etc. Pero el reto es mucho mayor ya que se pretende lograr que la tecnología con la que cuentan muchos centros de enseñanza se la aproveche mediante convenios con hospitales públicos bien dotados y que cuentan con médicos especialistas para dar servicio a personas de recursos limitados mediante la creación de pequeños centros de tele consulta y así ofrecer un servicio de salud a la mayor cantidad de población posible, servicios

médicos que sin duda alguna serán de buena calidad en menor tiempo y a menor costo.⁶

2.1.6 Beneficios de la Telemedicina

La implementación de sistemas de Telemedicina, lleva consigo una serie de beneficios para los hospitales, centros de salud y/o clínicas, en relación a los servicios brindados. A continuación se describen algunos de los beneficios:

- Ampliación de cobertura en la prestación de servicios médicos
- Mejora en la calidad de los servicios brindados
- Diagnósticos y tratamientos más oportunos
- Atención continuada
- Automatización de procesos en la atención de pacientes
- Disminución de riesgos profesionales
- Posibilidad de interconsulta
- Disminución de tiempos de respuesta de los centros médicos
- Disminución de costos en la utilización de equipos
- Disminución de tiempo y costos de traslado de médicos especialistas
- Disminución de tiempo y costos de traslado de pacientes⁷

Presión Arterial

“La Tensión o Presión Arterial es la fuerza que ejerce la sangre sobre las paredes de los vasos sanguíneos, y viene determinada por dos factores principales: la cantidad de sangre que tenemos y el calibre de los vasos sanguíneos por los que

⁶ Lopez, R. (2009). *Telemedicina en el Ecuador*. Consultado el 28 de Mayo de 2014, de http://telemedicina.cedia.org.ec/dmdocuments/Programa_Nacional_de_Telemedicina_y_Telesalud-Logros_planes_retos_pendientes_%28Ecuador%29.pdf

⁷ Milián, J. G. (Abril de 2011). *Beneficios de la Telemedicina*. Consultado el 29 de Mayo de 2014, de http://biblioteca.usac.edu.gt/tesis/08/08_0513_CS.pdf

circula. Para medirla se usan dos valores o cifras, el valor más alto es la presión sistólica o Máxima, el menor valor es la presión diastólica o Mínima”.⁸

2.1.7 Presión arterial sistólica

La presión arterial sistólica se refiere a la fuerza que se crea por la presión ejercida sobre las arterias cuando el corazón se contrae y bombea la sangre a través de las arterias. La presión arterial sistólica es generalmente considerada normal si está por debajo de 120, mientras que una presión sistólica mayor que 140 es considerada como presión sanguínea alta o hipertensión. Si una persona tiene una cifra de presión arterial sistólica entre estos dos números, por lo general se considera como límite para la presión arterial alta. En ese caso es aconsejable que se realice una monitorización regular la presión arterial.

2.1.8 Presión arterial diastólica

La presión arterial diastólica se entiende como la presión que está presente en las arterias entre latidos cardíacos, cuando el corazón está en reposo. Cuando se hace la lectura en el aparato medidor de presión es el número inferior. Es normal una lectura de la presión arterial diastólica si habitualmente está en un valor entre 80 o menos, mientras que una lectura que marca mayor que 90 se considera que existe una presión arterial alta. Ambas cifras suelen ser motivo de preocupación, lo que se recomienda hacer en estos casos es llevar con regularidad un control de la presión arterial.

⁸ Madrid, C. d. (Septiembre de 2008). Guía de Recomendaciones al Paciente: Hipertensión Arterial. 8. Consultado el 02 de Junio de 2014, de <http://www.madrid.org/cs/Satellite?blobcol=urldata&blobheader=application%2Fpdf&blobheadern ame1=Content-Disposition&blobheadervalue1=filename%3DHipertensi%C3%B3n+arterial.pdf&blobkey=id&blobtable=MungoBlobs&blobwhere=1271566209560&ssbinary=true>

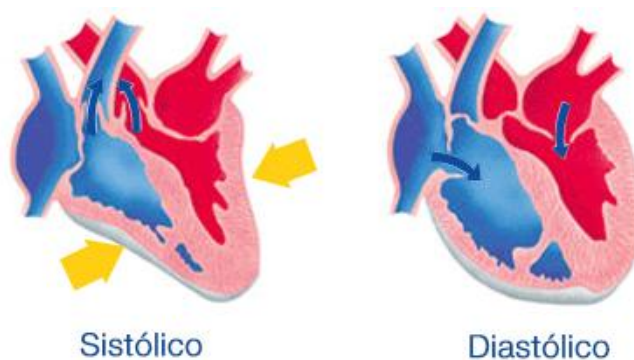


Figura 3: Presión Sistólica y Diastólica

2.1.9 Presión sanguínea diastólica y sistólica saludable

Se considera normal en la mayoría de los casos que la presión sistólica y diastólica tenga un valor menor de 120/80. Es aconsejable cuando estas cifras son mayores revisar la presión arterial con cierta regularidad, esto se lo puede realizar desde la casa con la ayuda de un medidor de presión arterial o visitando a su médico de confianza para que le ayude con la medición. Cuando la presión arterial tiene un valor superior a 140/90 indica que se tiene hipertensión y se debe seguir un tratamiento.

2.1.10 Métodos de Medición de la Presión Arterial

Para medir la presión arterial se tiene algunos métodos que se clasifican en:

- Pulso
- Cambio de color capilar.
- Auscultatorio.
- Oscilométrico.
- Ultrasonico / Doppler.
- Intraarterial o Directo.⁹

⁹ *Presion Sanguinea Sistólica y Diastólica.* (s.f.). Consultado el 02 de Junio de 2014, de <http://es.euroclinux.net/presion-sanguinea-sistolica-y-diastolica.html>

2.1.11 Qué Hacer Cuando se Mide La Presión Arterial

Se mide la presión máxima y mínima que ejerce la sangre a su paso por las arterias, para lo cual se puede utilizar métodos internos y externos. Externos como el manguito de goma e internos como el catéter intrarterial.

Al utilizar el manguito de goma se toma por comparación la tensión, ya que lo que se mide es la presión que hay dentro del manguito cuando se escucha una serie de ruidos en la arteria que se colapsa, o también el aspecto que toma la piel al cambio de irrigación sanguínea arteriocalilar, así también como la sensación que se siente al presionar la arteria con nuestros dedos al tomar el pulso arterial descubriendo la permanencia o ausencia del mismo en función de la presión de dicho manguito. Se toma como base las observaciones que se ha manifestado, es así que se puede describir cada método y sus limitaciones.

2.1.12 MÉTODO DE CAMBIO DE COLOR CAPILAR

Consiste en dejar isquémico la zona distal de un miembro con la aplicación de presión con un manguito de goma, midiendo la presión de dicho manguito con un anemómetro de mercurio o aneroide.

Procedimiento: Se debe elevar la presión del manguito hasta que la zona distal del miembro esté pálida, por efecto de dicha presión.

Después hay que bajar lentamente la presión de dicho manguito hasta observar que la piel de dicha zona distal se hace sonrosada. La PRESIÓN MEDIA DINÁMICA es la cifra que se observa en el esfigmomanómetro en el momento del cambio de color.

2.1.13 MÉTODO DEL PULSO

Consiste en hacer desaparecer y aparecer de nuevo el pulso de una arteria que ha sido colapsada con un manguito de goma.

Procedimiento:

Colocar un manguito de goma alrededor de un miembro mientras se observa el pulso en su zona distal (por debajo del manguito). Elevar la presión del manguito hasta que desaparezca el pulso y posteriormente se baja lentamente la presión observando el momento en que vuelve a observarse el pulso. La PRESIÓN MÁXIMA es la cifra que ese momento aparece en el esfigmomanómetro

2.1.14 MÉTODO AUSCULTATORIO

Consiste en hacer desaparecer los sonidos del pulso, luego de colapsar una arteria con un manguito de goma y posteriormente volverlos a escuchar determinando la tensión arterial máxima y mínima, según la naturaleza de dichos sonidos. Para este método se debe colocar un manguito de goma el mismo que está contenido dentro de una camisa de tela, alrededor de una extremidad fijándolo entre sí con correas, luego se debe elevar la presión del aire contenido en la citada bolsa de goma hasta conseguir que presione el miembro todo a su alrededor y con él a las arterias que suministran riego sanguíneo a la extremidad, se observa por audición con un fonendoscopio los sonidos que se originan por los cambios de régimen laminar a régimen turbulento de la sangre que circula por las arterias de dicha extremidad. Este es el método más usado y también el más investigado y se mide en mmHg la presión que se origina en el interior del manguito.

2.1.15 MÉTODO OSCILOMÉTRICO

Consiste en observar las oscilaciones que se muestran en un oscilómetro al colapsar una arteria y luego volverlos a retomar, logrando de esta manera determinar la tensión máxima y mínima en función de las características de las oscilaciones. Es un método menos exacto que el método auscultatorio.

En la medición oscilométrica, se usa un sensor de presión electrónico con lectura numérica de la presión arterial. El manguito se infla y libera mediante una bomba y válvula operadas eléctricamente. Se puede adaptar al brazo o la muñeca.

2.1.16 MÉTODO ULTRASÓNICO / DOPPLER

Se utiliza el ultrasonido para determinar las presiones máxima y mínima de la sangre a su paso por las arterias. Los valores quedan registrados digitalmente y solamente se utiliza en servicios especializados ya que su precio es elevado.

Procedimiento: Colocar la sonda arterial sobre la arteria a observar y por efecto Doppler se mide el valor de la tensión arterial máxima y mínima. El efecto Doppler se basa en la variación de la frecuencia de la onda debida al movimiento relativo entre el emisor y el observador.

2.1.17 MÉTODO DIRECTO O INTRAARTERIAL

Consiste en medir la presión máxima, mínima y media dinámica de la sangre al paso por la arteria canalizada a través de un transductor que hace transformar dichas presiones en dígitos observables. Es indispensable heparinizar las vías para

su correcto funcionamiento, es un método traumático pero a la vez el más exacto y solo se utiliza en monitorizaciones en Unidades de Cuidados Intensivos.

Los métodos más utilizados son el **Auscultatorio** y el **Oscilométrico**, métodos que se utilizan de una forma absoluta en la práctica asistencial y principalmente por enfermeras, técnica también utilizada por el médico, él mismo que suele delegarla al profesional enfermero. El método Auscultatorio por considerarse el menos traumático suele ser el más usado, el más exacto y el más estudiado e investigado.

Hipertensión Arterial

La hipertensión es conocida también como tensión arterial alta, es un trastorno en que los vasos sanguíneos tienen una tensión persistentemente alta. Los vasos sanguíneos trasladan la sangre desde el corazón a todas las partes del cuerpo, es así que cada vez que late el corazón bombea sangre a los vasos. La tensión arterial es la fuerza que ejerce la sangre contra las paredes de los vasos (arterias) al ser bombeada por el corazón. Mientras más elevada sea la tensión, el corazón necesita realizar un mayor esfuerzo para bombear la sangre.

La tensión arterial normal en adultos es de 120 mm Hg cuando late el corazón (tensión sistólica) y de 80 mm Hg cuando se relaja el corazón (tensión diastólica). Se considera que la tensión arterial es alta cuando la tensión sistólica es igual o superior a 140 mm Hg y/o la tensión diastólica es igual o superior a 90 mm Hg.

La hipertensión causa síntomas como dolor de cabeza, dificultad respiratoria, vértigos, dolor torácico, palpitaciones del corazón y hemorragias nasales, pero por lo general la mayoría de los hipertensos no presentan síntoma alguno.¹⁰

2.1.18 Por qué es peligrosa la hipertensión arterial

Mientras más alta es la tensión arterial el riesgo de daño al corazón es mayor así como también para los vasos sanguíneos de órganos principales como el cerebro y los riñones.

Si la hipertensión no es controlada puede provocar un infarto de miocardio, un ensanchamiento del corazón que esto a su vez con el pasar del tiempo puede provocar una insuficiencia cardíaca. Los vasos sanguíneos pueden desarrollar protuberancias (aneurismas) y zonas débiles que los hacen más susceptibles de obstruirse y romperse. La hipertensión también puede provocar deficiencia renal, ceguera y deterioro cognitivo además puede ocasionar que la sangre se filtre en el cerebro provocando un accidente cerebrovascular.

La hipertensión puede provocar consecuencias severas para la salud a eso se añade que existen otros factores como el consumo de tabaco, una dieta poco saludable, el uso nocivo del alcohol, la inactividad física y la exposición a un estrés permanente, así como la obesidad, el colesterol alto y la diabetes mellitus que pueden agravar y aumentar las probabilidades de sufrir un infarto de miocardio, un accidente cerebrovascular o insuficiencia renal.

¹⁰ Zambrano, A. C. (s.f.). *Medición de la Tensión Arterial*. Consultado el 05 de Junio de 2014, de http://www.bibliotecadigitalcecova.com/contenido/revistas/cat7/pdf/manual_2.pdf

2.1.19 La hipertensión un problema frecuente

Uno de cada tres adultos a nivel mundial tiene la tensión arterial elevada, perturbación causante de aproximadamente la mitad de todas las defunciones por accidente cerebrovascular o cardiopatía. Se considera que este problema fue la causa directa de 7,5 millones de defunciones en el 2010, lo que representa casi el 13% de la mortalidad mundial.

En casi todos los países de ingresos altos, el diagnóstico y tratamiento generalizado de esas personas con medicamentos de bajo costo ha propiciado una extraordinaria reducción de la tensión arterial media en todas las poblaciones, lo que ha contribuido a reducir la mortalidad por enfermedades del corazón.

Por ejemplo, en 1980, casi un 40% de los adultos de la Región de Europa de la OMS y un 31% de los adultos de la Región de las Américas padecían hipertensión. En 2010 la cifra había caído a menos del 30% y el 23%, respectivamente. En los países en desarrollo, existen personas que padecen de hipertensión sin estar diagnosticadas, lo cual les ha privado de un tratamiento que les ayudaría a reducir su riesgo de defunción y discapacidad por cardiopatía o accidente cerebrovascular.

2.1.20 RIESGOS Y COMPLICACIONES DE LA HIPERTENSIÓN ARTERIAL

- El principal riesgo que puede ocasionar la hipertensión es el Infarto de Miocardio, cuando un paciente hipertenso no es tratado tiene como media 10 veces más de riesgo de morir de infarto que un individuo con una tensión normal.

- A nivel cerebral puede producir trombos o roturas arteriales dando lugar a hemorragias con daño en las células nerviosas.
- A nivel renal puede producir insuficiencia renal que si no es tratada puede tener problemas de salud.
- A nivel del ojo puede producir hipertensión y rotura de los vasos oculares con la consiguiente pérdida de visión.

2.1.21 CAUSAS

- El mecanismo de la hipertensión arterial es desconocido, y es nombrada como “hipertensión esencial”, “Primaria” o “Idiopática”, es decir sin causa previa conocida.

Según la Organización Mundial de la Salud OMS.

- Puede ser originada por alguna otra enfermedad, en cuyo caso se denomina “hipertensión arterial secundaria”, como por ejemplo, las originadas por enfermedades endocrinas y del riñón.
- En la hipertensión arterial primaria no se conocen sus causas específicas, pero se relación a factores que suelen estar presente en la mayoría de personas.
- Se tiene que separar aquellos factores relacionados con la herencia, sexo, edad y raza, que se consideran factores poco modificables, de aquellos que sí son modificables y tiene que ver con los hábitos de vida, como son: un estilo de vida sedentario, el alcohol, la sensibilidad a la sal, la obesidad, y el uso de anticonceptivos.

2.1.22 SÍNTOMAS

- La hipertensión arterial generalmente no presenta síntomas.

2.1.23 DIAGNÓSTICO

Se puede hacer a partir de tres vías:

- Mediante revisiones periódicas.
- Historia familiar y antecedentes personales.
- Exploración física y pruebas complementarias.

2.1.24 TRATAMIENTO

Para tratar de controlar la tensión arterial las personas deben modificar el modo de vida: realizar ejercicio diario, adoptar una dieta saludable, evitar el alcohol y abandonar el consumo de tabaco, reducir la ingesta de sal. Para otras personas, estos cambios les resultan insuficientes y necesitan tomar medicamentos con prescripción médica.

Tomando la medicación prescrita y llevando un control de la salud se puede contribuir notablemente con al tratamiento. Las personas con hipertensión que tienen también un alto nivel de azúcar en la sangre están expuestos a un riesgo mayor de sufrir un infarto de miocardio o un accidente cerebrovascular, por lo que es indispensable realizar revisiones periódicas de la cantidad de colesterol y azúcar en la sangre, además del nivel de albúmina en la orina.

Para lograr minimizar las probabilidades de padecer hipertensión y sus consecuencias adversas se puede adoptar cinco medidas:

- Hay que tener en cuenta que en la mayoría de los casos “la hipertensión no puede curarse sólo puede controlarse”.

- El tratamiento de la hipertensión arterial se basa en los siguientes puntos:
 - Dieta
 - Ejercicio
 - Medicación.

- Se debe tomar la medicación indicada de forma correcta, no se puede olvidar tomar el medicamento a diario o abandonar el tratamiento por cuenta propia.

2.1.25 Dieta:

- Se debe tomar una dieta pobre en sal. La sal excesiva en la dieta causa retención de líquidos y aumento de la tensión arterial. No se debe tomar los siguientes alimentos por su elevado contenido en sodio:
 - Sal de cocina o de mesa.
 - Pan y biscotes con sal.
 - Carnes saladas, ahumadas
 - Aceitunas y salazones.
 - Pescados Ahumados y secos.
 - Frutos secos.
 - Moluscos.
 - Conservas.
 - Productos de Charcutería.
 - Condimentos y salsas saladas.
 - Quesos muy curados.

- Caldos preparados y sopas de sobre.

- Las necesidades diarias de sal están en torno a medio gramo al día (media cucharadita), pero en la dieta normal se consume ocho veces más dicha cantidad.
- Reducir la ingesta de alcohol.
- Reducir el consumo de café. No más de 2 o 3 cafés al día.
- Consumir alimentos ricos en potasio, como frutas, verduras y legumbres.
- Evitar el colesterol y los ácidos grasos saturados (grasas animales), se los puede cambiar por los ácidos grasos poliinsaturados, sobre todo los omega- 3 (que aparecen en los aceites vegetales como el de oliva y en el pescado).

2.1.26 Ejercicio:

- Un programa de ejercicio saludable ayuda a fortalecer el corazón, a bajar peso y a controlar la tensión arterial.
- Se debe realizar un ejercicio adecuado a cada edad y cada persona. La realización de media hora de ejercicio suave de 2 a 3 días por semana es suficiente para la mayoría de las personas.

2.1.27 Medicación:

- El tratamiento debe ser siempre individualizado. El médico a la hora de prescribir un fármaco debe tomar en cuenta el historial clínico y los posibles efectos secundarios del paciente, además de las pruebas necesarias para controlar la seguridad del mismo.
- Habitualmente los fármacos prescritos se toleran bien por parte de los pacientes, pero es posible que puedan dar efectos secundarios. Si estos

aparecen, se debe informar al médico para que ajuste la dosis o cambie de medicamento.

- En ningún caso se debe suspender la medicación sin conocimiento del médico o del equipo que le atiende.

2.1.28 TENSIÓN ARTERIAL

- Una parte importante del tratamiento de la hipertensión es el control de la misma, que se puede realizar en casa por medio de aparatos que se comercializan para el uso personal, son electrónicos y fáciles de usar.
- Un chip electrónico detecta el pulso y un lector digital lee la tensión arterial.
- Para tomarse la tensión arterial, se debe seguir las siguientes recomendaciones:
 - Se debe sentar tranquilamente durante cinco minutos.
 - Estar relajado.
 - El brazo debe estar apoyado sobre una superficie.
 - No haber fumado ni tomado café, ni estar recién comido.
 - Estar con la vejiga urinaria vacía.
 - Tener las piernas sin cruzar.
 - La ropa no debe presionarle el brazo.

Se debe acudir a los servicios sanitarios cuando:

- Se perciba sensación de mareo intenso.
- Presente fuertes dolores de cabeza.
- Si se ha tomado la tensión y tiene cifras muy elevadas.

2.1.29 RIESGO DE TENER PRESIÓN ARTERIAL ALTA

Esta lista puede ayudar a saber si se corre riesgo de tener presión arterial alta.

- Ser mayor de 50 años
- Tener sobrepeso
- Fumar
- Tener diabetes.
- Tener enfermedad renal
- Ser afroamericano
- Ser hispano o latino
- Ser hombre y beber más de 1 onza de alcohol por día
- Ser mujer y beber más de media onza de alcohol por día
- Un miembro de la familia (madre, padre u otro) tiene presión arterial alta.¹¹

¹¹ Madrid, C. d. (Septiembre de 2008). Guia de Recomendaciones al Paciente: Hipertensión Arterial. 8. Consultado el 06 de Junio de 2014, de <http://www.madrid.org/cs/Satellite?blobcol=urldata&blobheader=application%2Fpdf&blobheadername1=Content-Disposition&blobheadervalue1=filename%3DHipertensi%C3%B3n+arterial.pdf&blobkey=id&blobtable=MungoBlobs&blobwhere=1271566209560&ssbinary=true>

CAPÍTULO III

Sistema de Comunicación GSM

3.1.1 Tecnología GSM

GSM (Global System for Mobile Communication) es el nombre de un grupo de estandarización establecido en 1982 para la creación de un estándar habitual de telefonía móvil europeo para un sistema de radio que opere a 900 MHz, es un sistema estándar completamente definido, usado para la comunicación entre teléfonos móviles basada en la tecnología digital.

En 1989 el compromiso de GSM se trasladó al ETSI (European Telecommunications Standards Institute) y las primeras especificaciones se las publicaron en 1990. GSM emprendió su servicio de forma comercial a mediados de 1991, y en 1993 ya había 36 redes GSM en 22 países. Cuando los móviles usan la tecnología GSM se les conoce también por móviles 2g o de segunda generación, tomando en cuenta que su principal función es la telefonía.

En la actualidad es el sistema básico para todas las comunicaciones móviles y el más usado, pero debido por sus limitaciones técnicas está rápidamente siendo reemplazado en los países avanzados por los nuevos sistemas 3g y 4g, que usan el estándar UTMS, de mayor rapidez y prestaciones. El GSM apenas permitía una velocidad de descarga de datos de 100 kbps.

El estándar GSM presta los siguientes servicios:

- Transmisión y recepción de voz y datos
- Envío y recepción de mensajes cortos de texto (SMS)
- Servicios complementarios (llamada en espera, multi conferencias, identificador de llamadas, entre otros).¹²

3.1.2 Origen y Evolución de GSM

- 1982 Creación del Groupe Spéciale Mobile (GSM) en la CEPT. Reserva de frecuencias en Europa para el futuro sistema.
- 1987 Selección de características generales: tecnología digital, acceso múltiple TDMA.

¹²Sandoval, J. (Octubre de 2013). *Definición de GSM*. Consultado el 08 de Junio de 2014, de <http://186.42.96.211:8080/jspui/bitstream/123456789/1650/1/Sandoval%20Perugachi%20Juan%20Roberto.pdf>

- 1989 La estandarización del GSM pasa a depender del ETSI
- 1990 Termina la especificación de la fase 1: voz, SMS, datos en modo circuito, roaming internacional, servicios suplementarios (desvío, restricción de llamadas).
- 1992 Comienzo de operación en algunos países.
- 1996 Fase 2. Mejoras; DCS-1800 (luego GSM-1800).
- 1998 Fase 2+ (varias releases). Mejoras: mayor velocidad de datos; HSCSD; GPRS.
- 1998 La estandarización pasa realizarse en el 3GPP, en el cual se integra el ETSI.
- 2004 1000 millones de usuarios de GSM.
- 2006 2000 millones de usuarios de GSM.¹³

¹³ *Sistema GSM*. (s.f.). Consultado el 08 de Junio de 2014, de <http://ocw.upm.es/teoria-de-la-senal-y-comunicaciones-1/comunicaciones-moviles/contenidos/tema-6>

3.1.3 Arquitectura GSM

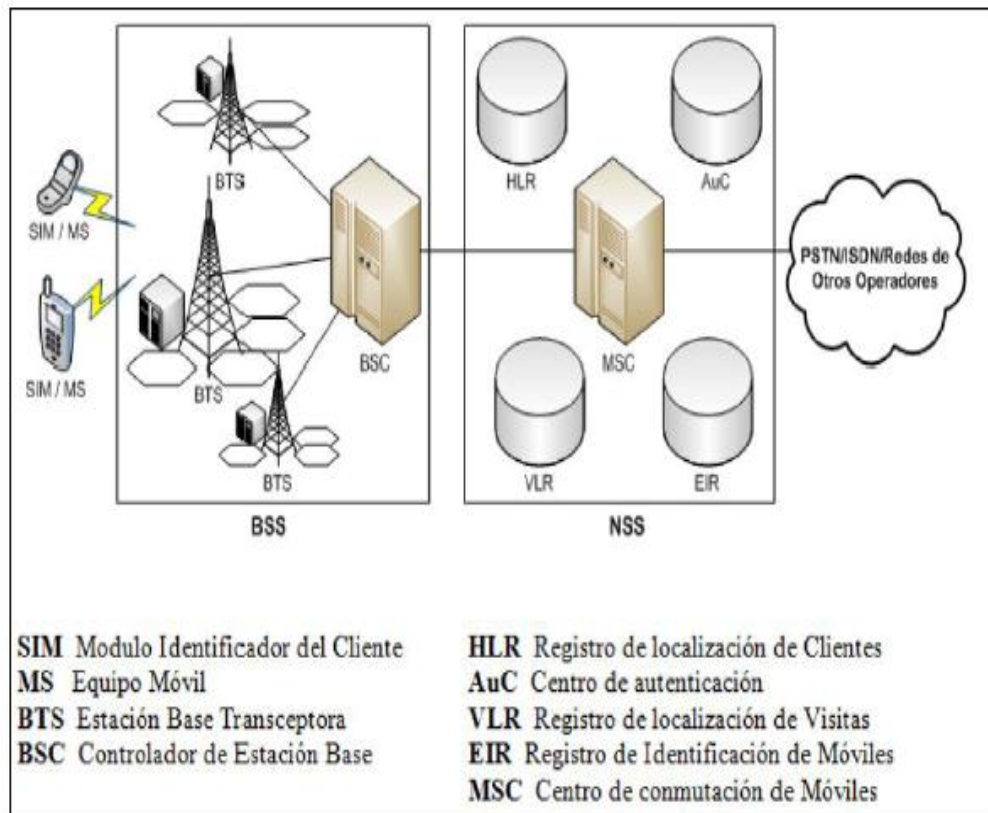


Figura 4: Arquitectura GSM

Una red GSM está constituida por tres elementos: el terminal, la estación-base (BSS) y el subsistema de red o nudo. Adicionalmente existen centros de operación establecidos por las operadoras, para monitorizar el estado de la red.

Base Substation System (Sistema de Subestación de Base)

Network Subsystem (Subsistema de Red)

TRX: Transceiver (Transrecibidor)

EIR: Equipment Identity Register (Registro de Identificación del Equipo)

MS: Mobile Station (Estación Móvil)

AC: Authentication Center (Central de Autenticación)

SIM: Subscriber Identity Module (Módulo de Identificación de Suscriptor)

HLR: Home Location Register (Registro de Localización de Llamada)

BTS: Base Transceiver Station (Estación Transreceptor de Base)

BSC: Base Station Controller (Estación Base de Control)

MSC: Mobile services Switching Center (Central Intercambiadora de Servicios Móviles)

VLR: Visitor Location Register (Registro de Localización del Visitante)

ISDN: Integrated Services Digital Network (Red Digital de Servicios Integrados)

PSTN: Public Switched Telephone Network (Red Telefónica Analógica Pública)

SMSC: Short Message System Center (Central de Sistema de Mensajes Cortos)

La estación móvil o también conocida como terminal contiene la tarjeta SIM, que se utiliza para identificar al usuario dentro de la red. El SIM confiere movilidad personal al usuario de la tarjeta, permitiéndole acceder a los servicios de la red independientemente del teléfono móvil que use o su localización. Puede ser protegido a través de un código (PIN) contra un uso indebido. Existe además un número que identifica cada terminal individualmente, el International Mobile Subscriber Identity (IMEI), pero que es independiente del SIM.

La estación-base es la encargada de controlar la conexión radio entre el teléfono móvil y la red, se le conoce como célula debido a que cubre una determinada área geográfica. Una BSS está compuesta por dos elementos: el BTS (Base Transceiver Station) y el BSC (Base Station Controller), cada BSS puede tener uno o más BTS. Las que albergan los equipos de transmisión / recepción (los TRX o transceivers) y gestionan los protocolos de radio con el terminal móvil son los BTS.

Existen más BTS en zonas urbanas que en rurales, cada estación para permitir que varios usuarios se unan a la red utiliza técnicas digitales lo que permite también que se realicen y se reciban llamadas simultáneamente.

El BSC es encargado de administrar los recursos de radio de una o más BTS. Entre sus funciones se incluyen el handoff (ocurre cuando el utilizador se mueve de una célula para otra, permitiendo que la ligación se mantenga), los cambios de frecuencias y el establecimiento de los canales de radio utilizados. Finalmente, establece la ligación entre el móvil y el Mobile Service Switching Center (MSC) que es considerado como el corazón del sistema GSM.

Como ya se explicó anteriormente el MSC es el centro de la red, por medio del cual se ejecuta la ligación entre una llamada realizada de un móvil hacia las otras redes fijas (las analógicas PSTN o digitales ISDN) o móviles. El nudo en el que se encuentra posee además una serie de equipos destinados a controlar varias funciones, como el cobro del servicio, la seguridad y el envío de mensajes SMS.

El Home Location Register (HLR) contiene toda la información administrativa sobre el cliente del servicio y la localización actual del terminal, la red a través del HLR puede verificar si un móvil que se intenta ligar posee un contrato de servicio válido. Si logra obtener una respuesta afirmativa el MSC envía un mensaje de vuelta al terminal para informarle que se encuentra autorizado a utilizar la red. Es así que aparece en la pantalla el nombre de la operadora informando que se puede realizar y recibir llamadas. Para recibir una llamada destinada a un móvil el MSC se dirige al HLR para verificar la localización, simultáneamente el terminal de tiempos a tiempos envía un mensaje para la red para informar del sitio donde se encuentra (este proceso es denominado polling).

El Visitor Location Register (VLR) se utiliza para controlar el tipo de conexiones que un terminal puede hacer. Por ejemplo, si un utilizador posee restricciones en las

llamadas internacionales el VLR impide que estas sean ejecutadas, las bloquea y envía un mensaje de vuelta al teléfono móvil.

El Equipment Identity Register (EIR) y el Authentication Center (AC) son utilizados para garantizar la seguridad del sistema, el EIR posee una lista de IMEI de terminales que han sido declarados como robados o que no son compatibles con la red GSM.

Si se encuentra el móvil en la lista negro el EIR no va a permitir que se conecte a la red, dentro del AC existe una copia del código de seguridad del SIM. El AC genera un número aleatorio que es enviado para el móvil cuando existe una autorización correspondiente, número que es utilizado junto al código del SIM y un algoritmo de encriptación denominado A3 creando otro número que es enviado de nuevo para el AC. El utilizador es autorizado a usar la red únicamente si el número enviado por el terminal es igual al calculado por el AC.

Finalmente el Short Message System Center (SMSC) es el responsable de generar los mensajes cortos de texto.

3.1.4 Características del GSM

- Posibilidad de usar el terminal y la tarjeta SIM en redes GSM de otros países (roaming).
- Servicio de mensajes cortos (SMS) con hasta 126 caracteres que se pueden enviar y recibir.
- Reenvío de llamadas para otro número.
- Transmisión y recepción de datos y fax con velocidades de hasta 9.6 Kbps.

- Difusión celular - mensajes con hasta 93 caracteres que pueden ser enviados para todos los teléfonos móviles en un área geográfica, mensajes que recibe el terminal cuando aún no está listo.
- CLIP (Calling Line Identification Presentation) permite visualizar en pantalla el número que nos está llamando, gracias al CLIP se impide que el número llamante sea visto por alguien (anónimo).
- Notificación de llamadas en espera cuando se está hablando por teléfono.
- Posibilidad de colocar una llamada en espera mientras se coge otra.
- Encriptación de llamadas para impedir que sean escuchadas por otros.
- Posibilidad de impedir la recepción / transmisión de ciertas llamadas.
- Llamadas de emergencia que puede ser marcado en cualquier red, incluso sin SIM.
- Posibilidad para que varios utilizadores puedan hablar entre si al mismo tiempo.

3.1.5 Características técnicas

Las bandas asignadas por parte de la UIT (Unión Internacional de Telecomunicaciones) para el estándar GSM en la mayor parte del mundo está en el rango de: 900MHz y 1800MHz; en el Ecuador se utilizan las bandas de frecuencia de 850MHz y 1900MHz, a partir de estos datos, las principales características técnicas de la red GSM son:

- Canales: 124 canales, cada canal da un servicio de 8 a 16 usuarios a la vez.
- Bandas: Usa dos bandas de 25 MHz para transmitir y recibir información.
- Subida: Usa la banda de 824MHz a 849MHz.
- Bajada: Usa la banda de 869MHz a 894MHz.

- Ancho de banda: 20KHz.
- Modulación: GMSK (Gaussian Minimum Shift Keying).
- Velocidad máxima de canal de radio: 270.833Kbps.
- Duración de bit: 3,692 ms.
- Longitud de trama: 6,615 ms.
- Longitud de slot de tiempo: 577us.
- Codificación de voz: PRE-LPT 13Kbps.
- Potencia de salida: 20mW a 20W

3.1.6 Ventajas de GSM

- Encriptación para proporcionar confidencialidad en las comunicaciones.
- Autenticación del abonado.
- Potentes códigos de control de errores que permite mejorar la calidad de las telecomunicaciones.
- Simplificación de los equipos de radiofrecuencia.
- Mayor grado de portabilidad.
- Menor consumo.
- Mayor flexibilidad a la hora de incorporar los avances y desarrollos tecnológicos (codificación de voz a 6,5 Kb/s).
- Transmisión de voz y datos a diferentes velocidades.¹⁴

SERVICIO DE MENSAJES CORTOS DE TEXTO SMS

El servicio SMS permite trasladar un mensaje de texto entre una estación móvil (MS) y otra entidad (SME) por medio de un centro de servicio (SC).

¹⁴ *Sistemas Móviles GSM, CDMS, TDMA.* (s.f.). Consultado el 10 de Junio de 2014, de www.une.edu.ve/~iramirez/te1/sistemas_moviles.htm

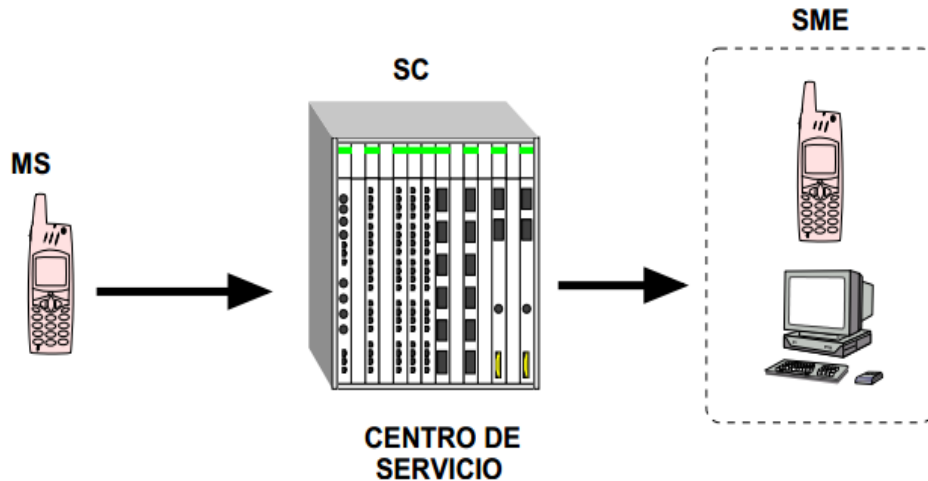


Figura 5: Proceso de envío de mensajes cortos

El servicio final que se ofrece es una comunicación extremo-extremo entre la estación móvil (MS) y la entidad (SME), puede ser otra estación móvil o puede estar situado en una red fija. Para solicitar algún tipo de servicio cuando se envía un mensaje un extremo es una estación móvil y la otra es un servidor que atiende las peticiones. Por ejemplo cuando se realiza alguna votación en algún programa de tv.¹⁵

3.1.7 Formato del SMS

Existen dos formas de enviar y recibir mensajes SMS, en modo texto y en modo PDU (Protocol Description Unit). El modo texto es una representación de la cadena de bits contenida en el campo de datos del modo PDU, pero no todos los teléfonos soportan el modo texto, sobre todo los teléfonos más antiguos.

Modo Texto

El envío de mensajes en modo texto es más sencillo ya que no requiere de ningún tipo de codificación del texto que se quiere enviar.

Modo PDU

¹⁵ Gomez, J. G. (Junio de 2002). *El Servicio SMS: un enfoque practico*. Consultado el 11 de Junio de 2014, de <http://www.learobotics.com/personal/juan/doctorado/sms/sms.pdf>

El modo PDU trata el SMS como una cadena de caracteres en octetos hexadecimales, de tal forma que de la codificación se obtiene como resultado el SMS en modo texto.

La ventaja que tiene el modo PDU en comparación con el modo texto es que en modo texto la aplicación queda limitada a la opción de codificación que se haya preestablecido, mientras que en modo PDU se puede implementar cualquier codificación.¹⁶

3.1.8 Arquitectura del SMS

La figura, muestra la estructura básica de la red SMS.

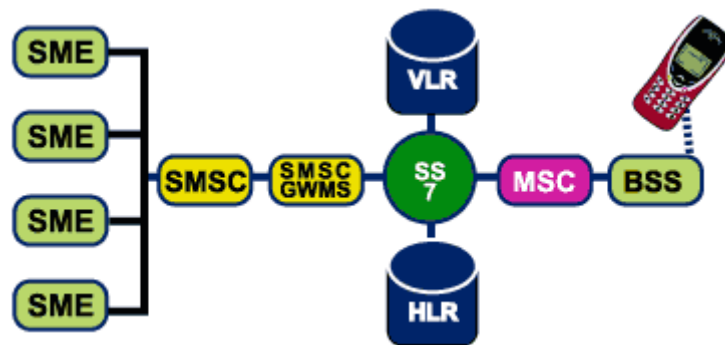


Figura 6: estructura básica de la red SMS

SME (Short Messaging Entity): Entidad que puede enviar o recibir mensajes cortos, pudiendo estar localizada en la red fija, una estación móvil, u otro centro de servicio.

SMSC (Short Message Service Center): Es el responsable de la transmisión y almacenamiento de un mensaje corto, entre el SME y una estación móvil.

SMS-Gateway/Interworking MSC (SMS-GMSC): Es un MSC capaz de recibir un mensaje corto de un SMSC, interrogando al HLR (Home Location Register) sobre la información de encaminamiento y enviando el mensaje corto al MSC.

¹⁶ *Formato de los Mensajes SMS.* (s.f.). Consultado el 11 de Junio de 2014, de <http://www.zonabot.com/28-formato-de-los-mensajes-sms.html>

HLR (Home Location Register): Es una base de datos usada para el almacenamiento permanente y gestión de los usuarios y el perfil del servicio. Cuando se realiza la interrogación del SMSC, el HLR le proporciona la información de encaminamiento para el usuario indicado.

MSC (Mobile Switching Center): Realiza funciones de conmutación del sistema y el control de llamadas a y desde otro teléfono y sistema de datos.

VLR (Visitor Location Register): Es una base de datos que contiene información temporal de los usuarios, la misma que necesita el MSC para dar servicio a los usuarios de paso.

BSS (Base Station System): Formada por el BSCs (base-station controllers) y por BTSs (base-transceiver stations), su responsabilidad es transmitir el tráfico de voz y datos entre las estaciones móviles.

MS (mobile station): Recibe y origina tanto mensajes cortos como llamadas de voz. El SMS hace uso del MAP (mobile application part), el cual define métodos y mecanismos de comunicación en las redes sin hilos, y usa el servicio del SS7 TCAP (transaction capabilities application part). Una capa del servicio SMS hace uso del MAP lo que permite la transferencia de mensajes cortos entre el par de entidades.¹⁷

COMANDOS AT

Los comandos AT son instrucciones codificadas que proporcionan la comunicación de los módems y les indican realizar una función determinada, permitiendo acceder a los datos, configuración y gestión de mensajes SMS de un móvil. La tecnología GSM utiliza este lenguaje como estándar para comunicarse con sus terminales, ubicando los comandos AT en el dispositivo y sin dependencia del canal de comunicación. Estos comandos se hicieron tan populares que se convirtieron en un estándar para la comunicación por módem.

¹⁷ *Elementos de red y Arquitectura.* (s.f.). Consultado el 11 de Junio de 2014, de http://www.gsmSpain.com/info_tecnica/sms/pagina3.php

3.1.9 Estructura de los comandos AT

Contiene una sintaxis definida por un prefijo en mayúsculas o minúsculas AT o at, el cuerpo del comando que será un conjunto de caracteres que dependerá del comando a usar y termina con un carácter de retorno de carro CR.

- ✓ Todos los comandos deben empezar por AT o at y terminar con un retorno de carro.
- ✓ Usar solo mayúsculas o solo minúsculas, no se admite combinación de estas.
- ✓ El máximo número de caracteres en un comando es de 128.
- ✓ Dentro de la sintaxis está incluido los espacios simplemente con la finalidad de incrementar la claridad pero estos en realidad son ignorados.
- ✓ Los caracteres que precedan a AT son ignorados.
- ✓ Ctrl-x puede ser usado para abortar la entrada en la línea de comandos.¹⁸

3.1.10 Comandos AT para modem GSM o teléfono celular

Los comandos más comunes utilizados son los que se presentan en la siguiente lista:

Comandos Generales

- ✓ **AT+CGMI:** Identificación del fabricante
- ✓ **AT+CGSN:** Obtener número de serie
- ✓ **AT+CIMI:** Obtener el IMSI.
- ✓ **AT+CPAS:** Leer estado del modem.

Comandos del Servicio de Red

- ✓ **AT+CSQ:** Obtener calidad de la señal
- ✓ **AT+COPS:** Selección de un operador
- ✓ **AT+CREG:** Registrarse a una red
- ✓ **AT+WOPN:** Leer nombre del operador

¹⁸ Jara, L. A. (2007). *Comandos AT*. Consultado el 11 de Junio de 2014, de <http://dSPACE.uazuay.edu.ec/bitstream/datos/2262/1/05777.pdf>

Comandos de Seguridad

- ✓ **AT+CPIN:** Introducir el PIN
- ✓ **AT+CPINC:** Obtener el número de reintentos que quedan
- ✓ **AT+CPWD:** Cambiar password

Comandos para la Agenda de Teléfonos

- ✓ **AT+CPBR:** Leer todas las entradas
- ✓ **AT+CPBF:** Encontrar una entrada
- ✓ **AT+CPBW:** Almacenar una entrada
- ✓ **AT+CPBS:** Buscar una entrada

Comandos para SMS

- ✓ **AT+CPMS:** Seleccionar lugar de almacenamiento de los SMS
- ✓ **AT+CMGF:** Seleccionar formato de los mensajes SMS
- ✓ **AT+CMGR:** Leer un mensaje SMS almacenado
- ✓ **AT+CMGL:** Listar los mensajes almacenados
- ✓ **AT+CMGS:** Enviar mensaje SMS
- ✓ **AT+CMGW:** Almacenar mensaje en memoria
- ✓ **AT+CMSS:** Enviar mensaje almacenado
- ✓ **AT+CSCA:** Establecer el centro de mensajes a usar
- ✓ **AT+WMSM:** Modificar el estado de un mensaje

3.1.11 Comandos AT para la configuración del modem GSM

La configuración del modem GSM se la realiza por medio de los siguientes comandos AT, que se detallan a continuación:

- **AT**, este es un comando de atención cuya función es monitorear si existe una buena conexión, si la conexión es buena el modem responde OK.
- **AT+CNMI=1,2,0,0,0**, este comando establece el formato del SMS que recibirá el modem. Cuando el modem recibe un nuevo SMS, en el HyperTerminal mostrará el siguiente encabezado en todos los SMS recibidos en el modem:

+ CMT: "61 xxxxxxxx",, "08/04/30, 23:20:00 40"

- **AT+CMGF=1**, este comando permite elegir el modo de interpretación de los datos por parte de modem, como se puede observar es igual a "1", esto quiere decir que todos los datos serán interpretados en modo de texto, es decir, la secuencia de caracteres que se envía al modem son ASCII normales. El modem al momento de recibir este comando responde OK, indicando que la petición ha sido aceptada.
- **AT+CMGS=**, este comando permite enviar un mensaje SMS. A continuación se muestra un ejemplo de la sintaxis de este comando:

```
AT+CMGS="99XXXXXXXX" alt 13 <CR>: Return Carrier ... (Enter)
```

```
> Modem Listo.... Alt 26... (ctrl+z)
```

```
Respuesta: +CMGS: xxx 19
```

Estudio de la tecnología Arduino

3.1.12 Definición de Arduino

Arduino es una plataforma de prototipos electrónica de código abierto (open-source) basada en hardware y software flexibles que son fáciles de usar. En la mayoría de ocasiones está considerado como hobby para cualquiera interesado en crear objetos o entornos interactivos.

Utilizando el Arduino Programming Language (basado en Wiring) y el Arduino Development Environment (basado en Processing) se puede lograr programar el microcontrolador de la placa. El ensamblaje de las placas se la puede realizar a mano o comprarlas pre ensambladas; el software que se utiliza se lo puede descargar gratuitamente. Se encuentran disponibles los diseños de referencia del hardware (archivos CAD) bajo licencia open-source, por lo que resulta fácil adaptarla a las necesidades que cada programador desee.²⁰

¹⁹ Gomez, J. G. (Junio de 2002). Comandos AT. Consultado el 11 de Junio de 2014, de <http://www.learobotics.com/personal/juan/doctorado/sms/sms.pdf>

²⁰ *Definicion de Arduino*. (s.f.). Consultado el 16 de Junio de 2014, de <http://arduino.cc/en/pmwiki.php?n=>

3.1.13 Características

- Es un tipo de controlador de código abierto y software de programación libre, muy versátil y accesible que cuenta con multitud de formatos.
- El chip Atmega es el corazón de la placa (existen versiones Atmega168 y Atmega328, este último tiene el doble de memoria) de la marca “ATMEL” que cuenta con entradas y salidas analógicas y digitales suficientes para cubrir las necesidades del proyecto a implementar.
- La programación se lo puede realizar mediante USB, de una forma rápida y sencilla, lo único que se debe hacer es simplemente conectar a un PC que contenga el software mediante el cable USB, se escribe el código del programa y luego se lo carga a la placa Arduino.
- Es sencillo y de bajo coste lo que permite el desarrollo de múltiples diseños, el voltaje que se necesita para poder trabajar está comprendido en un rango de 7 a 12 voltios DC, sin embargo se lo puede alimentar con una fuente de voltaje de 5 voltios DC estabilizada (por ejemplo con un LM7805).
- Al ser open-hardware tanto su diseño como su distribución es libre, por lo que se puede utilizar libremente para desarrollar cualquier tipo de proyecto sin la necesidad de adquirir ningún tipo de licencia.²¹

3.1.14 Ventajas de Arduino

Barato: Resultan relativamente baratas las placas de Arduino comparadas con otras plataformas microcontroladoras. La versión menos cara del módulo Arduino puede ser ensamblada a mano, y los módulos de Arduino pre ensamblados tienen un valor menor a 50 dólares.

²¹ *El Rincon de los Micorcontroladores*. (s.f.). consultado el 27 de Junio de 2014, de arduino Descripción: <https://sites.google.com/site/webmicrocontroladores/arduino/arduino-descripcion>

Multiplataforma: El software de Arduino se ejecuta en sistemas operativos Windows, Macintosh, OSX y GNU/Linux, hay que tomar en cuenta que la mayoría de los sistemas micro controladores están limitados a Windows.

Código abierto y software extensible: El software Arduino está publicado como herramientas de código abierto, se puede expandir el lenguaje mediante librerías C++, y si se desea entender los detalles técnicos se puede realizar el salto desde Arduino a la programación en lenguaje AVR-C en el cual está basado. De forma similar si se desea se puede añadir código AVR-C directamente en los programas Arduino.

Entorno de programación simple y clara: El entorno de programación de Arduino para principiantes resulta fácil, pero además es suficientemente flexible para que puedan ser aprovechados por usuarios avanzados en programación. Se puede familiarizarse con el entorno de programación Processing de tal manera que aprendiendo a programar en este entorno resulte sencillo acoplarse con el aspecto y la imagen de arduino.

Código abierto y hardware extensible: El Arduino está basado en microcontroladores ATMEGA8 y ATMEGA168 de Atmel. Los planos para los módulos están publicados bajo licencia Creative Commons, por lo que diseñadores experimentados de circuitos pueden hacer su propia versión del módulo, extendiéndolo y mejorándolo según sea la necesidad, inclusive usuarios ixepertos pueden construir la versión de la placa del módulo para entender su funcionamiento y lograr un ahorro de dinero.²²

3.1.15 Hardware Arduino

Existen en el mercado múltiples versiones de la placa Arduino, la mayoría usan el ATmega168 de Atmel, mientras que las placas más antiguas usan el ATmega8. Arduino por tener diferentes tipos hace un poco difícil la elección del más adecuado, pero la variedad también permite elegir la solución perfecta gracias a la flexibilidad que tiene.

²² *Guía de usuario de arduino.* (s.f.). consultado el 30 de Junio de 2014, de ventajas de arduino: http://www.jcarazo.com/tmp/Arduino_user_manual_es.pdf

A continuación se muestran algunas de las opciones importantes que tiene arduino.

3.1.15.1 Arduino Uno



Figura 7: Placa Arduino Uno

El Arduino Uno es una buena opción para empezar a usar Arduino y familiarizarse con el entorno, para los principiantes les facilita una base sólida y cuenta con muchas de las opciones que se desea a medida que se explora la plataforma. Otra de las ventajas es que trabaja con casi todos los shields disponibles. Más adelante se explicará que es un Shield en Arduino.

3.1.15.2 Arduino Nano



Figura 8: Placa Arduino Nano

El Arduino Nano es casi lo mismo que el Arduino Uno, se diferencia ya es aproximadamente 1/3 del tamaño y no se pueden usar con shields con tanta facilidad, además el Arduino Uno está destinado a ser utilizado como un elemento permanente en los proyectos para pruebas.

3.1.15.3 Arduino Lilypad

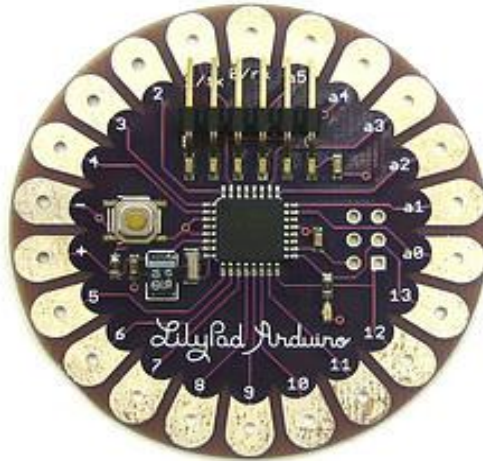


Figura 9: Placa Arduino Lilypad

El Lilypad tiene un diseño único, se puede coser en la tela para proyectos de vestir o arte, pero los shields no funcionan en este Arduino, de modo que la expansión puede ser difícil.

3.1.15.4 Arduino Mega 2560



Figura 10: Placa Arduino Mega 2560

El Mega 2560 posee más memoria y pines de E/S que cualquier otro Arduino. Este es el Arduino más grande y mejor que se puede conseguir, resulta también un poco más caro, por lo que antes de comprarlo se debe pensar si se desea para un proyecto permanente o para realizar montajes y pruebas con diferentes proyectos.

3.1.15.5 Netduino

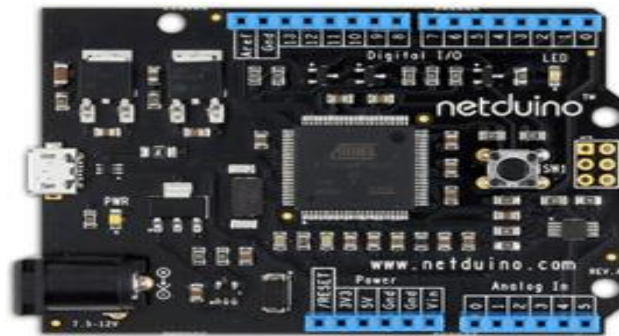


Figura 11: Placa Netduino

El Netduino es considerado como el primo del Arduino, continúa siendo una solución pirateada de hardware y de código abierto para prototipos. Sin embargo el Netduino ejecuta **.NET Micro Framework** para su software base y es compatible con los shields de Arduino, hay que tomar en cuenta que algunos requieren drivers para lograr su funcionamiento.

3.1.15.6 Arduino Fio



Figura 12: Placa Arduino Fio








El Arduino FIO está diseñado para aplicaciones inalámbricas. Se puede subir los sketches con un cable FTDI o una placa adicional adaptadora Sparkfun. Si se utiliza un adaptador de USB a XBee modificado también se puede subir los sketches o de forma inalámbrica. La tarjeta viene sin conectores pre-montados, permitiendo el uso de

diversos tipos de conectores o la soldadura directa de los cables. En el reverso de la placa tiene disponible un zócalo para módulos XBee.²³

3.1.16 Entorno Arduino

Arduino tiene un componente de software de código abierto que es similar a C++, el entorno de desarrollo integrado (IDE) de Arduino le permite escribir código, compilarlo, y luego subirlo a la placa para ser utilizado independiente en prototipos y proyectos. Esto fue diseñado con el propósito de que los inventores y usuarios puedan desarrollar libremente sus ideas en objetos reales.

3.1.16.1 Barra de Herramientas

-  Verificar/Compilar. Chequea el código, identificando los errores.
-  Parada, Detiene el monitor del puerto serial.
-  Crear nuevo proyecto.
-  Presenta un menú para abrir proyectos previamente guardados y también muestra los proyectos de ejemplos.
-  Guarda el proyecto actual dentro de la carpeta sketchbook/. Se puede guardarlo con un nombre distinto por el menú File → Save as.
-  Descarga el programa compilado desde el PC hasta la tarjeta Arduino.
-  Realiza el monitoreo del puerto serial, visualiza la data enviada desde la tarjeta Arduino.

3.1.16.2 Menús

Sketch

Verify/Compile: Comprueba la rutina para errores.

²³ *Hardware arduino.* (s.f.). consultado el 30 de Junio de 2014, de <http://domadis.com/2011/10/02/%C2%BFsabes-que-es-arduino-todo-lo-que-necesitas-saber-sobre-esta-plataforma-electronica-de-codigo-abierto/>

Import Library: Utiliza una librería en la rutina, trabaja añadiendo `#include` en la cima del código. Para no usar una librería se elimina el `#include` de la cima de la rutina.

Show Sketch Folder: Permite abrir la carpeta de rutinas en el escritorio.

Add File: Añade otro fichero fuente a la rutina. El nuevo archivo aparece en una nueva pestaña en la ventana de la rutina, lo que permite facilitar y agrandar proyectos con múltiples archivos fuente. Para eliminar los archivos de una rutina se usa el Tab Menu.

Auto Format: Formatea el código amigablemente.

Copy for Discourse: Copia el código de la rutina al portapapeles de forma conveniente para postear en un foro, completa con resaltado de sintaxis.

Board: Selecciona la placa que se utilizará, permitiendo controlar la forma en que la rutina es compilada y cargada, así como también el comportamiento de los elementos del menú Burn Bootloader.

Serial Port: Este menú contiene todos los dispositivos serie (real o virtual) de la máquina, se actualiza automáticamente cada vez que se abre el nivel superior del menú Tools.

Antes de subir la rutina, es necesario seleccionar el elemento de este menú que representa a la placa Arduino. En Windows aparece como COM1 o COM2 (para una placa Serie) o COM4, COM5, COM7 o superior (para una placa USB) para saber se debe buscar USB serial device en la sección puertos del Gestor de dispositivos de Windows.

Burn Bootloader: Los elementos en este menú permiten grabar un bootloader en la placa con una variedad de programadores. No es necesario para el uso normal de una placa Arduino, pero es útil si se desea incorporar ATmegs adicionales o se pretende construir una placa por cuenta propia.

Hay que asegurarse que se ha realizado la selección de la placa correcta del menú Boards. Para grabar un bootloader con el AVR ISP es necesario seleccionar el elemento que corresponde al programador del menú Serial Port.

Preferencias

Algunas de las preferencias se pueden ajustar en el diálogo Preferences (se encuentra bajo el menú Arduino en el Mac, o File en Windows y GNU/Linux), el resto se lo encuentra en los archivos de preferencias.

3.1.17 Comenzando con Arduino

Estructura

Es bastante simple la estructura básica que tiene el lenguaje de programación Arduino, se organiza en al menos dos partes o funciones que encierran bloques de declaraciones.

```
void setup()
{
  statements;
}
void loop()
{
  statements;
}
```

Son requeridas estas dos funciones para que el programa funcione.

setup()

Esta función debe contener la declaración de cualquier variable al comienzo del programa. Es la primera función que se ejecuta en el programa, es ejecutada una vez y se la utiliza para asignar pinMode o inicializar las comunicaciones serie.

```
void setup()
{
pinMode(pin, OUTPUT);    //ajusta 'pin' como salida
}
```

loop()

Este comando se ejecuta a continuación e incluye el código que se ejecuta continuamente para leer entradas, activar salidas, etc. Se constituye en el núcleo de todos los programas Arduino realizando la mayor parte del trabajo.

```
void loop()
{
digitalWrite(pin, HIGH);    //Activa 'pin'
delay(1000);                //espera un segundo
digitalWrite(pin, LOW);     //Desactiva 'pin'
delay(1000);                //espera un segundo
}
```

Funciones

Una función es un bloque de código que tiene un nombre y un grupo de declaraciones que se ejecutan cuando se llama a la función. Se puede hacer uso de funciones integradas como void setup() y void loop() o a la vez escribir nuevas funciones.

Las funciones se las escribe para ejecutar tareas repetitivas y tratar de reducir el desorden en un programa. Lo que primero se realiza es declarar el tipo de la función, que será el valor retornado por la función (int, void...). A continuación del tipo se debe declarar el nombre de la función y entre paréntesis cada uno de los parámetros que se pasan a la función.

```
type functionName(parameters)
{
```

```
statements;  
}
```

La siguiente función `int delayVal()`, asigna un valor de retardo en un programa por lectura.

```
int delayVal()  
{  
int v;           //crea una variable temporal 'v'  
v = analogRead(pot); //lee el valor del potenciómetro  
v /= 4;         //convierte 0-1023 a 0-255  
return v;       //devuelve el valor final de v  
}
```

Llaves {}

Definen el comienzo y el final de bloques de función y bloques de declaraciones como `void loop()` y sentencias `for` e `if`. Las llaves deben estar balanceadas (a una llave de apertura {debe seguirle una llave de cierre}), si no se encuentran balanceadas provocan errores de compilación.

```
void loop()  
{  
statements;  
}
```

Incluye el entorno Arduino una práctica característica para lograr chequear el balance de llaves, sólo se debe seleccionar una llave y su compañera lógica aparecerá resaltada.

Punto y coma;

Un punto y coma se usa al final de cada declaración, permite separar los elementos del programa y también separa los elementos en un bucle `for`.


```
int x = 13;           //declara la variable 'x' como el entero 13
```

Nota: Si se olvida de colocar un punto y coma al final de una declaración se producirá un error de compilación.

Bloques de comentarios /*...*/

Los bloques de comentarios, o denominados también comentarios multilínea, son áreas de texto que ignora el programa y se utilizan para realizar descripciones de código o comentarios que ayudan a otras personas a entender partes del programa. Empiezan con `/*` y terminan con `*/` y pueden abarcar múltiples líneas.

```
/*
```

Este es un bloque de comentario encerrado no olvides cerrar el comentario tienen que estar balanceados

```
*/
```

Como los comentarios son ignorados por el programa y no ocupan espacio en memoria deben usarse generosamente, también se utilizan para comentar bloques de código con propósitos de depuración.

Comentarios de línea //

Empiezan con `//` y terminan con la siguiente línea de código, es ignorado por el programa y no toman espacio en memoria.

```
// Este es un comentario de una línea
```

Los comentarios de una línea se usan por lo general después de declaraciones válidas proporcionando mayor información sobre qué lleva la declaración o proporcionar en el futuro un recordatorio²⁴

²⁴ *Guía de usuario de arduino.* (s.f.). Consultado el 30 de Junio de 2014, de Entorno de arduino: http://www.jcarazo.com/tmp/Arduino_user_manual_es.pdf

3.1.18 Consideraciones antes de adquirir una placa Arduino

Es necesario conocer las características para poder realizar una comparación entre las placas más comunes de Arduino: Arduino UNO, Arduino Leonardo, Arduino Mega 2560 y Arduino DUE.

La diferencia más importante entre los cuatro, es que todas las I/O del Arduino DUE trabajan a 3.3V, mientras que el resto de modelos lo hacen a 5V, sin embargo el Arduino DUE tiene una capacidad de memoria y sobre todo una velocidad de proceso muy superior a sus hermanos “pequeños”, indudablemente lo primero que se necesita es comparar las características principales de cada una de las placas.



Figura 13: Placa arduino Uno y Placa Arduino Leonardo



Figura 14: Placa Arduino Mega y Placa Arduino DUE

3.1.19 Características de modelos estándar de Arduino

Modelo	Arduino UNO	Arduino Leonardo	Arduino Mega 2560	Arduino DUE
Microcontroller	ATmega328	ATmega32u4	ATmega2560	AT91SAM3X8E
Operating Voltage	5V	5V	5V	3.3V
Input Voltage	7-12V	7-12V	7-12V	7-12V
Input Voltage (limits)	6-20V	6-20V	6-20V	6-20V
Digital I/O Pins	14	20	54	54
Digital I/O Pins PWM output	6	7	15	12
Analog Input Pins	6	12	16	12
Total DC Output Current on all I/O lines	40 mA	40 mA	40 mA	130 mA
DC Current for 3.3V Pin	50 mA	50 mA	50 mA	800 mA
DC Current for 5V Pin				800 mA
Flash Memory	32 KB 0.5 KB used by bootloader	32 KB 4 KB used by bootloader	256 KB 8 KB used by bootloader	512 KB available
SRAM	2 KB (ATmega328)	2.5 KB	8 KB	96 KB Two banks: 64KB y 32 KB
EEPROM	1 KB (ATmega328)	1 KB	4 KB	
Clock Speed	16 MHz	16 MHz	16 MHz	84 MHz
Tipo de USB	Estándar	Mini	Estándar	Mini
Precio Local US\$ (Sin imp.)	\$33.00	\$33.00	\$64.00	\$64.00

Tabla I: Características de los modelos mas estándar de arduinos

Memoria

Al realizar el análisis entre los cuatro modelos, lo primero que se encuentra es la gran diferencia de memoria disponible que hay entre ellos, mientras que el Arduino UNO y Arduino Leonardo tienen 32k, el Arduino Mega 2560 tiene 256k y el Arduino DUE 512k.

La memoria es considerada más importante que la velocidad, sobre todo cuando se tiene pensado utilizar librerías complejas en algún proyecto, por ejemplo una simple demostración que utilice las librerías para el manejo de pantallas UTFT ocupa más del 80% de la memoria disponible en el Arduino UNO y no es posible de compilar en el Arduino Leonardo (usaría el 104% de la memoria), mientras que en el Arduino Mega 2560 solamente se utiliza el 11% de la memoria, dejando cerca de un 90% de sus 256K para el código.

Velocidad

En la velocidad se encuentra una gran diferencia entre los tres primeros modelos y el Arduino DUE, este funciona con un reloj 5 veces más rápido que los anteriores y su procesador es mucho más potente, utiliza un ARM de 32-bit, pero la desventaja que presenta es la incompatibilidad en el código de las librerías que se suelen encontrar en la red..

Disponibilidad de pines digitales para PWM

Si un proyecto necesita manejar varios motores con control de velocidad y varios servos, se debe tener en cuenta el número de pines PWM que la placa tiene disponible para conocer la cantidad considerable que se va a utilizar de ellos, por ejemplo un brazo robot autosoportado con 6 servos y 2 motores, sin duda hay que pensar en un Arduino Mega 2560 con 16 pines PWM disponibles o un Arduino DUE con 12, tanto el Arduino UNO con 6 o el Arduino Leonardo con 7, se quedarán cortos si el proyecto es mínimamente complejo.

Disponibilidad de pines digitales estándar

Hay que tener en cuenta el número de pines de I/O Digitales estándar disponibles, estos se los utiliza dentro de proyectos para conectarse con sensores u otro tipo de periféricos, ya que cada uno necesita uno o varios pines disponibles, por ejemplo si en el proyecto se desea utilizar una placa de 8 relés, un teclado numérico y un display 2x16 estándar: los relés necesitan 8, el teclado 5 y el display unos 6, es decir se necesita más de 18 pines disponibles, por lo tanto el Arduino UNO y el Arduino Leonardo se quedarán cortos ante esta situación, volviendo a ser la elección el Mega 2560 con 54 pines disponibles, el Arduino DUE tiene la misma cantidad disponible, pero hay que tener en cuenta que el voltaje de funcionamiento en el I/O es de 3.3V y se necesita convertidores de voltaje para conectarlo a la mayoría de periféricos que son de uso común en el mercado.

Disponibilidad de pines Analógicos

Al momento de entrar en contacto con el mundo de los sensores, la mayoría necesita conectarse a pines analógicos de entrada, en este caso también gana el Arduino Mega 2560 y que dispone de 16 pines, le sigue el Arduino Leonardo con 20 y en última posición el Arduino UNO con 6. Si el proyecto necesita utilizar salidas analógicas, por ejemplo para reproducir sonido de calidad, el Arduino DUE es el único que ofrece esa posibilidad con 2 salidas analógicas (DAC).

Flexibilidad en la reparación

Cuando ya se tiene experiencia es mínima la posibilidad de que la placa produzca “humo mágico” es decir se “fría” por haberlo conectado mal. Arduino Uno es el único modelo que es reparable fácilmente cambiando el chip del microcontrolador (viene en zócalo), en el resto los microcontroladores son componentes de superficie y se encuentran soldados directamente a la placa por lo que la probabilidad de repararlos en caso de avería es prácticamente nulas.

3.1.20 Análisis para la adquisición de la plataforma Arduino

Complejidad en el manejo

Ahora que ya se conoce un poco más las características de los cuatro modelos, resulta un poco más fácil realizar la elección de la placa a utilizar, en el caso de ser principiantes se aconseja el Arduino UNO ya que es posible de reparar, todas los shields del mercado son compatibles con él y las librerías de los ejemplos funcionan sin problemas, de esta forma se puede ahorrar horas de búsqueda y modificaciones en librerías hasta hacer funcionar dispositivos complejos.

Disponibilidad de pines

Si se conoce que el proyecto que se va a implementar necesita muchos sensores y actuadores, es lógico que se va a utilizar muchos pines de I/O disponibles pero para el código no se necesitará mucha memoria, entonces se puede elegir el Arduino Leonardo, pero se debe tener en cuenta que hay numerosos shields que no son compatibles con él, para lo cual se debe estudiar cuidadosamente sus características antes de comprar alguna.

Si se elige el Arduino UNO o el Arduino Leonardo se debe de tener en cuenta, que aunque se tenga pines de I/O libres, si se utiliza Shields colocadas encima de ellos aunque se tenga pines de I/O libres, físicamente estarán tapados por la Shield conectada y no se tendrá posibilidades de usarlos, excepto si se hace algunos puentes medio chapuceros con cables colgando.

Selección de Arduino Mega 2560

Si se tiene experiencia y se ha previsto realizar un proyecto complejo o simplemente se desea realizar una buena inversión pensando en que “es mejor que sobre a que falte y que tarde o temprano se lo va a necesitar”, se recomienda comprar un Arduino Mega 2560, ya que es compatible casi al 100% con el Arduino UNO a nivel de librerías y shields.

Cuando se coloca encima una Shield para el Arduino UNO, deja acceso a una buena cantidad de pines para conectar los sensores y actuadores, no se debe olvidar que se

tiene 54 I/O digitales y 16 entradas analógicas, pero no solo eso, además se tiene 4 UARTs, es decir cuatro puertos serie por hardware, lo que permiten conectarse con dispositivos serial o incluso con otros Arduinos.

El conector adicional (Pines 22 a 53) es muy práctico para utilizar, sobre todo con proyectos que necesitan conexiones múltiples por dispositivo, por ejemplo motores paso a paso, que cada uno necesita 4 pines, el disponer de más interrupciones (External Interrupts pines: 2 (interrupt 0), 3 (interrupt 1), 18 (interrupt 5), 19 (interrupt 4), 20 (interrupt 3), y 21 (interrupt 2) ayuda mucho con los proyectos complejos que necesitan tener los valores de ciertos sensores independientemente del funcionamiento del resto de código, por ejemplo los sensores de detección de choque, interruptores de fin de carrera, tacómetros, compases magnéticos, etc.

Selección de Arduino DUE

El Arduino DUE se lo debe elegir cuando se necesita implementar proyectos muy complejos que necesiten necesidades de cálculo y memoria. Para sistemas de control especializados tiene la ventaja de que es posible asignar una External interrupt a cualquier pin de I/O, esto es imprescindible cuando se necesita trabajar con aplicaciones en tiempo real.

Selección de Arduino ADK REv3

Hay que tener en cuenta que el Arduino ADK Rev3 es imprescindible para los fanáticos de Android, básicamente es un Arduino Mega 2560 al que se le ha añadido un puerto USB adicional para conectarse directamente a los dispositivos Android. Es compatible con los ejemplos contenidos en el Android Accessory Development Kit. Cuenta con 54 I/O digitales (14 se pueden utilizar como salidas PWM), 16 entradas analógicas, 4 UARTs (puertos serie de hardware), un oscilador de cristal de 16 MHz, una conexión USB, un conector de alimentación, un conector ICSP y un botón de reset.²⁵

²⁵ *Consideraciones antes de comprar una placa arduino.* (s.f.). Consultado el 02 de Julio de 2014, de <http://openhardware.pe/que-modelo-de-arduino-debo-comprar-para-mis-proyectos/>

3.1.21 Descripción del Arduino a Utilizar en el Proyecto

Previamente basándonos en el análisis anterior, en nuestro proyecto se utilizará la placa Arduino Mega 2560 porque se adapta mejor a nuestros requerimientos.

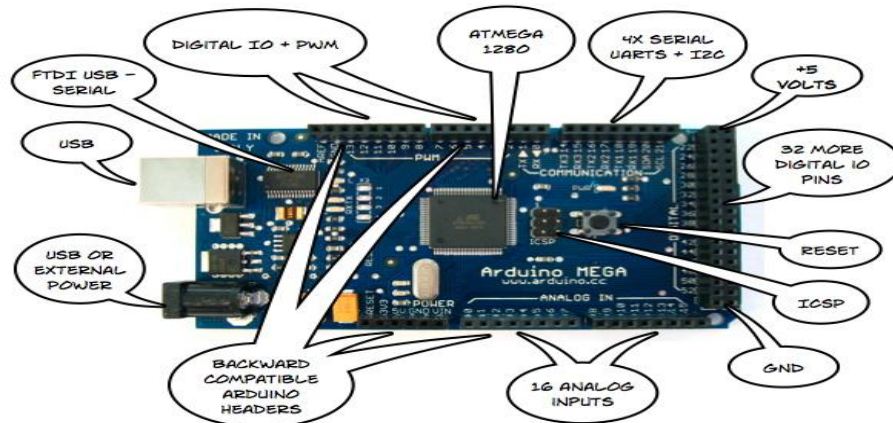


Figura 15: Descripción Arduino Mega 2560

La placa Arduino Mega 2560 posee:

- 54 entrada/salida digitales, de los cuales 14 pueden ser usados como salidas PWM
- Posee 16 entradas analógicas
- Posee 4 puertos seriales por Hardware (UART)
- Un oscilador de cristal de 16 MHz
- Conector USB
- Un jack de poder
- Una conector ICSP
- Botón de Reset

Algunas características son:

Microcontroller	ATmega2560
Operating Voltage	5V
Input Voltage (recommended)	7-12 V
Input Voltage (limits)	6-20 V
Digital I/O Pins	54 (of which 14 provide PWM output)
Analog Input Pins	16
DC Current per I/O Pin	40 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	256 KB (of which 8 KB used by bootloader)
SRAM	8 KB
EEPROM	4 KB
Clock Speed	16 MHz

Tabla II: Características de Arduino Mega

26

Accesorios de Arduino (Shields)

Un Shield es una placa impresa que se puede conectar en la parte superior de la placa Arduino para ampliar sus capacidades. La placa Arduino adquiere mayor funcionalidad a través del shield, poseen pines que se colocan justo encima del Arduino logrando aprovechar inmediatamente lo que sea que el Shield pueda hacer. Se puede agregar múltiples shields al mismo tiempo.

Por ejemplo, se puede usar un Arduino para que busque en Twitter por un hashtag específico (Shield de Ethernet) y luego controle un carrito a control remoto con radiofrecuencia (Shield RF), gracias a la flexibilidad de los shields se pueden alcanzar realizar una gran variedad de proyectos.

²⁶ *Características Arduino.* (s.f.). Consultado el 29 de Junio de 2014, de <http://www.arduino.cl/int/caracteristicas.html>

3.1.22 Tipos de Shield

Algunos shields tienen similitud a los Arduino pero no se debe confundir, ya que los shields no poseen la capacidad de procesamiento principal para ejecutar el código que se escribe en los sketches o bocetos (más sobre esto más adelante).

3.1.22.1 Shield de Ethernet

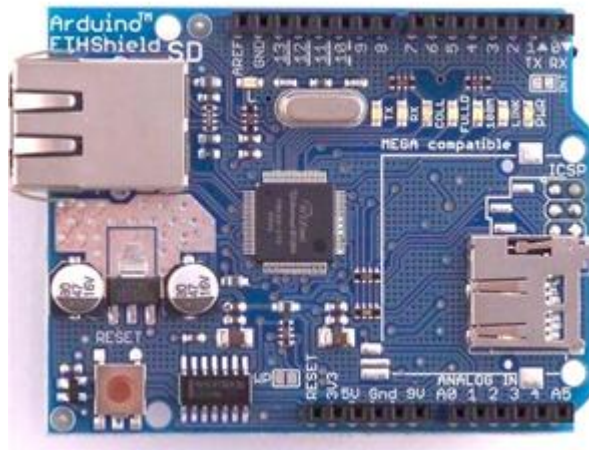


Figura 16: Shield de Ethernet

Es uno de los shields más populares, permite al Arduino poder usar el internet para fines de comunicación y control. El Shield de Ethernet es uno de los más versátiles del mercado, es imprescindible y se puede lograr realizar cualquier proyecto con este shield.

3.1.22.2 XBee



Figura 17: Shield Xbee

El Shield XBee permite que la comunicación inalámbrica de punto a punto resulte sencilla. Se puede utilizar este Shield para conectar en red dos Arduinos o para crear una malla de red completa de Arduinos para cualquier proyecto.

3.1.22.3 Motor



Figura 18: Shield de Motor

Arduino tiene la capacidad para controlar motores y servos sin necesidad de un shield, pero el shield motor permite elevar esta capacidad. Se puede utilizar este shield para lograr diseñar un propio robot y controlarlo a distancia, existe una gran cantidad de shields para Arduino disponibles, hay shields para música, videojuegos, bluetooth, pantallas táctiles, LCD. Se puede decir que hay shields para casi todo.²⁷

3.1.23 Descripción del ECom GPRS/GSM Shield a utilizar en el proyecto

GPRS Shield ECom es un ultra módulo inalámbrico compacto y fiable. Este GPRS Shield es compatible con todas las placas que tienen el mismo factor de forma como una placa Arduino estándar. ECom es la base en SIM900 módulo GPRS Frecuencia, que ofrece GSM / GPRS 850/900/1800 / 1900MHz para voz, SMS, datos y fax en un pequeño factor de forma y con bajo consumo de energía. SIM900 es un módulo de cuatro bandas GSM / GPRS completa en un tipo de SMT y diseñado con un potente

²⁷ *Accesorios de Arduinos Shields.* (s.f.). Consultado el 04 de Julio de 2014, de <http://domadis.com/2011/10/02/%C2%BFsabes-que-es-arduino-todo-lo-que-necesitas-saber-sobre-esta-plataforma-electronica-de-codigo-abierto/>

procesador de un solo chip de integración AMR926EJ-S núcleo, lo que le permite beneficiarse de pequeñas dimensiones y de soluciones rentables.

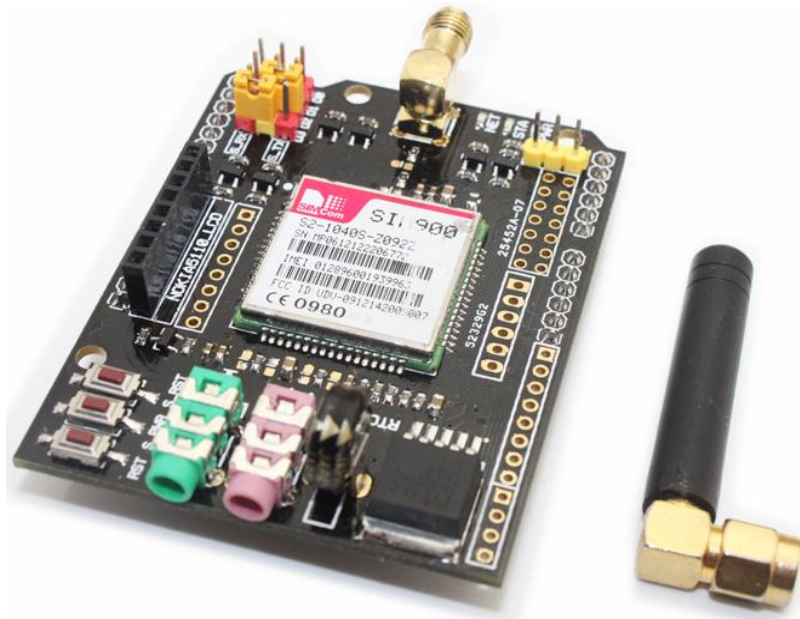


Figura 19: ECom GPRS/GSM Shield

El Shield GPRS se configura y se controla a través de su UART usando simples comandos AT. Sólo se tiene que conectar este Shield en la placa Arduino y se puede fácilmente usar comandos AT para el control del ECom Shield. Se puede utilizar el bloque 2 de puente para conectar el mensaje URAT SIM900 a los pines dentro de D0-D3 (para Hardware / Software puerto serie). Hay un interruptor en el tablero, se puede utilizar para seleccionar la conexión del puerto UART o puerto de depuración, incluso activarse en Arduino, pero por el bloque de conmutadores y puentes, el SIM900 se puede conectar a la PC a través de FT232RL.

El shield permite lograr esto a través de cualquiera de los tres métodos:

- Servicio de mensajes cortos
- Audio
- GPRS

Características

- Totalmente compatible con Arduino / Freaduino y Mega.
- Puerto serie libre de conexión, puede seleccionar el control de hardware de puerto serie (D0 / D1) o software de puerto serie (D2 / D3) controlarlo.
- Súper condensador de fuente de alimentación para el RTC.

- EFCOM no sólo se puede utilizar el botón de encendido, pero también puede utilizar el pin digital de Arduino para encender y reiniciar el módulo SIM900.
- Quad-Band 850/900/1800/1900 MHz
- GPRS clase 10/8 multi-slot
- GPRS clase de estación móvil B
- Control a través de comandos AT (GSM 07.07, 07.05 y EFCOM mejorado Comandos AT)
- SIM Application Toolkit
- Suministro rango de tensión: 3,1 ... 4,8 V
- Bajo consumo de energía: (modo de ahorro) 1.5 mA
- Temperatura de funcionamiento: -40 ° C a +85 ° C
- Dimensión: 68.33x53.09mm (Igual dimensión de la placa principal de Arduino)

Precauciones

- Se debe asegurarse al usar el cargador que provea de 9V para la alimentación de la placa Arduino y EFCOM, La fuente de alimentación debe ser capaz de proporcionar suficiente corriente hasta 2A. El puerto USB no puede suministrar una corriente tan grande.



Figura 20: Cargador de EFCOM GPRS/GSM Shield

- Se debe estar seguro de que la tarjeta SIM está desbloqueado.
- El producto se proporciona tal cual y sin un recinto aislante. Hay que tener en cuenta las precauciones de ESD especialmente en (baja humedad) seco clima.
- El ajuste predeterminado de fábrica para el GPRS Shield UART es 19200 bps 8-N-1. (Se puede cambiar utilizando los comandos AT).

Definición de pines y clasificación

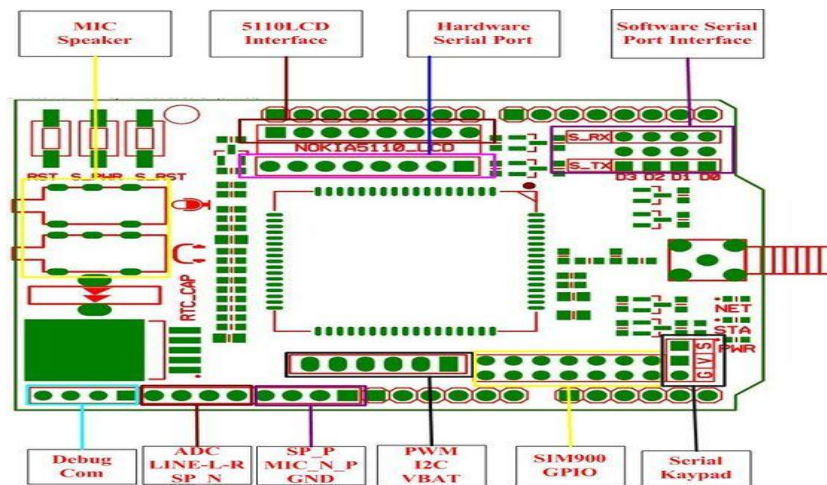


Figura 21: Top - view

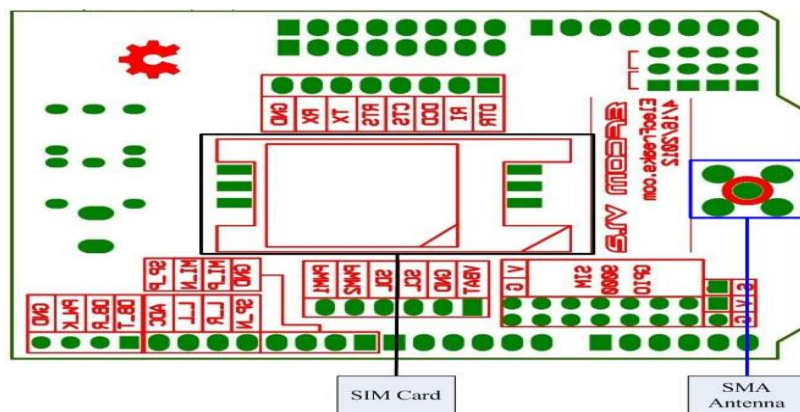


Figura 22: Bottom view

Dimensiones Mecánicas

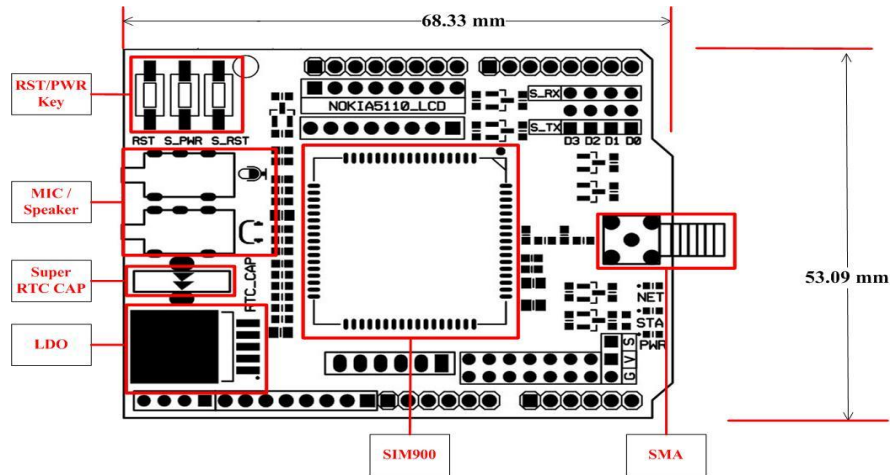


Figura 23: Dimensiones mecánicas de ECom GPRS/GSM Shield

Luz de Estado

LED	Estado	Función
Estado	Off	Apagado
	On	Encendido
NetLight	Off	SIM900 no está funcionando
	64ms On / Off 800ms	SIM900 no encuentra la red
	64ms On / Off 3000ms	SIM900 encontrar la red
	64ms On / Off 300ms	Comunicación GPRS

Tabla III: Luz de estado de ECom GPRS/GSM Shield

Shield GSM y Arduino

El GPRS Shield viene con todos los accesorios que se necesita para empezar con el envío de datos a través de la red GSM a excepción de una placa Arduino y una tarjeta SIM GSM con una suscripción activa en el plan de datos. Si desea realizar llamadas de voz, también requeriría unos auriculares con micrófono.

3.1.24 Creación de una configuración de prueba para Shield GPRS

Al recibir el Shield GPRS lo que sería la primera actividad en realizar con ella es enviar un texto (SMS) a un teléfono celular o llamar a alguien. Se puede hacer todo esto al

comunicarse con el Shield GPRS utilizando comandos AT que es un lenguaje especial que entiende. Los comandos AT son comandos de texto simples enviados al módem GPRS en su interfaz serial (UART), por lo que puede hacer uso de cualquier software de terminal serie para comunicarse con él, tal es el caso de HyperTerminal, Serial Monitor de Arduino IDE o Serial herramienta .

1. Instalación de la tarjeta SIM en su Shield GPRS. Sólo se necesita un plan activo, datos si desea utilizar GPRS. Si usted está buscando solamente para enviar mensajes de texto (mensajes SMS) o realizar llamadas de voz, entonces no se requiere.
2. Se debe conectar la antena al Shield GPRS.
3. Se instala el shield GPRS sobre la placa arduino.
4. Hay que asegurarse de que los puentes del GPRS_TX y GPRS_RX sobre el Shield GPRS están montados en posición SWSerial es decir que queremos GPRS_TX para conectarse a D2 (RX) y GPRS_RX a D3 (TX).
5. Conectar el Arduino al ordenador mediante un cable USB.
6. Se necesita un Sketch Arduino que se ejecuta dentro de la ATmega328P que emula un segundo puerto serie (UART) utilizando software en los pines digitales D2 y D3 y parches a través de toda la comunicación entre este segundo puerto de serie del software y el puerto serie hardware real. De esta manera, todos los datos que llegan desde el ordenador (conectado a la UART hardware real) se transmiten como es el Shield GPRS (conectado a UART software) y que sería capaz de emitir comandos AT para controlar el Shield GPRS.²⁸²⁹
7. El entorno de código abierto Arduino hace fácil la escritura del código y cargarlo a la placa de entrada y salida que permitirá realizar determinadas funciones. El software Arduino puede ser utilizado en Windows, Mac, Linux y Androide.³⁰

²⁸ *EFCOM GPRS/GSM Shield*. (s.f.). Consultado el 05 de Julio de 2014, de http://www.electronics.com/wiki/index.php?title=EFCOM_GPRS/GSM_Shield

²⁹ *GSM Shield SIM 900*. (s.f.). Consultado el 04 de Julio de 2014, de http://www.fut-electronics.com/wp-content/plugins/fe_downloads/Uploads/GSM-shield-datasheet-Arduino-tutorial.pdf

³⁰ *Arduino*. (s.f.). Consultado el 12 de Junio de 2014, de <http://arduino.cc/en/pmwiki.php?n=main/software>

3.1.25 Opciones de la Interfaz

- ▶ Verificar/Compilar. Chequea el código, identificando los errores.
- ◻ Parada, Detiene el monitor del puerto serial.
- 📁 Crear nuevo proyecto.
- 📂 Presenta un menú para abrir proyectos previamente guardados y también muestra los proyectos de ejemplos.
- 💾 Guarda el proyecto actual dentro de la carpeta sketchbook/. Se puede guardarlo con un nombre distinto por el menú File → Save as.
- 📡 Descarga el programa compilado desde el PC hasta la tarjeta Arduino.
- 🖨 Realiza el monitoreo del puerto serial, visualiza los datos enviados desde la tarjeta Arduino.

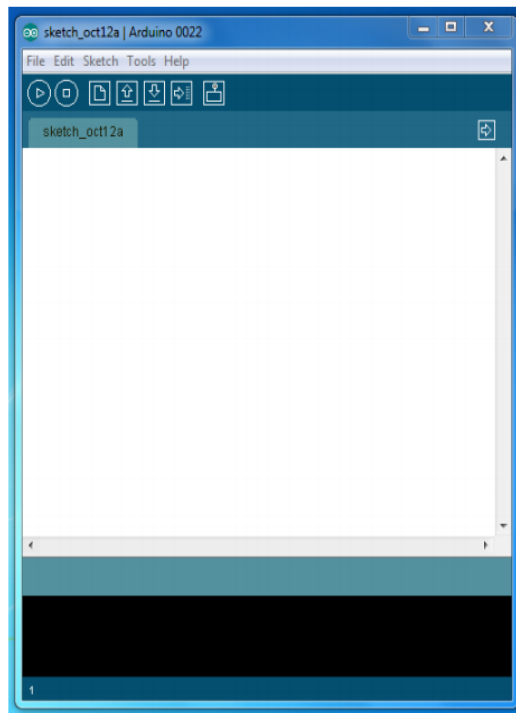


Figura 24: Interfaz de Arduino

3.1.26 Conociendo el software Arduino

Para ejecutar el programa Arduino, se ingresa a la carpeta de Arduino y se debe dar doble click en el icono de Arduino.

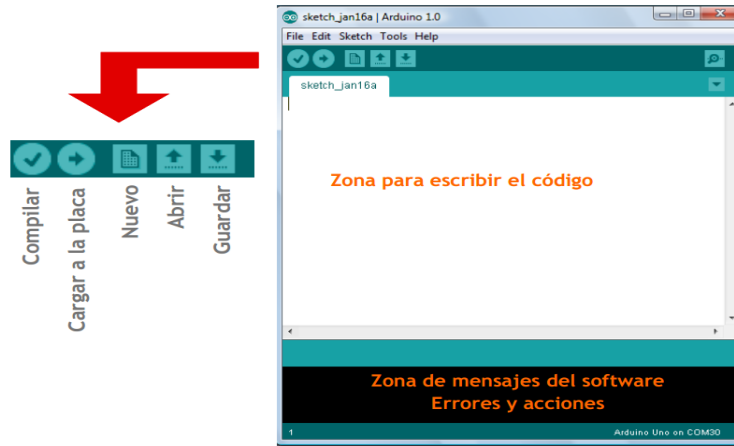


Figura 25: Ejecución de Arduino

1. Puerto COM

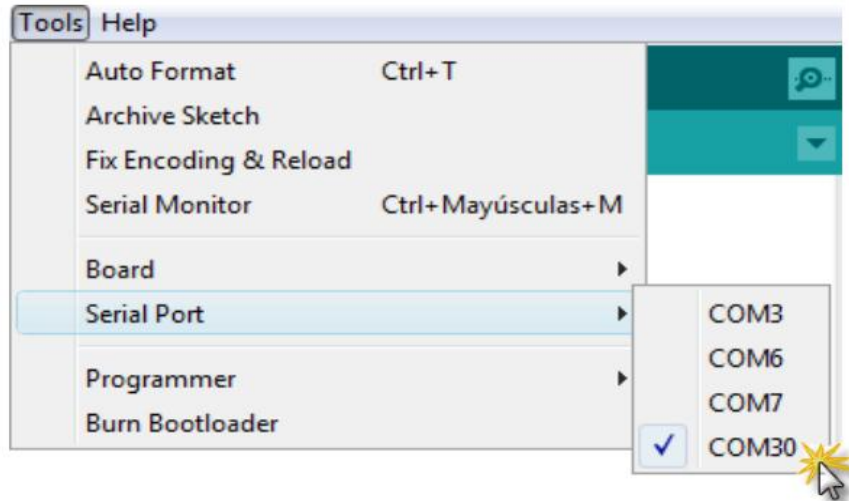


Figura 26: Selección de Puerto COM en arduino

2. Seleccionar la placa a trabajar

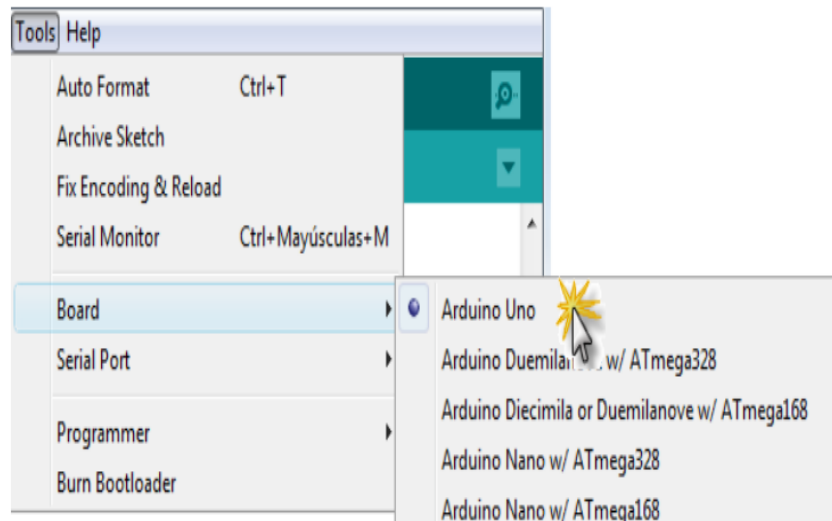


Figura 27: Selección de placa a trabajar en arduino

3. Consola serial ³¹

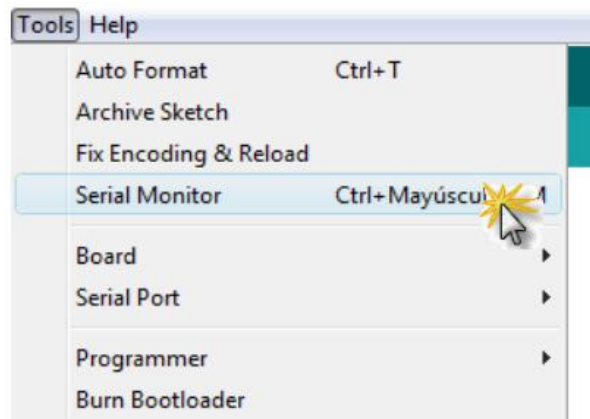


Figura 28: Consola serial en Arduino

³¹ *Guia Práctica Sobre el Mundo del Arduino*. (s.f.). Consultado el 14 de Julio de 2014, de http://www.tiendaderobotica.com/download/Libro_kit_Basico.pdf

3.1.27 Comunicación de Arduino con Puerto Serie

Los puertos serie permiten comunicar una placa Arduino con un ordenador. Gracias al puerto de serie se puede, por ejemplo, mover el ratón o simular la escritura de un usuario en el teclado, controlar un robot realizando los cálculos en el ordenador, enviar correos con alertas, encender o apagar un dispositivo desde una página Web a través de Internet, o desde una aplicación móvil a través de Bluetooth.

Existen un sin fin de posibilidades en las que se requiere el empleo del puerto serie, convirtiéndose en un componente fundamental de una gran cantidad de proyectos de Arduino, y es uno de los elementos básicos para poder sacar todo el potencial que tiene Arduino.

3.1.27.1 Puerto Serie

Un puerto es el nombre genérico que se asigna a los interfaces, físicos o virtuales, permitiendo la comunicación entre dos ordenadores o dispositivos. Se envía la información mediante una secuencia de bits, por lo que se necesitan al menos dos conectores para realizar la comunicación de datos, RX (recepción) y TX (transmisión). Sin embargo pueden existir otros conductores para referencia de tensión, sincronismo de reloj, etc.

Por el contrario, un puerto paralelo envía la información mediante múltiples canales de forma simultánea. Para lograr este proceso necesita un número superior de conductores de comunicación, que varían en función del tipo de puerto. De la misma forma existe la posibilidad de conductores adicionales además de los de comunicación.

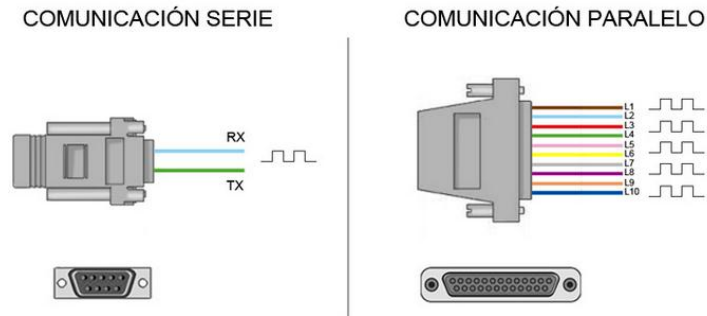


Figura 29: Comunicación Serie y Paralelo

Históricamente ambos tipos de puertos han convivido en los ordenadores, se ha empleado los puertos paralelos en aplicaciones que han requerido la transmisión de mayores volúmenes de datos. Sin embargo, a medida que los procesadores se hicieron más rápidos los puertos de serie fueron desplazando progresivamente a los puertos paralelos en la mayoría de aplicaciones.

Un ordenador convencional dispone de varios puertos de serie, entre los más conocidos se tiene: USB (universal serial port) y el ya casi olvidado RS-232 (el de los antiguos ratones). Dentro del ámbito de la informática existen una gran cantidad adicional de tipos de puertos serie, como por ejemplo el RS-485, I2C, SPI, Serial Ata, Pcie Express, Ethernet o FireWire, entre otros.

En ocasiones se refiere a los puertos de serie como UART. La UART (universally asynchronous receiver/transmitter) es una unidad que incorporan ciertos procesadores, encargada de realizar la conversión de los datos a una secuencia de bits y transmitirlos o recibirlos a una determinada velocidad.

Por otro lado, también se hace referencia a TTL (transistor-transistor logic). Esto significa que la comunicación se realiza mediante variaciones en la señal entre 0V y Vcc (donde Vcc suele ser 3.3V o 5V). Por otro lado otros sistemas de transmisión emplean variaciones de voltaje de -Vcc a +Vcc (por ejemplo, los puertos RS-232 típicamente varían entre -13V a 13V).

3.1.28 Arduino y el Puerto Serie

Usualmente todas las placas Arduino disponen al menos de una unidad UART. Las placas Arduino UNO y Mini Pro disponen de una unidad UART que operan a nivel TTL 0V / 5V, por lo que son directamente compatibles con la conexión USB. Arduino Mega y Arduino Due disponen de 4 unidades UART TTL 0V / 5V.

Están físicamente los puertos serie unidos a distintos pines de la placa Arduino. Evidentemente, mientras se usa los puertos de serie no se puede usar como entradas o salidas digitales los pines asociados con el puerto de serie en uso.

Los pines empleados en Arduino UNO y Mini Pro son 0 (RX) y 1 (TX). En el caso de Arduino Mega y Arduino Due el puerto de serie 1 está conectado a los pines 0 (RX) y 1 (TX), el puerto de serie 2 a los pines 19 (RX) y 18 (TX) el puerto de serie 3 a los pines 17 (RX) y 16 (TX), quedando el puerto serie 4 para conectarse a los pines 15 (RX) y 14 (TX).

La mayoría de placas Arduino disponen de un conector USB o Micro USB que se encuentra conectado a uno de los puertos de serie, lo que facilita el proceso de conexión con un ordenador. Sin embargo algunas placas, como por ejemplo la Mini Pro, prescinden de este conector por lo que la única forma de conectarse a las mismas es directamente a través de los pines correspondientes.

3.1.29 Conexión de Arduino con un Ordenador

Para realizar la conexión mediante puerto serie se necesita únicamente conectar la placa Arduino utilizando la misma interface que se emplea para programarlo. A continuación se debe abrir el IDE Standard de Arduino y hacer click en el "Monitor Serial" como se indica en la imagen.

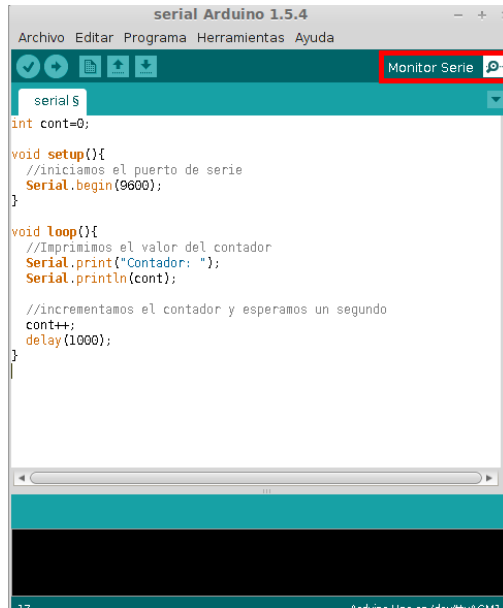


Figura 30: Monitor serial Arduino

El monitor de puerto serie es una utilidad pequeña que se encuentra integrada dentro de IDE Standard lo que permite enviar y recibir fácilmente información a través del puerto serie. Es muy sencillo su uso y se dispone de dos zonas, una que muestra los datos recibidos, y otra para enviarlos. En la siguiente imagen se muestran dichas zonas

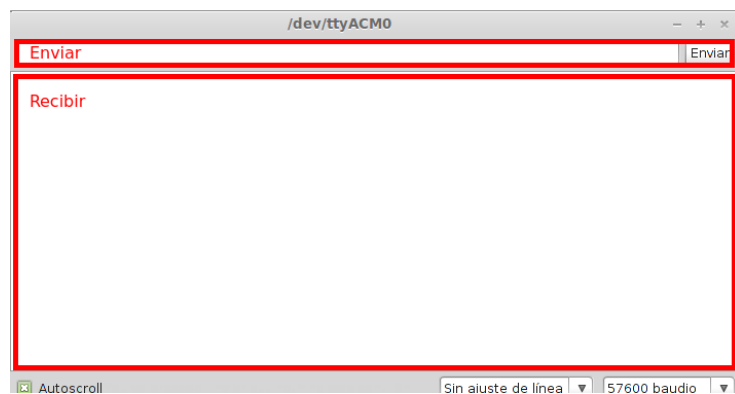


Figura 31: Zonas del monitor del puerto serie de Arduino

Este monitor de puerto serie a pesar de su sencillez es suficiente, resultando muy útil para realizar test o experimentos rápidos.³²

Tecnología Omron

Omron es una empresa japonesa que se dedica al desarrollo y producción de equipos y monitores para mejorar la calidad de vida humana. Las últimas tecnologías que surgen dentro del Centro de Desarrollo Tecnológico de Japón pasan a ser concretadas a los productos de última generación. Omron es el líder mundial en tensiómetros de uso hogareño fabricando 9.000.000 de unidades por año. La alta calidad de sus productos y servicios están avalados institucionalmente por la Federación Argentina de Cardiología (FAC). Su misión es ofrecer productos y servicios tanto al público general como al médico profesional para que sirvan realmente mejorar la calidad de vida.

No se puede sustituir la opinión médica con el uso de monitores de presión, aun cuando al usar estos el resultado de las lecturas está dentro de los rangos normales, no se debe realizar el cambio en las indicaciones que él médico ha dispuesto en cuanto a medicamentos o dietas, es de suma importancia que se pueda llevar un resumen de los resultados de las lecturas para luego compartirlas con el doctor.

3.1.30 Historia del Medidor de Presión Arterial Omron

La innovación del monitor de presión arterial Omron empezó con el concepto de "Ingeniería en salud" a inicios de los años sesenta. Ingeniería en salud es una idea de la Corporación Omron, Kazuma Tateisi quien se inspiró por el ultimo sistema de automatización de fábrica por aquellos tiempos, teniendo un concepto de comparar los sistemas de comunicación y control del cuerpo humano con máquinas que existían y lograr manejar la salud de una persona, buscando diagnosticar enfermedades al utilizar una combinación de automatización y tecnologías de informática.

La investigación basada en esta lógica única comenzó en la central del laboratorio R&D en 1961. Desde entonces, Omron ha estado trabajando constantemente en el desarrollo de un monitor de presión arterial para uso doméstico con la política de

³² *Comunicaion de Arduino con Puerto Serie.* (s.f.). Consultado el 14 de Agosto de 2014, de <http://www.luisllamas.es/2014/04/arduino-puerto-serie/>

contribuir al cuidado de la salud mediante tecnologías que permitan diagnóstico con una política corporativa presente de “trabajar para el beneficio de la sociedad”.

El primer monitor digital de presión arterial Omron, “monitor de presión arterial manual del tipo manómetro (HEM-1)”, fue lanzado en 1973, y el “monitor de presión arterial digital para uso doméstico (HEM-77)”, fue desarrollado en 1978, año en el que se estableció la Sociedad de Hipertensión de Japón, “monitor de presión arterial con método oscilo métrico (HEM-400C/HEM-700C)” que es adaptado al innovador método oscilo métrico en 1985, “monitor de presión arterial digital auto con lógica difusa (HEM-706)” que es el primer modelo basado en lógica difusa del mundo en 1991, “y un modelo con sistema de brazalete totalmente automático (HEM-1000)” adaptado al concepto de diseño universal en 2004.

A través del tiempo los monitores de presión arterial se han constituido altamente confiables no solo en Japón sino también en todo el mundo como resultado de la búsqueda de la “exactitud” y “facilidad de uso”.

3.1.31 Listado Cronológico de Monitores de Presión Arterial

- 1961 La idea se estableció en el laboratorio central R&D de Electrónica Tateisi que es anterior a la Corporación Omron. Esta idea ingresó oficialmente al negocio de la ingeniería para el cuidado de la salud.
- 1964 Empezó el desarrollo de un medidor de presión arterial simple en el laboratorio central R&D.
- 1970 Es el paso de la “Ingeniería para el cuidado de la salud” al primer monitor de presión arterial.
- 1973 Aparece el primer Monitor de presión arterial Omron manual de tipo manómetro.
- 1974 Establecimiento del laboratorio de ciencia Tateisi (cuyo nombre fu cambiado a laboratorio de ciencia Omron en 1990).
- 1978 Primer monitor de presión arterial digital Omron para uso doméstico.
- 1981 Primer monitor de presión arterial automático digital Omron para uso doméstico.
- 1982 Primer ingreso al mercado exterior (Alemania).

- 1984 Monitor de presión arterial con impresora, capaz de medir los valores de presión arterial.
- 1986 Monitor de presión arterial con método oscilo métrico.
- 1988 Monitor de presión arterial para colocar en el dedo. Aquí recibe el premio de producto para el bienestar del buen diseño.
- 1990 Primera innovación mundial de monitor de presión arterial basada en lógica difusa.
- 1991 primer monitor de presión arterial digital auto con lógica difusa.
- 1992 Monitor de presión arterial con deflación de velocidad constante. Alcanzó las metas de desarrollo “amigable, rápido y silencioso”. En este año aparece el primer monitor de presión arterial de muñequera del mundo.
- 1999 monitor de presión arterial con la tecnología de medición más rápida en el mundo. Disminuyó en tiempo de medición a 21 – 22 segundos que anteriormente llevaba un tiempo de 4 segundos, a menor tiempo de medición menor dolor durante la inflación.
- 2000 Monitor de presión arterial de pulsera más pequeño del mundo alcanzando 100 millones de unidades vendidas.
- 2001 Monitor de presión arterial con nuevo pre-formado para un rápido ajuste.
- 2002 Monitor de presión arterial de muñequera con sensor de posición avanzada.
- 2003 Las ventas de los monitores de presión arterial Omron alcanzan los 50 millones de unidades.
- 2004 primer modelo en el mundo con sistema completamente automático de ajuste del brazalete.
- 2006 Primer monitor de presión arterial en la Industria con indicador de Hipertensión Matutina.
- 2008 Modelo reducido en tamaño y peso, aparece un monitor de presión arterial a energía solar.
- 2009 Las ventas de monitores de presión arterial para uso doméstico superaron los 100 millones de unidades en todo el mundo.³³

³³ *Historia del Medidor de Presión Arterial Omrom.* (s.f.). consultado el 14 de Julio de 2014, de http://omronecuador.com/100millones/historia/02_es.htm

3.1.32 Características

- Precisión, siendo esto lo más primordial
- Comodidad en el uso, se debe tomar en cuanto el tamaño del brazo vs el tamaño de la manga medidora del monitor, ya que si se utiliza el tamaño equivocado puede resultar en medidas erradas.
- Facilidad de lectura e interpretación de resultados.
- Capacidad de almacenamiento.
- Capacidad para más de un usuario.

Es importante resaltar que las medidas de la manga medidora no necesariamente son del tamaño del brazo de cada persona, una manga de 17" puede medir un brazo de mayor tamaño.

Los monitores de presión arterial omron tienen varias características de tecnologías propietarias de estos tales como:

- Comfit Cuff TM es una característica patentada de Omron donde el mango de medidor viene con la forma del brazo y está diseñado para medir tamaños de brazos regulares y grandes garantizando una medición correcta y confortable.
- Trueread TM puede realizar 3 tomas consecutivas con intervalos de un minuto entre estas y mostrando el promedio de las mismas, para esto se debe seguir las recomendaciones internacionales para la correcta medición de la presión en el hogar.
- Tecnología de Inflado Intelisense® es una tecnología exclusiva de Omron que permite a sus medidores inflarse y desinflarse al nivel adecuado de cada individuo de modo que presente lecturas más correctas.

3.1.33 Especificaciones del medidor de presión arterial Omron

El medidor de presión arterial HEM-7113 que se va a utilizar es de fácil uso, no solo puede realizar lecturas de la presión sistólica, presión diastólica y del pulso, además tiene la capacidad de alertar al usuario informando cuando el monitor detecta

hipertensión o un latido irregular del corazón. Gracias a su operación simple, silenciosa, con tan sólo tocar un botón, se obtendrá una medición precisa y confiable de la presión arterial, en cuestión de segundos.

Cuenta con las siguientes especificaciones:

30 memorias: El HEM-7113 recuerda las últimas 30 mediciones, lo cual permite al usuario encontrar previas mediciones con facilidad y monitorear cambios en la presión.

Control de inflación óptima: Solamente los productos de Omron utilizan IntelliSense para automáticamente aplicar la cantidad correcta de presión y así obtener lecturas rápidas, precisas y más cómodas.

Detección de latido irregular: este monitor alerta al usuario de la presencia de latidos irregulares potencialmente peligrosos.

Indicador de hipertensión: la unidad alertara el usuario cuando la presión arterial exceda los estándares establecidos por reconocidas organizaciones internacionales.



Figura 32: Omron Medidor de presión arterial

3.1.34 Especificaciones técnicas

Modelo..... HEM – 7113

Pantalla..... Pantalla Digital LCD

Rango de Medición..... Presión: 0 a 299 mmHg, Pulso: 40 a 180/min

Precisión/calibración..... Presión: ± 3 mmHg o 2% de lectura, Pulso: ± 5 % de lectura

Inflado Automático por bomba eléctrica

Desinflado.....	Válvula de liberación automática de presión
Liberación rápida de Presión.....	Válvula de liberación automática
Detección de presión.....	Sensor de presión capacitivo
Método de medición.....	Método oscilo métrico
Detección de pulso.....	Sensor de presión de tipo de capacitancia
Fuente de alimentación.....	4 pilas "AAA" de 1.5 V
Vida útil de las pilas.....	Aproximadamente 300 usos con 4 pilas nuevas
Tamaño del brazalete.....	Aproximadamente 146 mm x 446 mm. ³⁴

Microcode Studio

Microcode Studio es un software gratuito, el cual está diseñado para que se pueda ejecutar en los sistemas operativos Windows de Microsoft, es un editor de texto como el bloc de notas de Windows, el programa está hecho principalmente para la programación de los microcontroladores, este software puede trabajar con muchos tipos de microcontroladores. Este software al momento de realizar un programa da varios archivos con los cuales se puede elegir el formato adecuado a utilizar.

El microCode Studio ayuda de una manera sencilla a crear programas y con la ayuda del compilador se puede identificar si existe algún error en el código, es importante mencionar que este programa trabaja en lenguaje Basic. Como trabaja con lenguaje Basic el microCode Studio también soporta sentencias en Assambler, si es necesario al momento de programar se tuviera que poner sentencias en assambler si lo puede recibir pero llamando por medio de subrutinas en el programa que se está realizando.

Isis Profesional

El módulo que presenta ISIS es un programa en donde se puede dibujar sobre un área de trabajo, un circuito que después se podrá simular.

³⁴ *OMRON monitores de Presion Arterial.* (s.f.). consultado el 14 de Julio de 2014, de http://www.omronmexico.com/productos_med_presion_hem7113int_new.htm#7113

Los componentes se sitúan sobre un área determinada por el programa y desde ahí se va estructurando el circuito con los símbolos de los componentes que se van uniendo por medio de conexiones simples o por medio de buses que generan mayor aprovechamiento del espacio y una mayor capacidad de estructuración de los circuitos.

Se pueden también cargarse programas de microcontroladores virtualmente para ser simulados y poder estudiar las variables electrónicas requeridas para poder avanzar en el diseño de los sistemas electrónicos necesarios para realizar proyectos.

3.1.35 Área de Trabajo

El entorno que ofrece ISIS profesional es:

- Ventana de vista completa.
- Barra de herramientas.
- Barra de menús.
- Barra de títulos.
- Ventana de componentes.
- Barra de estado.
- Zona de trabajo.

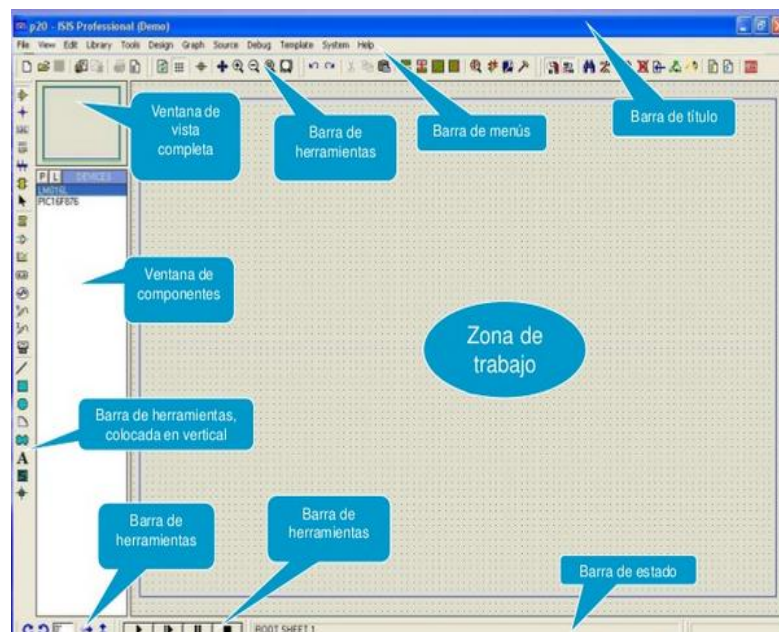


Figura 33: Área de trabajo de ISIS

3.1.35.1 Barra de título

Se encuentra situada en la parte superior de la pantalla, ahí se muestra el icono del programa, el nombre del fichero abierto (p20), la leyenda “ISIS Professional (Demo)” y, en ocasiones mensajes de que el programa ha entrado en un modo particular de funcionamiento (por ejemplo, Animating cuando se realiza una simulación).

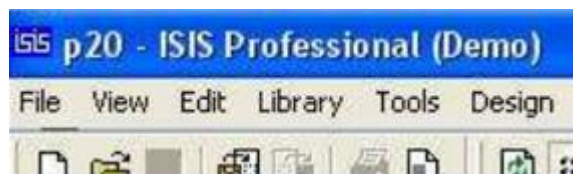


Figura 34: Barra de título de ISIS

3.1.35.2 Menús

Permite el acceso a la mayor parte de opciones que tiene el programa, sin embargo existen algunas opciones que están solo disponibles en los iconos de la barra de herramientas.

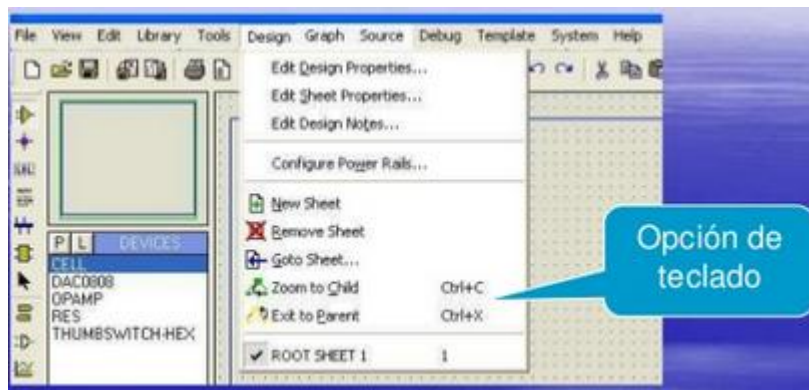


Figura 35: Menú de ISIS

3.1.35.3 Herramientas

Las herramientas son varias y se puede colocar en cualquier parte de la pantalla.

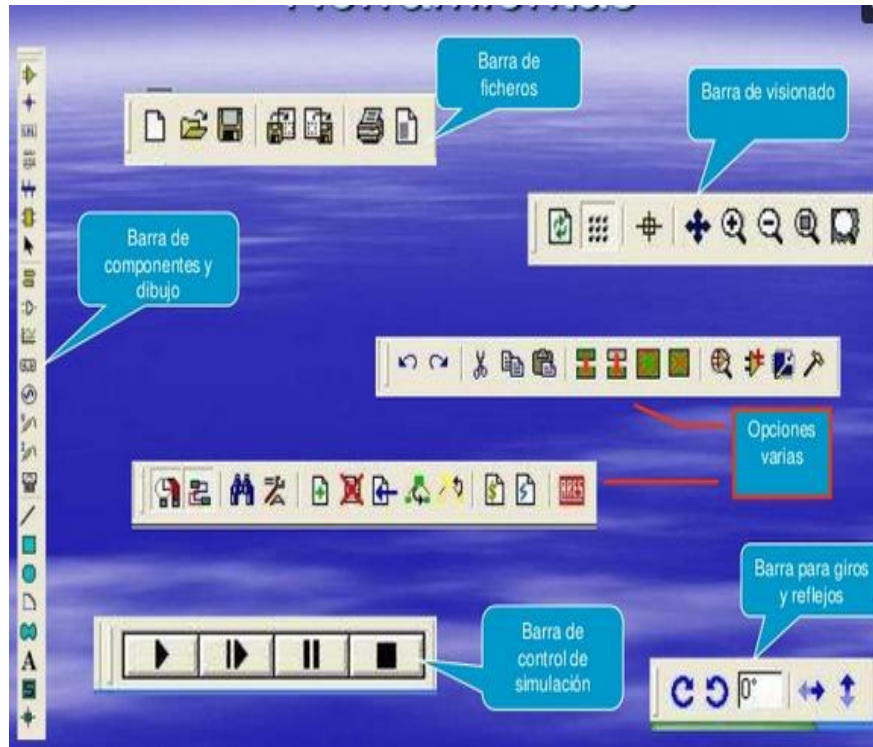


Figura 36: Herramientas de ISIS

3.1.35.4 Área de trabajo

Tiene este aspecto y es donde se realizará los circuitos³⁵

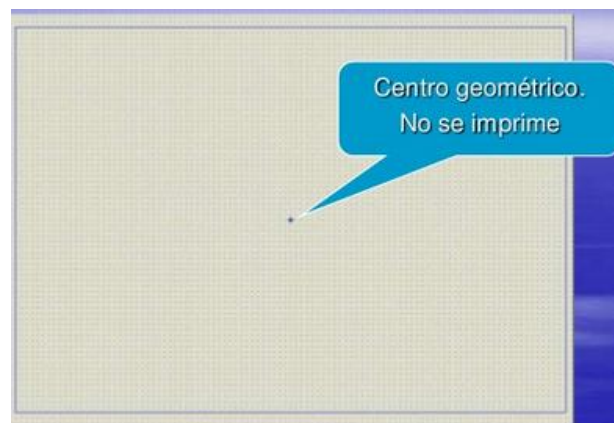


Figura 37: Area de trabajo de ISIS

³⁵ ISIS DE PROTEUS. (s.f.). Consultado el 18 de Julio de 2014, de <http://es.slideshare.net/ayreonmx/manual-del-proteus>

CAPÍTULO IV

DISEÑO E IMPLEMENTACION DEL SISTEMA DE COMUNICACIÓN OMROM – CELULAR PARA MEDIR LA PRESIÓN ARTERIAL.

Software y Hardware Arduino

4.1.1 Instalando Arduino

Es indispensable haber instalado previamente el entorno de desarrollo para poder hacer uso de las distintas funciones que ofrece Arduino, en donde se puede generar el sketch con el código del programa que se desea cargar en el dispositivo.

Primero se descarga la última versión de arduino disponible en la propia página web de arduino, una vez descargado ejecutamos el archivo y seguimos los pasos del asistente de instalación, ya instalado Windows le detectará automáticamente.

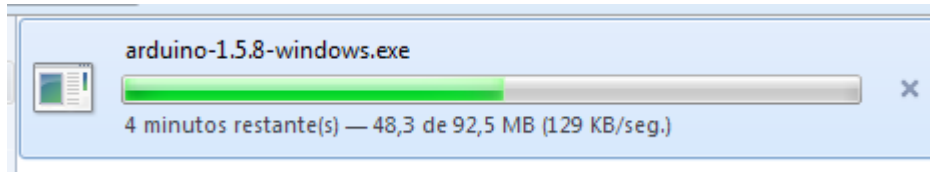


Figura 38: Descargando Arduino

Completada la instalación e identificado el puerto en el que está conectado el Arduino, se debe seleccionar el modelo concreto de la placa. Para ello se accede al menú “Herramientas”, dentro del mismo se coloca el cursor sobre la pestaña “Board”, y a continuación se da click sobre la versión correspondiente al modelo, en este caso, “Arduino Mega”.

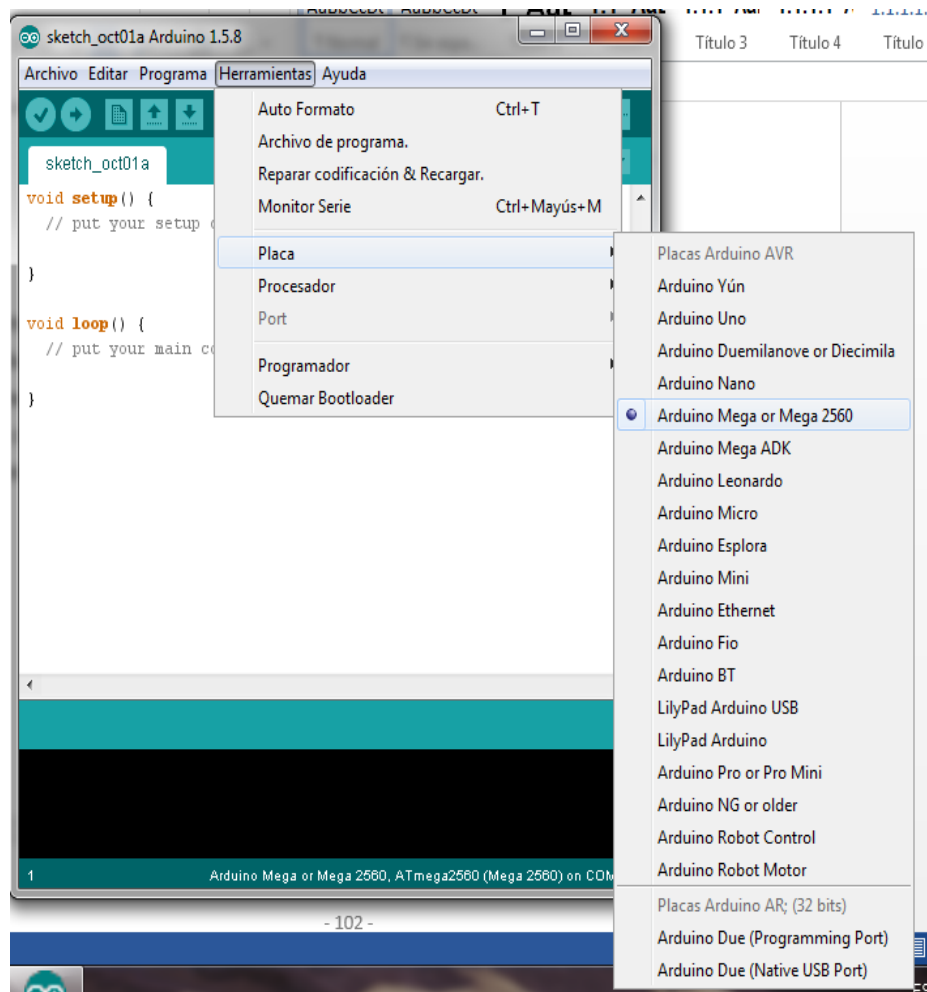


Figura 39: Menú herramientas de Arduino

Para poder empezar a trabajar con la placa una vez ya instalado y configurado el IDE de Arduino se debe generar un nuevo sketch y empezar a escribir el código de programación con las instrucciones que se desee que ejecute.

4.1.2 Primera Prueba con Arduino

A partir de este momento se empieza a desarrollar la parte experimental del proyecto, como es lógico hasta llegar a construir un dispositivo de cierta complejidad se debe pasar por una serie de etapas más sencillas hasta lograr alcanzar el objetivo final.

Es así que para conocer más a fondo el entorno de desarrollo Arduino se empezó por realizar pruebas con un sketch básico llamado "Hola Mundo" que tiene como finalidad hacer parpadear un led de acuerdo a unos intervalos de tiempo predefinidos por el usuario en el código del sketch, con el objetivo de familiarizarse con la mecánica para la carga y ejecución de programas. La prueba siguiente se realiza en proteus, se debe tener instalado las librerías para esta función previamente y de esta manera se podrá simular la placa Arduino como si fuera una real.

1. Se procede a generar el código que se utilizará.
2. Dirigirse al entorno Arduino, en donde se busca en el menú la opción Archivo y se procede a seleccionar Preferencias como se ve a continuación.

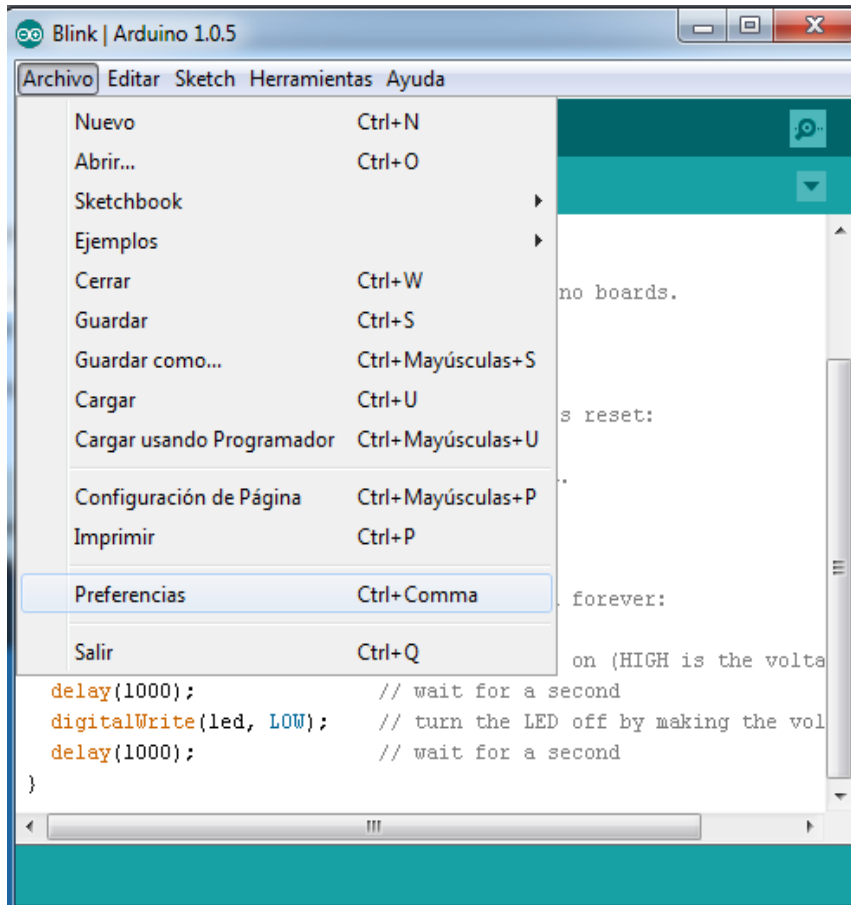


Figura 40: Menú Archivo de Arduino

3. Se mostrará las siguientes opciones marcamos la opción carga y aceptamos como se ve en la figura 43.

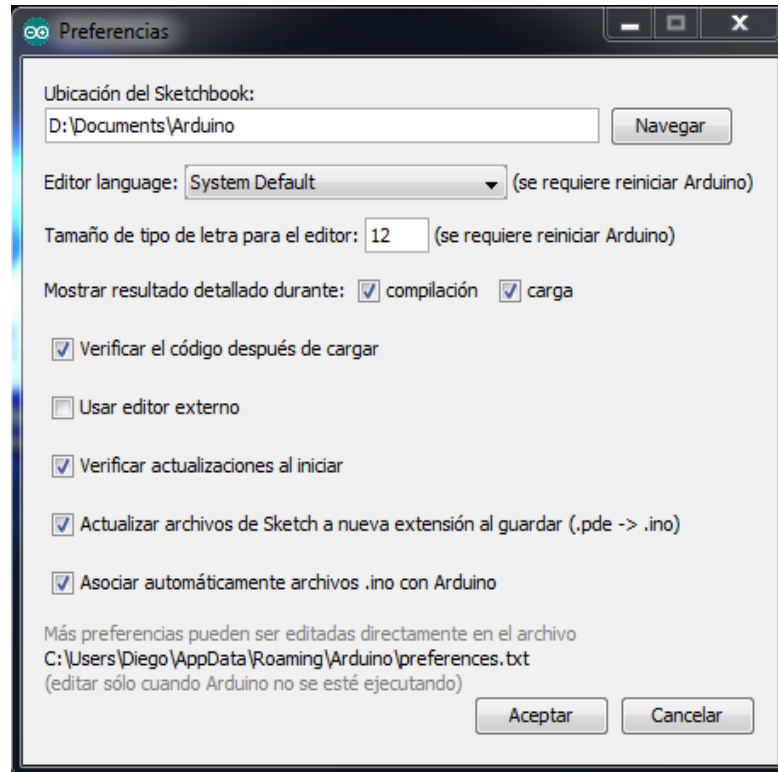


Figura 41: Opción Preferencias del menú archivo de Arduino

4. El momento del compilar el sketch aparecerá lo siguiente, se debe fijarse la dirección en la que se está compilando el código que se utilizara.

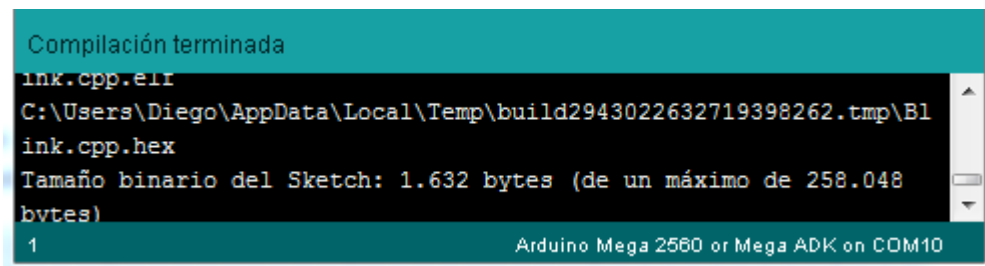


Figura 42: Compilación en Arduino

5. En esa dirección se deberá el archivo que se va a utilizar, en este caso Blink.cpp.hex y se lo carga al Arduino en la opción PROGRAM FILE como se muestra a continuación.

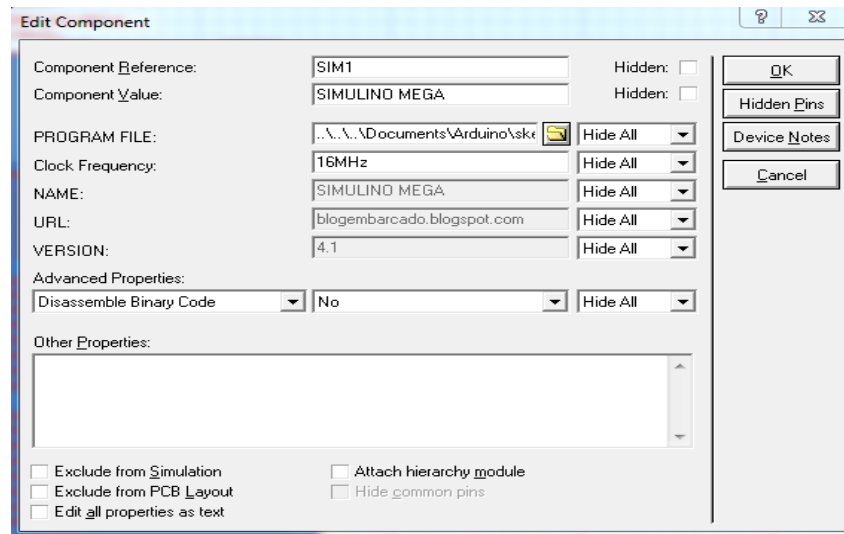


Figura 43: Selección de la Ubicación de carga del sketch en Arduino

6. A continuación se muestra como queda el circuito armado listo con el código cargado para empezar a funcionar.

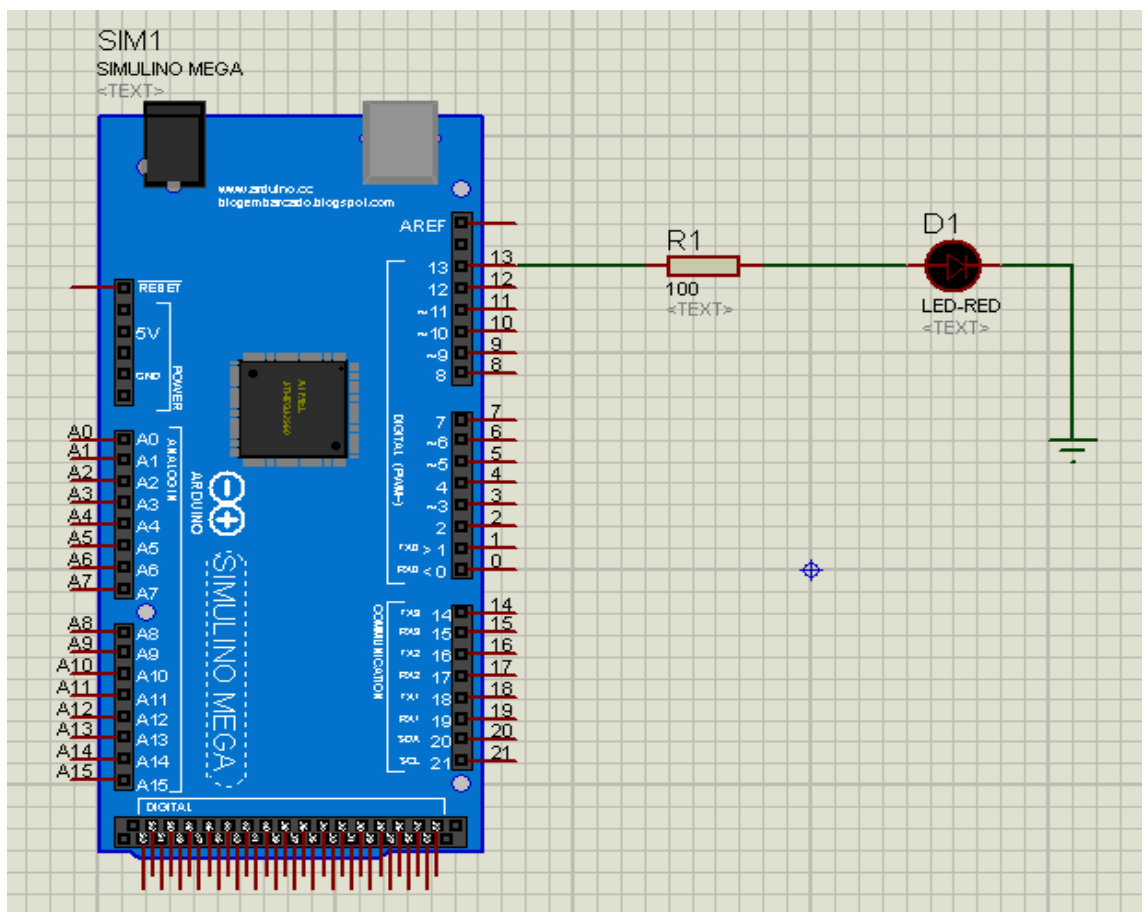


Figura 44: Esquema de montaje en ISIS con placa Arduino

Lo único que se necesita para lograr cumplir el primer ejercicio son: una placa arduino, un led y el cable usb, en este caso el modelo del arduino es MEGA, que incorpora un LED integrado en la propia placa de esta forma hacer parpadear el LED resulta muy sencillo, pero también permite conectar uno externamente utilizando el pin digital 13, teniendo en cuenta que si se desea utilizar otro pin digital para conectar un led se recomienda intercalar una resistencia externa de 1kΩ.

Para realizar el montaje simplemente se debe identificar el ánodo y el cátodo del LED, y conectarlos directamente en el pin 13 y GND respectivamente como se muestra en la figura.

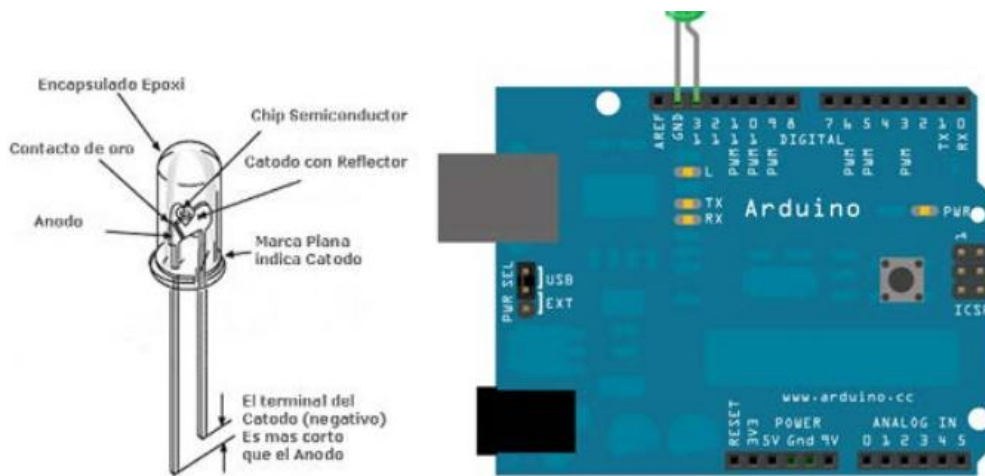


Figura 45: Esquema de montaje para hacer parpadear un LED con Arduino

Luego se debe generar el código con las instrucciones que queremos que lleve a cabo el Arduino.

```
void setup() {  
  pinMode(13, OUTPUT);  
}  
  
void loop() {  
  digitalWrite(13, HIGH);  
  delay(1000);  
  digitalWrite(13, LOW);  
  delay(1000);  
}
```

The image shows a screenshot of the Arduino IDE interface. The title bar reads "Blink Arduino 1.5.8". The menu bar includes "Archivo", "Editar", "Programa", "Herramientas", and "Ayuda". Below the menu bar is a toolbar with icons for checkmark, play, file, upload, and download. The main editor area shows the following code:

```
void setup() {  
  pinMode(13, OUTPUT);  
}  
  
void loop() {  
  digitalWrite(13, HIGH);  
  delay(1000);  
  digitalWrite(13, LOW);  
  delay(1000);  
}
```

Figura 46: Código generado en Arduino

Se puede ver que el código es bastante sencillo, lo que se debe tener en cuenta es que a todos los sketch siempre se les debe incluir funciones “void setup()” y “void loop()”, ya que sin ellas cuando se compile el código siempre dará un error, y no se podrá proceder con la carga en Arduino.

Luego se procede a compilar el sketch en busca de posibles errores, en caso de no existir se deberá cargar el programa a la placa, para lo cual no se debe olvidar tener conectado el arduino al puerto usb.

Funcionamiento Específico del Omrom HEM – 7113

Al realizar el desmontaje del equipo resultó fácil darse cuenta que la placa venía con pines disponibles Tx, Rx y GND (los datos se están enviando por comunicación serial) ya que se encontraban marcados con sus respectivas siglas, situación que no sucedió

con otros 2 modelos anteriores en los cuales no se podía acceder a extraer la información necesaria que enviaba el omrom al realizar las mediciones respectivas.

Como se puede observar en la figura 47 se muestra como está constituido el circuito de la placa y los cables soldados que se colocó en los pines respectivos para obtener los datos de la medición.

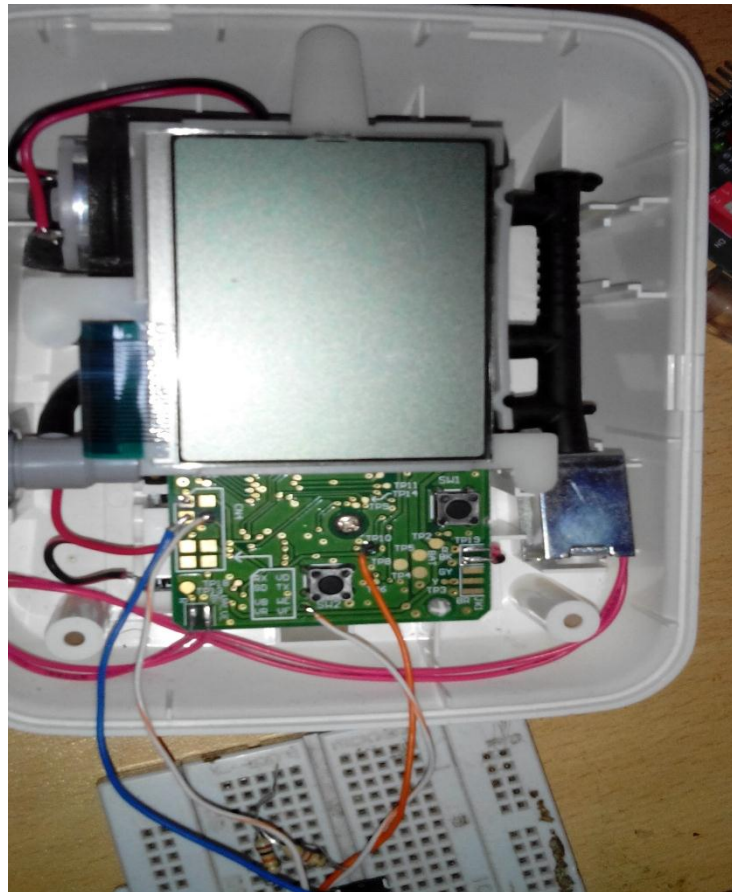


Figura 47: Circuito de la Placa Omron

La estructura interna que presenta la placa del omrom esta conformada de un sensor de presión, que es el encargado de recolectar las oscilaciones provenientes del brazalete, el cual se encuentra conectado a un amplificador de instrumentación el mismo que se encarga de acondicionar la señal hacia un microcontrolador.

Al momento que el dispositivo realiza la medición se crean dos señales de presión, la primera es la señal de desinflado calibrada, la segunda medición obtenida a partir de la amplificación de la anterior y un filtro pasa altos, esta es la que mostrará las pequeñas oscilaciones que impone el arterial sobre el brazalete durante el desinflado.

El microcontrolador es el encargado de controlar la válvula de escape, digitalizar las 2 señales descritas y enviar los datos a un display (LCD).

Presión máxima =175mmHg.

Presión mínima =50mmHg.

4.1.3 Recepción de la Señal del Omrom

Para lograr recibir los datos a través del puerto serial, se conectó a un software en el cual se puede visualizar la información que nos da el medidor de presión arterial. El software que primero se utilizó fue el serial comunicator de Microcode studio en donde se visualizó cadenas de caracteres y entre los mismos un código en hexadecimal.

Para poder visualizar en microcode se debe ingresar a la opción view y dar click sobre serial comunicator, luego se configura el número del puerto (es el que se detecta en el computador), el baurate que debe ser 9600 baudios, la paridad que debe estar sin paridad, el tamaño del bit que debe ser 8 y los bits de parada deben estar en 1.

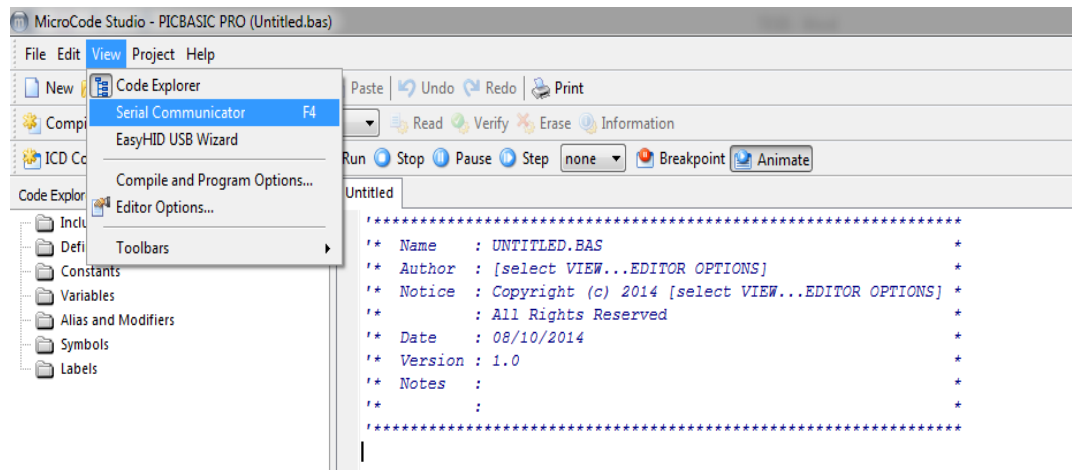


Figura 48: Selección de Serial Comunicator en la opción view de microcode

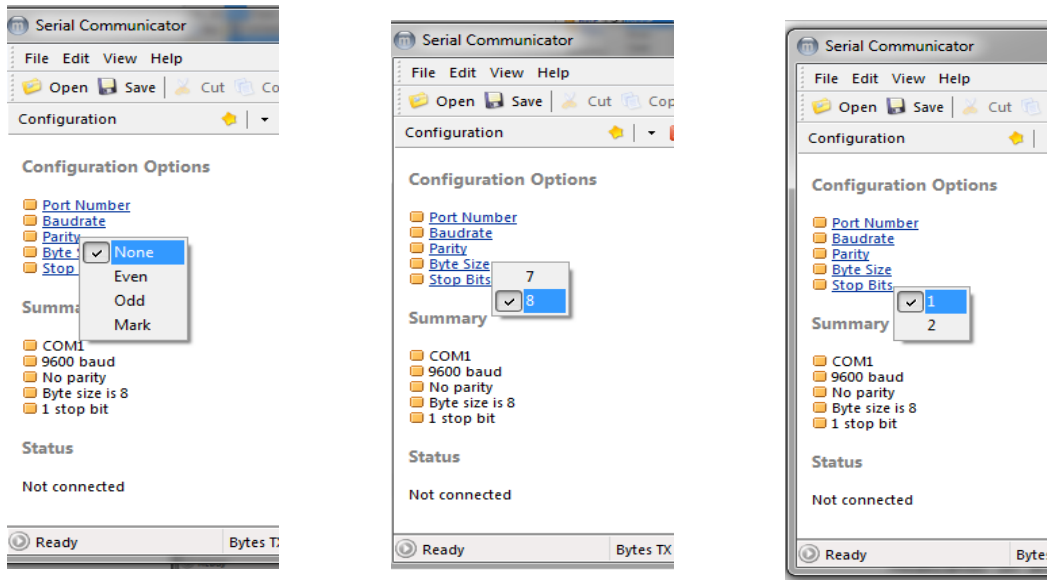


Figura 49: Valores de Configuración de Serial Comunicador

Luego de haber realizado las mediciones de la presión y pulso desde el omron se observabó un código en hexadecimal. Realizando un análisis lógico y luego de descartar las cadenas que tenían menos caracteres se eligió para realizar las pruebas respectivas una cadena de caracteres que fue del siguiente tipo.

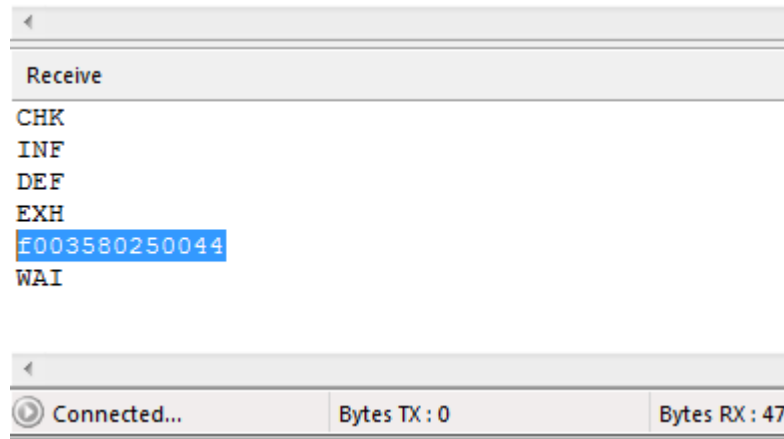


Figura 50: Código 1 extraído del Omron

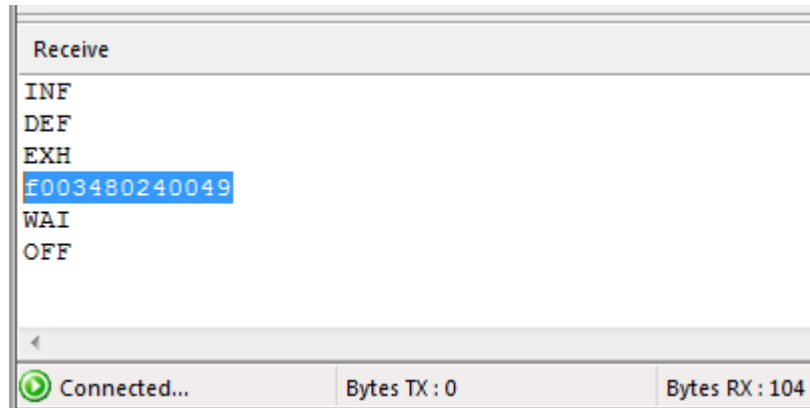


Figura 51: Código 2 extraído del Omron

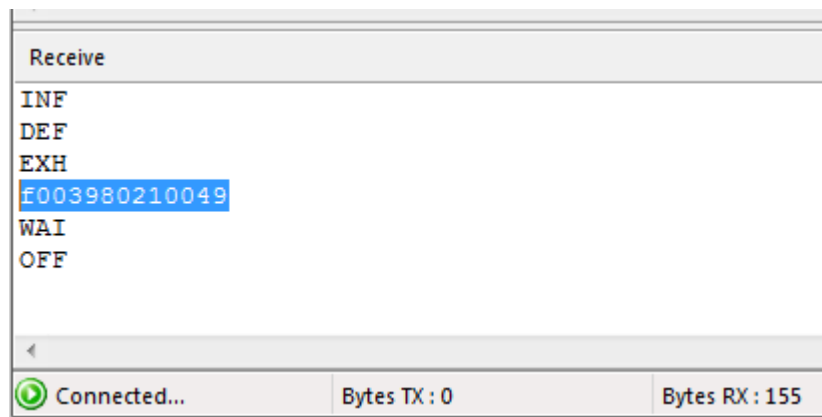


Figura 52: Código 4 extraído del Omron

Luego de realizar varias mediciones se observó que los 3 primeros caracteres no variaban y asumiendo que toda la cadena de caracteres se encontraba después de las siglas EXH en valores hexadecimales, se procedió a realizar pruebas de conversión a decimal para comparar con los datos que se mostraban en el display del omrom.

Es así que luego de realizar algunas pruebas se obtuvo información de que los 3 primeros caracteres no tenían relación alguna con los datos de la medición del dispositivo ya que solo indicaba donde empieza a receptarse los datos.

Otro aspecto importante que se obtuvo es que los caracteres que se encontraban en la cuarta, quinta, octava, novena, decima segunda y décima tercera posición variaban constantemente, no siendo así el caso de los ceros intermedios que variaban únicamente en que los datos que arrojaba el medidor de presión arterial resultaban impares presentándose en escasas ocasiones.

Con esta información y realizando la conversión de hexadecimal a decimal se llegó a la siguiente conclusión:

Los datos obtenidos ya en decimal se multiplican por dos ya que después de algunas pruebas realizadas es el valor medio de las medidas mostradas en el display del omrom, las cuales se describen en las siguientes ecuaciones.

$$\text{Presion sistólica} = (\text{cadena hexadecimal}) * 2$$

$$\text{Presión diastólica} = (\text{cadena hexadecimal}) * 2$$

$$\text{Pulso} = (\text{cadena hexadecimal})$$

La cadena hexadecimal que se obtiene debe ser transformada a decimal.

Cuando se tiene la cadena hexadecimal se puede interpretar los datos de la siguiente manera: el cuarto, quinto, sexto y séptimo carácter representan los dígitos de un número que se relaciona con la presión sistólica.

El octavo, noveno, décimo y décimo primero carácter son los dígitos de un número que se relaciona con la presión diastólica.

Los dígitos que relacionan el número del valor del pulso son: el décimo segundo y décimo tercer carácter.

Ya con los datos identificados y logrando despejar algunas incógnitas el propósito era conseguir que un programa realice automáticamente la transformación de hexadecimal a decimal anteriormente explicada, por lo que se optó trabajar con el software y hardware arduino.

4.1.4 Código que se utilizó para la transformación.

Se empieza por la declaración de variables, y la función 'setup' con la que se inicia el programa. En la función 'setup' se establece el valor de 'baudrate' correspondiente a los puertos serie. En este caso, se necesita dos puertos para el intercambio de datos, uno dedicado a la comunicación entre el Omron y la plataforma Arduino ('Serial1'), y otro para la comunicación entre la placa Arduino y la PC ('Serial0').

```
#include <string.h>

// DECLARACION DE VARIABLES PARA CALCULO DE LA PRESION
int i=0;           // Contador
int a; int b;     // Sistolica
float c; float S; // Sistolica
int a1; int b1;  // Diastolica
float c1; float D; // Diastolica
int a2; int b2; float P; // Pulso
char var[255];   // Variable que guarda datos que vienen del Omron

void setup() {
  Serial.begin(9600); // baudrate del puerto con la PC
  Serial1.begin(9600); // baudrate del puerto con el Omron
  Serial.println("He recibido:");
  delay(10000);
}
```

Una vez que se tiene todo correctamente configurado y listo para el inicio del programa, se deberá centrar en la función principal, la famosa void 'loop', obligatoria en cada sketch de Arduino. Cada vez que cumpla cierta condición ejecutara la parte del código correspondiente y tras llegar al final del proceso se volverá a entrar en el 'loop', a la espera de nuevos datos.

A continuación se presenta el código para la recepción de datos desde el Omron por el puerto serie1 del Arduino, datos que vienen en formato hexadecimal que serán transformados a datos decimales tanto de la presión sistólica, diastólica y pulso. Como se explicó anteriormente solo una parte de la trama tiene los datos que se necesita, es por aquello que después de leer la parte buscada se presentara la letra 'W' y se ejecuta la transformación.

```
void loop() {

  //----Preguntamos si hay datos desde el Omron

  if (Serial1.available()>0) {
    char e= Serial1.read();
    var[i]=e; // variable que almacena datos que vienen del Omron

    if(e=='W'){ // condicion para traduccion de datos
```

A continuación se muestra el calculo de la medida Sistolica de la Presion

```
// CALCULO SISTOLICA
// Sistolica 1
if (var[i-12]=='A'){
a=(var[i-12]-55)*16;
}else {
  if (var[i-12]=='B'){
a=(var[i-12]-55)*16;
  }else {
    if (var[i-12]=='C'){
a=(var[i-12]-55)*16;
    }else{
      if (var[i-12]=='D'){
a=(var[i-12]-55)*16;
      }else{
        if (var[i-12]=='E'){
a=(var[i-12]-55)*16;
        }else{
          if (var[i-12]=='F'){
a=(var[i-12]-55)*16;
          }
          else{
            a=(var[i-12]-48)*16;
          }
        }
      }
    }
  }
}

// Sistolica 2
if (var[i-11]=='A'){
b=(var[i-11]-55);
}else{
if (var[i-11]=='B'){
b=(var[i-11]-55);
}else{
  if (var[i-11]=='C'){
b=(var[i-11]-55);
  }else{
    if (var[i-11]=='D'){
b=(var[i-11]-55);
    }else{
      if (var[i-11]=='E'){
b=(var[i-11]-55);
      }else{
        if (var[i-11]=='F'){
b=(var[i-11]-55);
        }
        else{
          b=(var[i-11]-48);
        }
      }
    }
  }
}

c=(var[i-10]-48)*0.0625;
S=(a+b+c)*2;
```

A continuación se muestra el cálculo para la medida Diastólica de la Presión

```
// CALCULO DIASTOLICA
// Diastolica 1
if (var[i-8]=='A'){
al=(var[i-8]-55)*16;
}else {
  if (var[i-8]=='B'){
al=(var[i-8]-55)*16;
  }else {
    if (var[i-8]=='C'){
al=(var[i-8]-55)*16;
    }else{
      if (var[i-8]=='D'){
al=(var[i-8]-55)*16;
      }else{
        if (var[i-8]=='E'){
al=(var[i-8]-55)*16;
        }else{
          if (var[i-8]=='F'){
al=(var[i-8]-55)*16;
          }
          else{
al=(var[i-8]-48)*16;
          }}}}
        }}}}

//Diastolica 2
if (var[i-7]=='A'){
bl=(var[i-7]-55);
}else{
  if (var[i-7]=='B'){
bl=(var[i-7]-55);
  }else{
    if (var[i-7]=='C'){
bl=(var[i-7]-55);
    }else{
      if (var[i-7]=='D'){
bl=(var[i-7]-55);
      }else{
        if (var[i-7]=='E'){
bl=(var[i-7]-55);
        }else{
          if (var[i-7]=='F'){
bl=(var[i-7]-55);
          }
          else{
bl=(var[i-7]-48);
          }}}}
        }}}}

cl=(var[i-6]-48)*0.0625;
D=(al+bl+cl)*2;
```


A continuación se presenta el Cálculo de la medida del Pulso

```
// CALCULO DEL PULSO
//Pulso 1
if (var[i-4]=='A'){
a2=(var[i-4]-55)*16;
}else {
    if (var[i-4]=='B'){
a2=(var[i-4]-55)*16;
    }else {
        if (var[i-4]=='C'){
a2=(var[i-4]-55)*16;
        }else{
            if (var[i-4]=='D'){
a2=(var[i-4]-55)*16;
            }else{
                if (var[i-4]=='E'){
a2=(var[i-4]-55)*16;
                }else{
                    if (var[i-4]=='F'){
a2=(var[i-4]-55)*16;
                    }
                    else{
a2=(var[i-4]-48)*16;
                    }
                }
            }
        }
    }
}

//Pulso 2
if (var[i-3]=='A'){
b2=(var[i-3]-55);
}else{
    if (var[i-3]=='B'){
b2=(var[i-3]-55);
    }else{
        if (var[i-3]=='C'){
b2=(var[i-3]-55);
        }else{
            if (var[i-3]=='D'){
b2=(var[i-3]-55);
            }else{
                if (var[i-3]=='E'){
b2=(var[i-3]-55);
                }else{
                    if (var[i-3]=='F'){
b2=(var[i-3]-55);
                    }
                    else{
b2=(var[i-3]-48);
                    }
                }
            }
        }
    }
}

P=(a2+b2);
```

A continuación se muestra una condición como control para posibles errores en la medición o cuando no esté en uso, también se puede ver como se incrementa el contador para almacenar nuevos datos en otras variables y se termina reiniciando el contador.

```
// Condicion de error
if (S<=0 || D<=0 || P<=0)
{
Serial.println("Error");
Serial.println("Mal colocado o");
Serial.println("No esta en uso");
}else{
// Condicion de medida correcta
if (S>0 && D>0 && P>0)
{
Serial.print("La presion sistolica es: ");
// .....

Serial.println(D);
Serial.print("El pulso es: ");
Serial.println(P);
}
}
}
i=i+1; // incrementamos contador para almacenar datos
// en nueva posicion
if(i==51){ // controlamos que no exeda las posociones en 51
i=0;
}
}
}
```

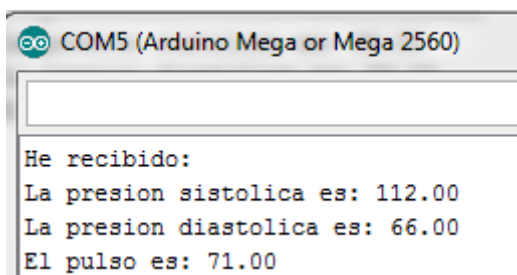


Figura 53: Resultado 1 de la transformación de los valores obtenidos con el Omron en programación Arduino

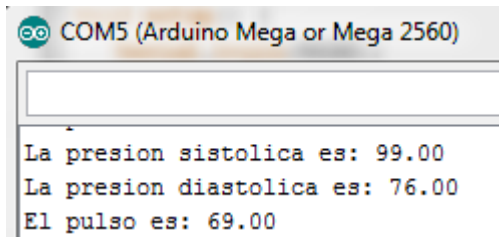


Figura 54: Resultado 2 de la transformación de los valores obtenidos con el Omron en Programación Arduino

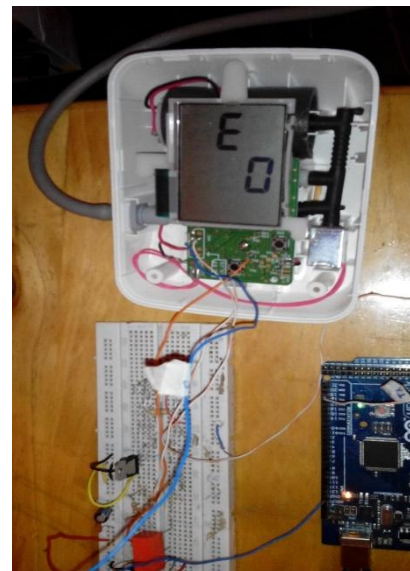
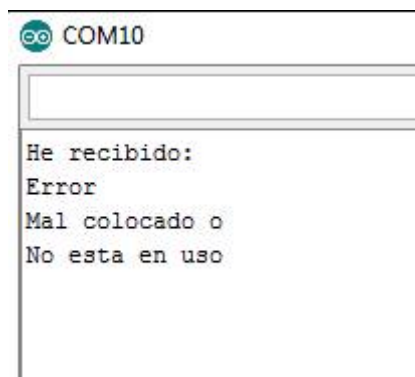


Figura 55: Resultado 3 de la transformación de los valores obtenidos con el Omron en programación Arduino

4.1.5 Captura de datos del Omron con proteus

En esta parte antes de utilizar la placa real de Arduino, se presenta una opción en proteus bastante sencilla de manejar, y se seguirá los siguientes pasos:

1. Generar sketch

2. Compilar para ver posibles errores
3. Buscar el archivo .cpp.hex en la dirección de compilación que aparezca como se explicó anteriormente en el encendido del led en los pasos 2-4.
4. Cargar el programa en la placa simulada en proteus, el esquema del circuito queda como se ve en la siguiente figura.

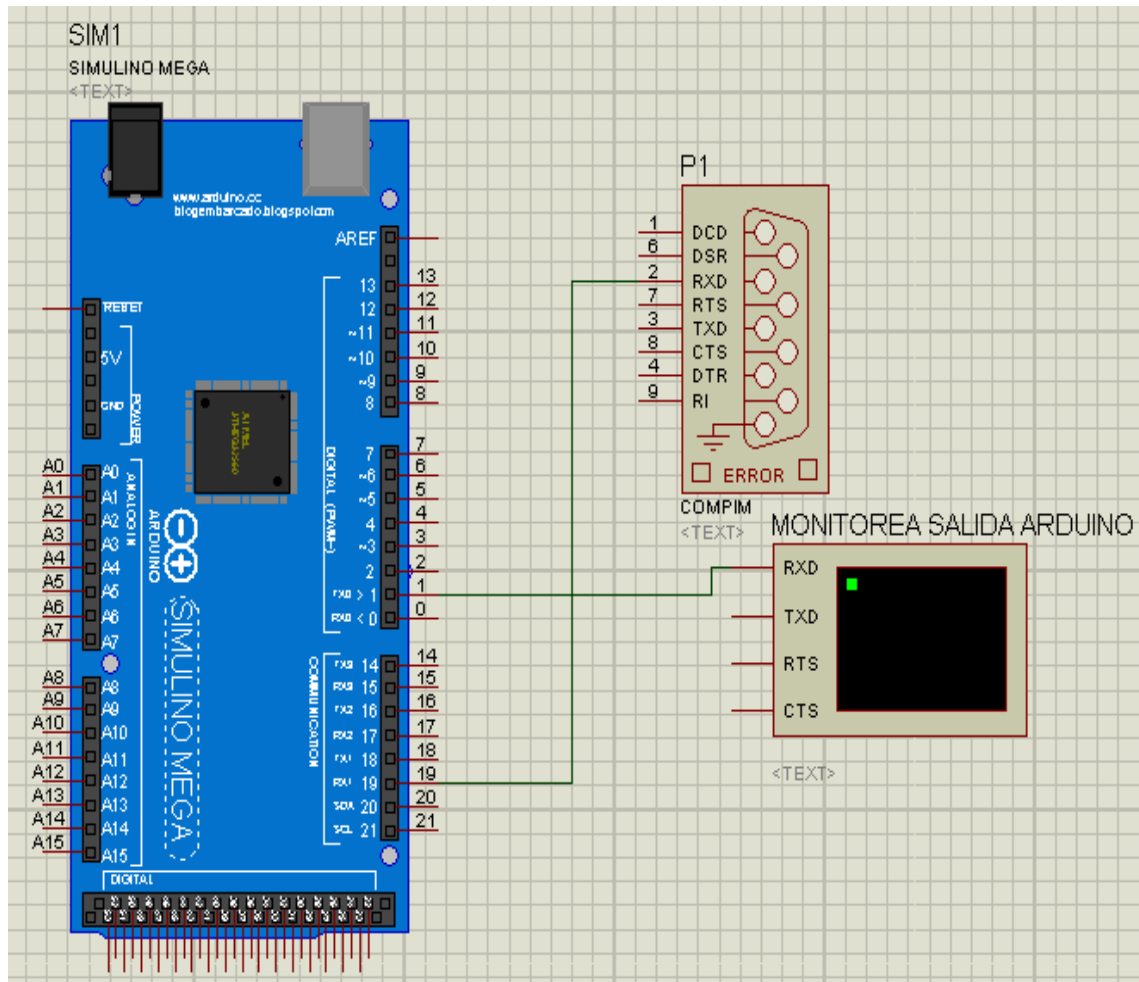


Figura 56: Montaje del circuito en Proteus para capturar los datos del Omron

5. No se debe olvidar configurar en el puerto serial virtual la velocidad que utiliza el Arduino y el dispositivo Omron en este caso 9600 baudios figura 57.

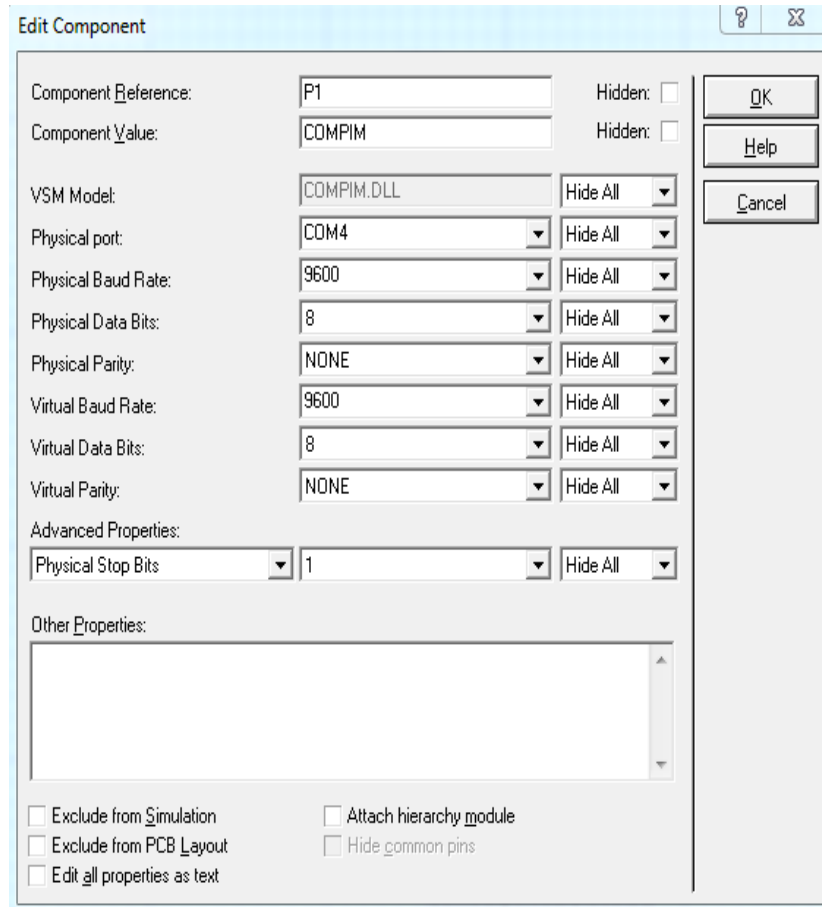


Figura 57: Valores de Configuración de puerto serial del arduino en Proteus

6. En el terminal virtual también configuramos la velocidad de transmisión como se ve en la siguiente figura 60..

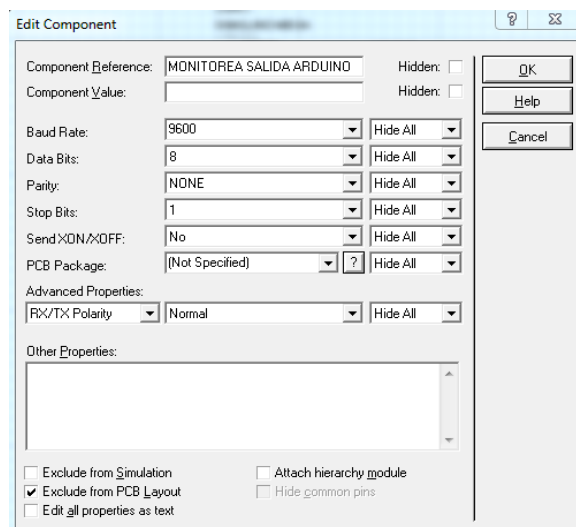


Figura 58: Valores de configuración de la velocidad de transmisión de arduino en Proteus

7. Conectar un cable usb a serial para proceder a la simulación como se ve en la siguiente figura 59.



Figura 59: Conexión del omron de cable serial a usb

8. Ya con el esquema listo y todas las configuraciones pertinentes realizadas se procede a la ejecución del programa.

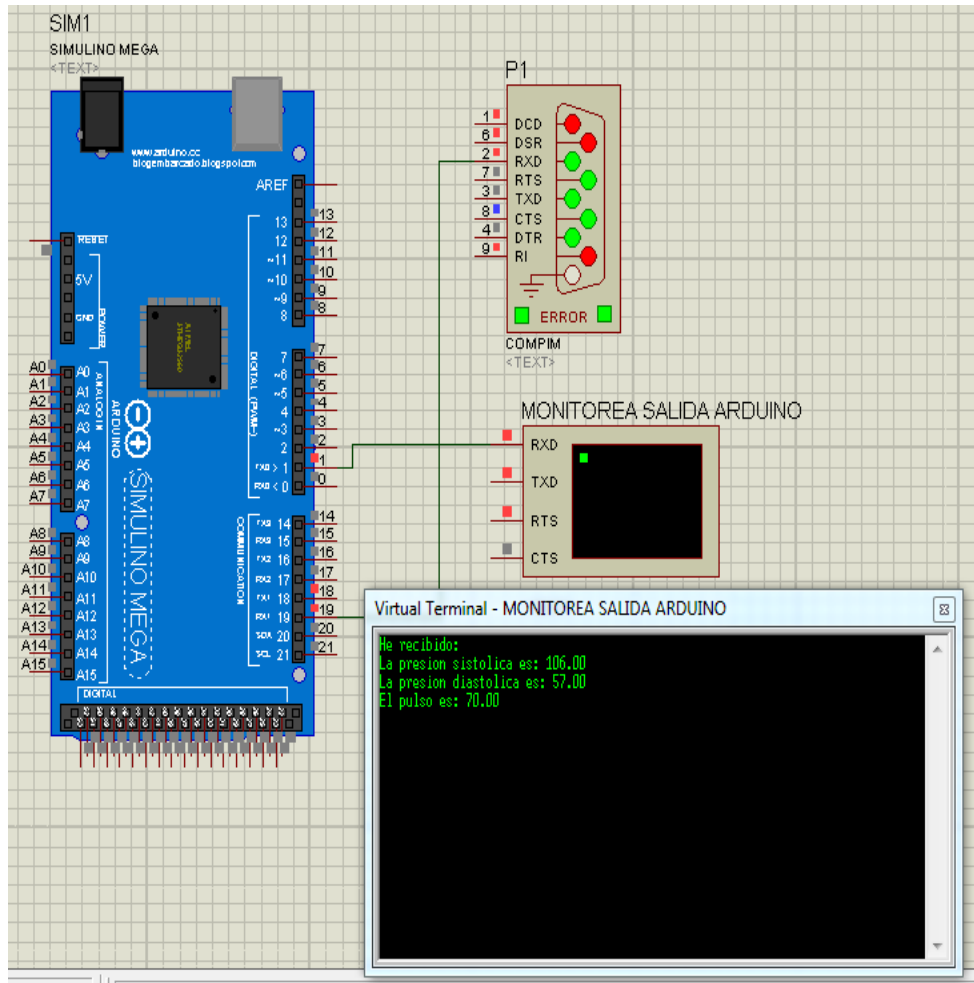


Figura 60: Resultado de los datos obtenidos del Omron en Proteus

El dispositivo Omron tiene que estar conectado al ordenador mediante el cables USB a serial como se muestra en la figura, donde se podrá ver que los datos que se obtienen en el Omron son los mismos que se traducen a la pantalla del terminal virtual en proteus una vez realizada las operaciones de trasformación de hexadecimal a decimal de los valores de la presión arterial.

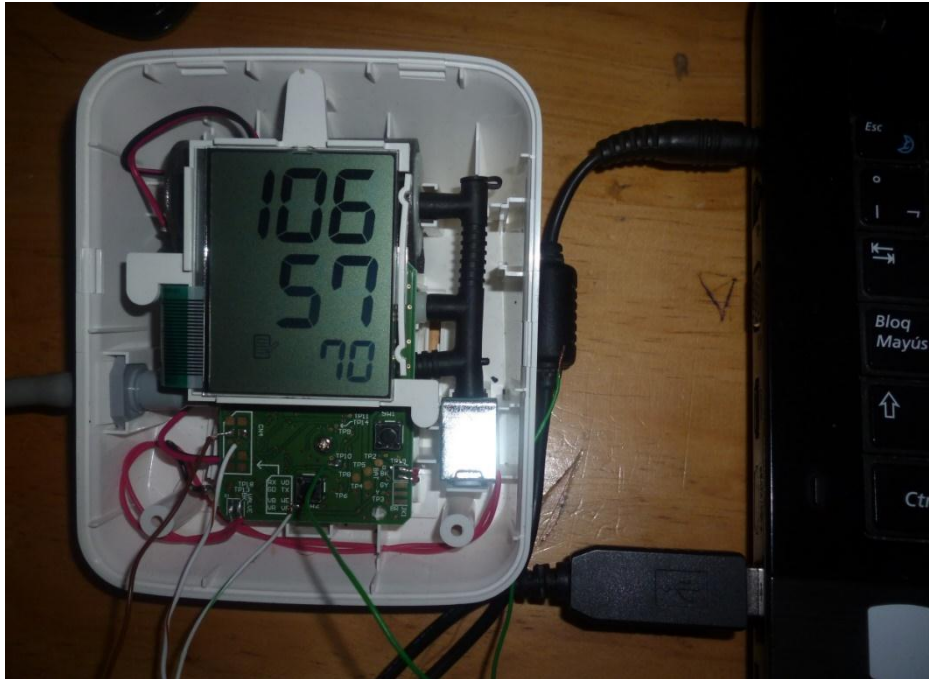


Figura 61: Conexión del Omron al Computador con cable serial a usb

Se puede observar las mediciones traducidas en el proteus, en este caso nos da que la presión sistólica es 106, la diastólica 57 y el pulso 70.

Envío de SMS por comandos AT

En esta parte se procede con la primera de las pruebas recomendadas por el fabricante del módulo GPRS/GSM, que es el envío de un SMS mediante comandos AT.

A la hora de querer realizar alguna prueba con el modulo lo primero que se debe hacer es tener en cuenta la configuración de sus jumpers. En este caso, como se desea tener una comunicación directa con el shield no es necesario cargar ningún código en la memoria de Arduino, su conexión con los jumpers son como se muestra en la figura 62.



Figura 62: Conexión de jumpers del Shield

Con los jumpers listos y con la ayuda de una cable USB a serial conectamos el Shield a la PC, los pines del shield a utilizar será el D2 conectado al RX y el D3 al TX del cable. Hay que tener en cuenta el número de puerto COM que utilizaremos en la conexión para eso nos dirigimos al Administrador de dispositivos de la PC y observamos la opción Puertos (COM y LPT) en ese caso utilizaremos el COM4 como se ve en la figura.

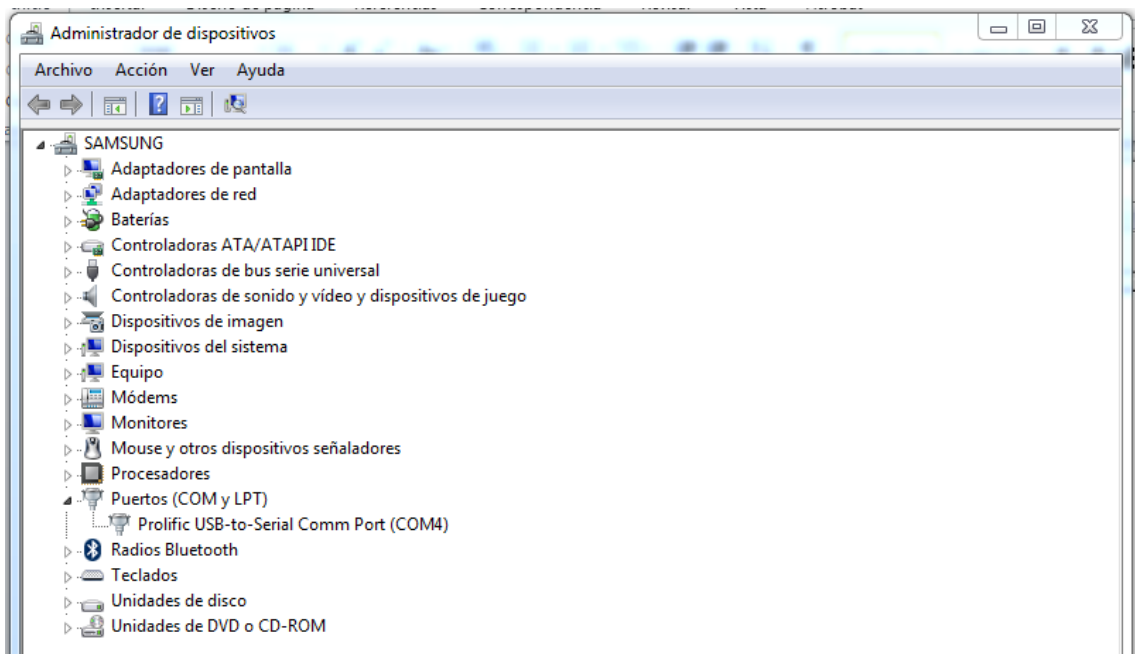


Figura 63: Puerto COM a utilizar

Al terminar con lo anterior se debe abrir el Hyper Terminal y se coloca un nombre a la nueva conexión en este caso Prueba Shield.

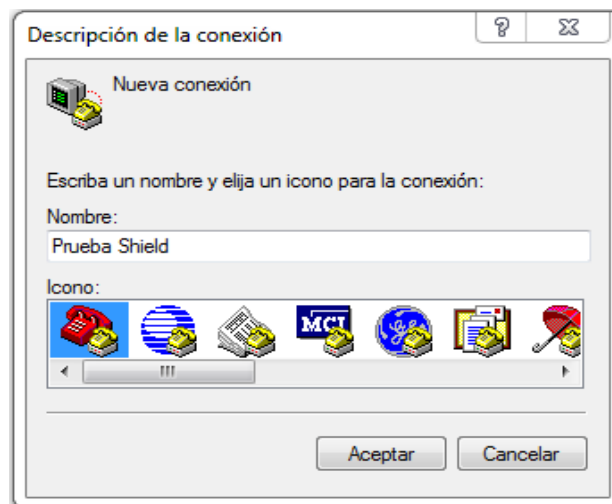
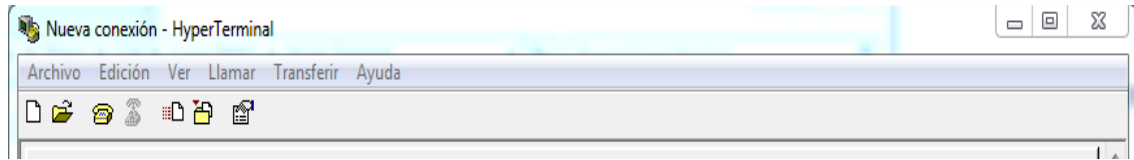


Figura 64: Entorno hyperterminal

En la siguiente ventana se debe escoger la opción Conectar y se selecciona el COM4 que estamos utilizando y aceptamos como se muestra en la siguiente figura.

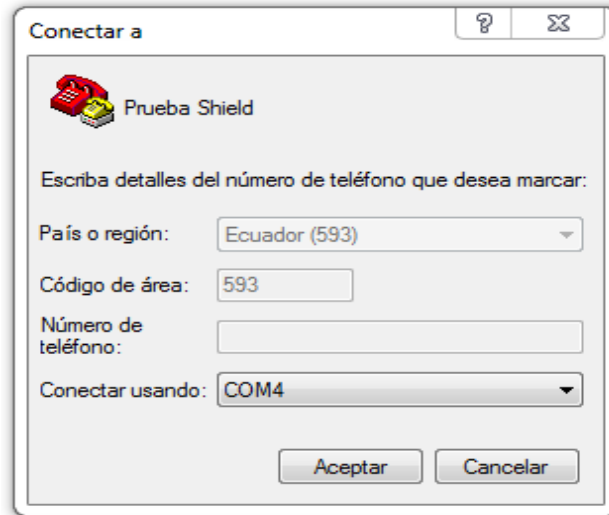


Figura 65: Selección del puerto COM en hyperterminal

Luego se debe configurar el puerto con los siguientes valores establecidos por ser comunicación serial y la velocidad a la que trabaja el Shield, se acepta y ya está listo para poder interactuar con el Shield.

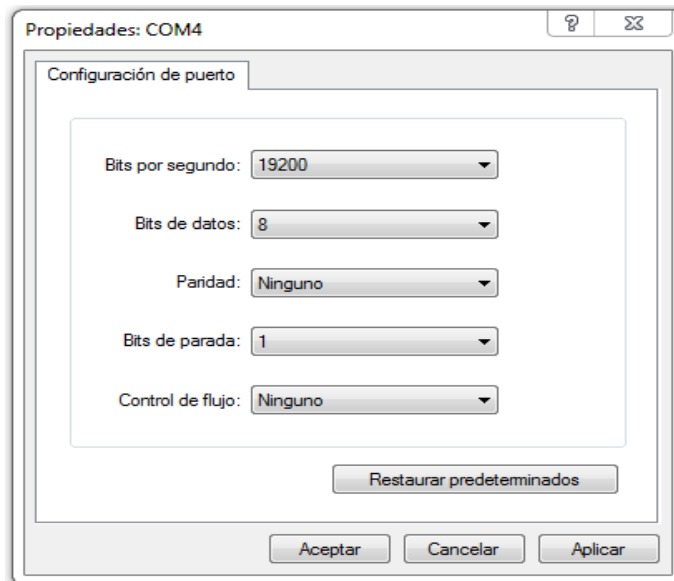
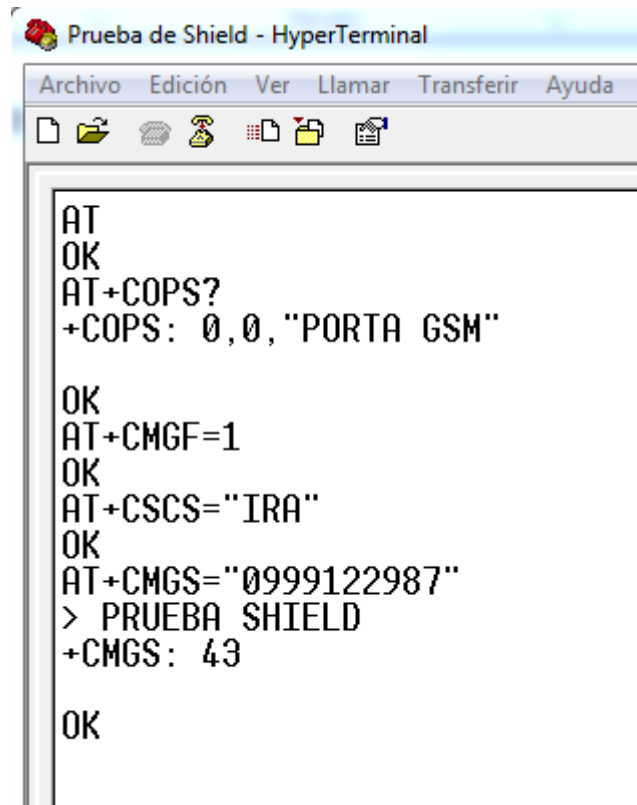


Figura 66: Valores de configuración del puerto en hyperteminal

El siguiente paso es encender el módulo. Para lo cual se mantiene presionado el botón de encendido de la shield durante 2 segundos y se empieza con el intercambio de comandos AT, obteniendo como resultado de la comunicación lo que se muestra en la figura.



```
AT
OK
AT+COPS?
+COPS: 0,0,"PORTA GSM"

OK
AT+CMGF=1
OK
AT+CSCS="IRA"
OK
AT+CMGS="0999122987"
> PRUEBA SHIELD
+CMGS: 43

OK
```

Figura 67: Prueba de recepción de datos con comandos AT

Como se muestra en la figura 67, se empieza enviando el comando "AT" para comprobar que el módulo responde correctamente ante las instrucciones, si es así, responderá con un "OK"

Para comprobar el estado de la red y ver si ya ha sido registrado y está habilitado para el envío de SMS se introduce el comando "AT+COPS?" para ver si responde correctamente con el nombre de la compañía telefónica a la que pertenece la SIM (en este caso claro).

Lo siguiente es seleccionar el modo para el envío de SMS, se debe introducir los comandos "AT+CMGF=1" y "AT+CSCS="IRA"".

Una vez recibidos los "OK" de respuesta, se puede pasar a introducir el número al que se desea enviar el SMS y el texto que se desea incluir en el mismo.

Para registrar el número se escribe el comando "AT+CMGS="numero móvil"", y se introduce el texto a continuación.

Por último, se debe presionar Ctrl+z, con lo que se indica el fin del mensaje. Inmediatamente después, el módulo procederá a su envío, y transcurridos unos segundos el SMS enviado llegará a nuestro móvil, con el texto "PRUEBA SHIELD".

Como todo salió según lo esperado y se recibió correctamente el SMS en nuestro móvil, se procede ahora a realizar la prueba del envío de un SMS a través de Arduino, sin necesidad de teclear manualmente los comandos AT.

Envío SMS a través de Arduino

Para lograr una comunicación con el módulo GPRS/GSM a través de Arduino y conseguir enviar un SMS se tiene que generar un código compuesto por las mismas instrucciones (comandos AT) que se envió en el apartado anterior, de manera que cuando se cargue el programa en la memoria de nuestro Arduino éste se encargue de transmitirle esos comandos en el mismo orden que se lo haría manualmente.

Como ya se explicó anteriormente lo primero que se debe hacer es verificar la colocación que deben tener los jumpers del módulo. En este caso se utiliza la opción GPRS_TX que será conectada a D2 (RX) y GPRS_RX a D3 (TX).

Los pasos a seguir son:

1. Generar el sketch con los comandos AT adecuados para que el módulo envíe un SMS. Se adjunta el código del programa:

```
int led=13;           //pin correspondiente al led
int onModulePin = 2; //pin correspondiente al boton de encendido
                    //para no tener que encenderlo manualmente
int timesToSend = 1; //cuantos mensajes queremos enviar
int count = 0;

void switchModule()
{
  //equivale a presionar 2 segundos el boton de encendido
  digitalWrite(onModulePin,HIGH);
  delay(2000);
  digitalWrite(onModulePin,LOW);
}

void setup(){
  Serial.begin(19200); //baudrate
  //delay(2000);
  pinMode(led, OUTPUT);
  pinMode(onModulePin, OUTPUT);
  switchModule(); //para encender el modulo automaticamente
  delay(20000);
  for (int i=0; i < 5 ;i++){
    //tiempo de espera para asegurarnos de tener conexion a la red
    //hacemos parpadear el led para saber que esta en espera
    digitalWrite(led,HIGH);
    delay(1500);
    digitalWrite(led,LOW);
    delay(1500);
  }
  Serial.println();
  Serial.println("AT+CMGF=1"); //seleccionamos el modo texto
  //Serial.println(Serial.read());
  delay(1000);
}

void loop() {
  while (count < timesToSend) {
    delay(1500);
    Serial.println("AT+CMGS=\"0999122987\"");
    delay(1000);
    Serial.print("PRUEBA DE ENVIO");
    delay(500);
    Serial.write(0x1A);
    delay(5000);
    count++;
    digitalWrite(led,HIGH);
  }
}
```

2. Se debe observar que los jumpers este bien colocados.
3. Se conecta el Arduino y el Shield con cables de puente, el puerto serial D2, D3 del Shield al puerto Serial0 del Arduino, el voltaje de 5V para el Shield desde Arduino y también gnd.
4. Se debe cargar el sketch en la memoria del Arduino y se lo desconecta del PC.
5. Se conecta la fuente externa que suministrara 9V tanto para el Arduino como para el Shield.

El programa se debe empezar a ejecutar y si todo está bien, tras unos segundos se deberá recibir el SMS en nuestro celular, tal y como se muestra en la figura



Figura 68: Recepción del SMS en el teléfono móvil

Lectura de SMS por comandos AT

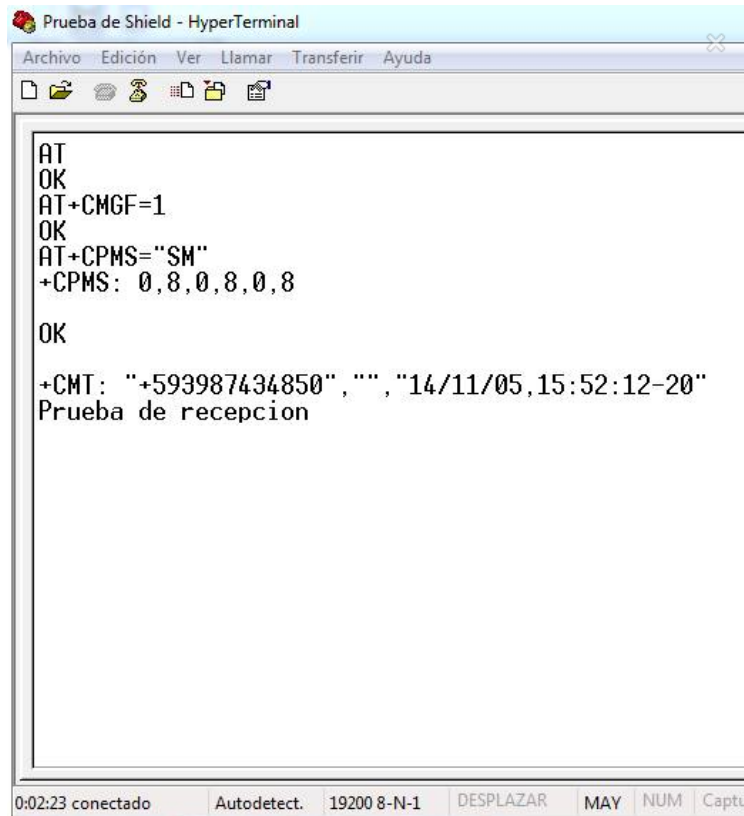
En esta parte se empieza enviando directamente los comandos AT a través del interfaz de Arduino, por lo tanto lo primero que se debe de hacer es colocar los jumpers de nuestro shield de modo correcto.

A continuación, se debe proceder a conectar la plataforma al puerto USB del ordenador, y alimentar también a través de la fuente de alimentación, de esta forma se puede estar seguro de que no se produzcan apagones del módulo.

Se debe abrir nuevamente el Hyper terminal, se le asigna un nombre, se configura el puerto y de esta manera ya se podrá interactuar con los comandos AT para la recepción de mensajes como se ve a continuación.

Los pasos que después se deben seguir para la lectura de un SMS desde la memoria de la tarjeta SIM se detallan a continuación. En la figura se muestran los resultados.

1. Se enciende manualmente el shield. Para ello se debe mantener pulsado el botón de encendido durante 2-3 segundos.
2. Enviar como de costumbre un simple comando "AT" para ver si el módulo responde correctamente y atiende a los datos que se le envían por el puerto serie. Si es así, responderá con un "OK".
3. A continuación se selecciona el formato del mensaje a través del comando "AT+CMGF=1", indicando que se trata de manejar SMS en modo texto. Si ha interpretado el comando correctamente nos responderá de nuevo con un "OK".
4. Este paso es quizá el más importante, ya que se debe indicar la memoria de la que se pretende leer el SMS en cuestión, en este caso la de la tarjeta SIM (siglas "SM"). Para ello debemos incluir la siguiente instrucción "AT+CPMS="SM"". La respuesta por parte del módulo debe ser "+CPMS: <x>, <y>", donde <x> indica el número de SMS almacenados en la memoria, e <y> hace referencia al número máximo de mensajes que se pueden llegar a ella. Como vemos en la figura 71, en un principio no se muestra ningún mensaje, pero nuestra SIM es capaz de almacenar hasta 8 mensajes de texto (0, 8).



```
AT
OK
AT+CMGF=1
OK
AT+CPMS="SM"
+CPMS: 0,8,0,8,0,8

OK

+CMT: "+593987434850", "", "14/11/05,15:52:12-20"
Prueba de recepcion
```

Figura 69: Ejecución de comandos AT

Hasta esta instancia todo ha funcionado correctamente, y se ha conseguido con éxito llevar a cabo las pruebas de envío y lectura de SMS a través de comandos AT.

Lectura de SMS a través de Arduino

Para este ejercicio se enviará un mensaje de texto desde un teléfono móvil cualquiera para que el Arduino conectado al shield sea capaz de interpretar la instrucción enviada y ejecutarla, en este caso se encenderá 4 leds con diferentes instrucciones.

Posteriormente se debe proceder a realizar el código correspondiente al sketch para la lectura de mensajes y ejecución de la misma. Para verlo en detalle, se irá comentando brevemente cada una de las funciones de las que se compone el código.

Lo primero que se debe realizar es la declaración de variables y la función 'setup'. Esta función es la primera en ejecutarse tras el arranque del programa. Se debe establecer el valor de baudrate (tasa de transmisión). A continuación, se enciende el módulo de manera automática, evitando que se tenga que hacerlo manualmente como ocurre cuando se intercambia comandos AT. Para esto se llama a la función 'SIM900power'.

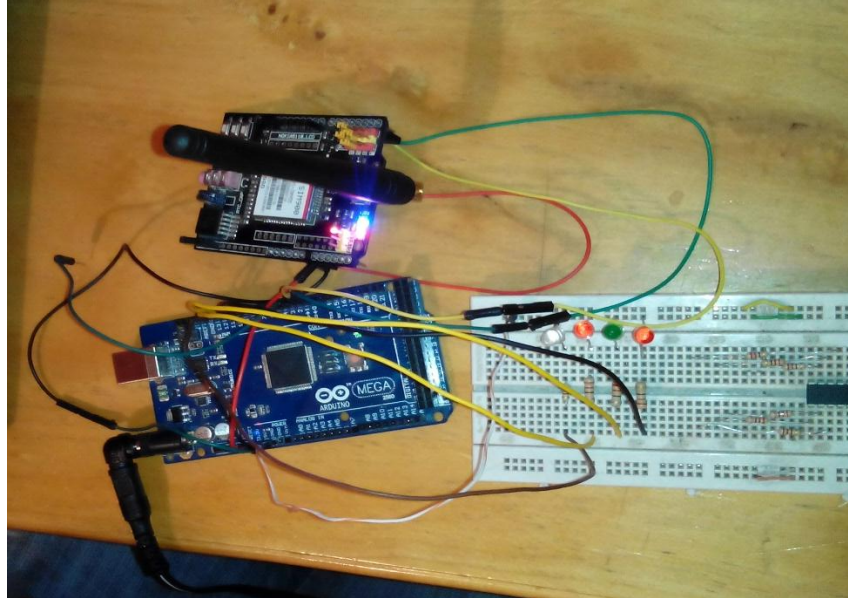


Figura 70: Montaje del circuito para encender 4 leds

```
// Ejemplo prender leds

char inchar; // Almacenara las instrucciones enviadas
int onModulePin = 2;

int led1 = 10;
int led2 = 11;
int led3 = 12;
int led4 = 13;

void SIM900power()
// Equivale a tener presionado el boton de encendido durante 2 segundos
{
    digitalWrite(onModulePin, HIGH);
    delay(2000);
    digitalWrite(onModulePin, LOW);
}
void setup()
{
    Serial.begin(19200);
    // Establecemos los pines digitales para el control
    pinMode(led1, OUTPUT);
    pinMode(led2, OUTPUT);
    pinMode(led3, OUTPUT);
    pinMode(led4, OUTPUT);
    pinMode(onModulePin, OUTPUT);
    digitalWrite(led1, LOW);
    digitalWrite(led2, LOW);
```

```
digitalWrite(led3, LOW);
digitalWrite(led4, LOW);

// Llamamos a la funcion que enciende el shield
SIM900power();
delay(20000); // tiempo para que se enganche a la red
Serial.print("AT+CMGF=1\r"); // establece el modo texto para los sms
delay(100);
Serial.print("AT+CNMI=2,2,0,0,0\r");
// contenedor de nuevos sms
delay(100);
Serial.println("Ready...");
}

void loop()
{
  //ejecucion si coincide con las opciones
  if(Serial.available() >0)
  {
    inchar=Serial.read();
    if (inchar=='#')
    {
      delay(10);

      inchar=Serial.read();
      if (inchar=='a')

      {
        delay(10);
        inchar=Serial.read();
        if (inchar=='0')
        {
          digitalWrite(led1, LOW);
        }
        else if (inchar=='1')
        {
          digitalWrite(led1, HIGH);
        }
        delay(10);
        inchar=Serial.read();
        if (inchar=='b')
        {
          inchar=Serial.read();
          if (inchar=='0')
          {
            digitalWrite(led2, LOW);
          }
          else if (inchar=='1')
          {
            digitalWrite(led2, HIGH);
          }
          delay(10);
          inchar=Serial.read();
          if (inchar=='c')
```

```
{
  inchar=Serial.read();
  if (inchar=='0')
  {
    digitalWrite(led3, LOW);
  }
  else if (inchar=='1')
  {
    digitalWrite(led3, HIGH);
  }
  delay(10);
  inchar=Serial.read();
  if (inchar=='d')
  {
    delay(10);
    inchar=Serial.read();
    if (inchar=='0')
    {
      digitalWrite(led4, LOW);
    }
    else if (inchar=='1')
    {
      digitalWrite(led4, HIGH);
    }
    delay(10);
  }
}
Serial.println("AT+CMGD=1,4"); // borra todos los SMS
}
}
}
}
```

A continuación se detalla los pasos a seguir para conseguir leer un SMS a partir de un programa cargado previamente en la memoria de nuestro Arduino.

1. En primer lugar se procede a cargar el programa. Para ello, como ya se ha comentado otras veces en apartados anteriores, se debe conectar la plataforma compuesta por Arduino y el shield GPRS a nuestro PC.
2. Una vez que el código ha sido cargado correctamente, se alimenta también la plataforma con la fuente de alimentación externa, con 9V para que pueda engancharse a la red.

3. Ahora se tiene que esperar a los resultados correspondientes al desarrollo del programa, y por consiguiente, el contenido del SMS que se ha leído, y que la orden sea ejecutada.

Hasta aquí se puede dar por finalizadas las pruebas referentes al comportamiento del módulo GPRS/GSM, y se deberá ya pasar a la integración de las distintas funciones en el programa correspondiente a la aplicación general para la medición de la presión arterial.

Desarrollo de Aplicaciones y Evaluación del Sistema

Por fin se entrará en la fase final del proyecto, donde se llevará a cabo el desarrollo de las aplicaciones para la implementación del sistema.

En primer lugar se procederá a diseñar una aplicación que sea capaz de integrar las distintas funcionalidades evaluadas hasta el momento (medidor de presión arterial, Shield EFcom SIM900, Arduino, relés, envío de mensajes).

Los datos que salen desde el puerto serial que envía el Omron medidor de presión arterial, vienen en formato hexadecimal y mediante un código de transformación realizado en IDE de Arduino se obtendrá datos decimales, los mismos que en primera instancia serán enviados a la red GSM manipulando manualmente el encendido del tensiómetro.

La información que se obtendrá será: presión sistólica, diastólica y pulso útiles para la toma de decisiones en el campo de la medicina.

En segundo lugar se realizará otras pruebas para la medición de la presión arterial, pero esta vez la gestión se hará de forma remota, a través del envío y recepción de SMS ya sin manipular manualmente el encendido, el encargado de controlar el sistema lo podrá hacer desde cualquier parte.

Tras el desarrollo de cada una de las aplicaciones se procederá a realizar la respectiva evaluación, con el objetivo de poder exponer los resultados con detalle en los apartados correspondientes.

Código de la aplicación general para el sistema de control de presión arterial

Se empieza con el desarrollo del programa correspondiente a la aplicación general. Se tendrá la opción para poder accionar el tensiómetro manualmente y remotamente con un SMS.

El material del que se dispone para la implementación del sistema se compone de:

- Arduino MEGA
- Shield GPRS/GSM (SIM900)
- Medidor de presión arterial Omron
- 1 Relé
- 2 Transistores
- 4 Resistencias
- 1 Diodo Zener
- Una fuente regulable a 9V
- Cables para puente

Recordando algunas de las justificaciones de que porque elegimos Arduino Mega, en primer lugar, la memoria de programa del modelo UNO (32 KB) es bastante limitada como para permitir el desarrollo de una aplicación en la que ya se integre varias funcionalidades, Arduino MEGA en cambio, dispone de hasta 256 KB para memoria de programa, con lo que se tiene un margen mucho mayor a la hora de generar un sketch más completo.

Otra de las razones es que con Arduino MEGA se provee al sistema de 50 pines adicionales en comparación de los que ofrecía Arduino UNO, facilitando la conexión de los diferentes elementos y ofreciendo la posibilidad de escalar las prestaciones del proyecto en un futuro, y la principal que en este proyecto lo que se necesita es que tenga más de un puerto serial para la comunicación con el Omron y el Shield, tomando

en cuenta que Arduino Uno tiene un solo puerto serial mientras tanto que Arduino mega dispone de 4 puertos seriales lo que permite trabajar con mayor facilidad.

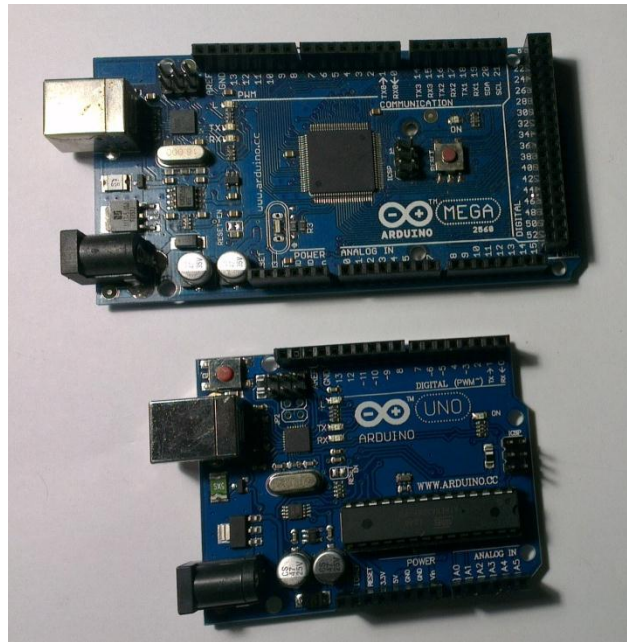


Figura 71: Arduino Uno y Mega

UNO		MEGA	
Microcontroller	ATmega328	Microcontroller	ATmega2560
Operating Voltage	5V	Operating Voltage	5V
Input Voltage (recommended)	7-12V	Input Voltage (recommended)	7-12V
Input Voltage (limits)	6-20V	Input Voltage (limits)	6-20V
Digital I/O Pins	14 (of which 6 provide PWM output)	Digital I/O Pins	54 (of which 15 provide PWM output)
Analog Input Pins	6	Analog Input Pins	16
DC Current per I/O Pin	40 mA	DC Current per I/O Pin	40 mA
DC Current for 3.3V Pin	50 mA	DC Current for 3.3V Pin	50 mA
Flash Memory	32 KB (ATmega328) of which 0.5 KB used by bootloader	Flash Memory	256 KB of which 8 KB used by bootloader
SRAM	2 KB (ATmega328)	SRAM	8 KB
EEPROM	1 KB (ATmega328)	EEPROM	4 KB
Clock Speed	16 MHz	Clock Speed	16 MHz

Tabla IV: Comparacion entre Arduino Uno y Mega

A continuación se irá fragmentando el código general de la aplicación mientras también se comenta las distintas funciones que lo componen. El resultado del sketch completo se llegará a obtener de la unión de cada una de las partes que se van a ir argumentando.

Se debe empezar por la declaración de variables, y la función 'setup' con la que se inicia el programa. En la función 'setup' se establece el valor de 'baudrate' correspondiente a los puertos serie. En este caso, se necesita dos puertos para el intercambio de datos, uno dedicado a la comunicación entre el Omron y la plataforma Arduino ('Serial1'), y otro para la comunicación entre la placa Arduino y el módulo GPRS ('Serial0').

Dentro del 'setup' se tendrá que configurar también como salidas los pines asociados a los diferentes actuadores (Omron, botón de encendido Shield GSM).

Es importante y prioritario encender el Shield GPRS/GSM al comienzo del programa, proceso que se menciona en el código como encendido del módulo GSM.

```
#include <string.h>
#include <Stream.h>

// DECLARACION DE VARIABLES PARA CALCULO DE LA PRESION
int i=0; int J=0;          // Contadores
int a; int b;             // Sistolica
float c; float S;         // Sistolica
int a1; int b1;           // Diastolica
float c1; float D;        // Diastolica
int a2; int b2; float P;  // Pulso

//DECLARACIONES DE PINES
int ONOFFOMRON =11; // pin correspondiente al boton del Omron
int PWR=10;         // pin correspondiente al encendido del Shield
//----
char var[255]; // variable que almacena los datos que vienen del Omron
char var1[255]; // variable que almacena los datos que viene de un dispositivo GSM
```

```
void setup() {  
  
  pinMode(ONOFFOMRON,OUTPUT);  
  pinMode(PWR,OUTPUT);  
  Serial.begin(19200); // baudrate del puerto serie0 con Shield GSM  
  Serial1.begin(9600); // baudrate del puerto serial con el Omron  
  
  delay(3000);  
  // encendido del modulo GSM  
  digitalWrite(PWR,HIGH);  
  delay(1000);  
  digitalWrite(PWR,LOW);  
  delay(20000);  
  
}
```

Cuando ya se tiene todo correctamente configurado y listo para el inicio del programa, se deberá tener muy en cuenta la función principal, la famosa void 'loop', que es obligatoria en cada sketch de Arduino. Cada vez que se cumpla cierta condición se ejecutará la parte del código correspondiente y tras llegar al final del proceso volverá a entrar en el 'loop', a la espera de nuevos datos.

En esta parte del código se define además el formato de los SMS modo texto para nuestra aplicación y se comunica al módulo que esté listo para recibir nuevos mensajes. Posteriormente mediante el puerto serial0 se detecta la letra 'P' y enciende el dispositivo medidor de presión arterial.

```
void loop() {

//-----Dejamos listo el modulo GSM para la recepcion de SMS
if (J==0){
    Serial.println("AT+CMGF=1");
    delay(2000);
    Serial.println("AT+CNMI=1,2,0,0,0");
    delay(2000);
    J=1;
}

//---recepcion celular //SE LEE 54 DATOS CON LA P
if (Serial.available()>0){

    char d= Serial.read(); // variable que almacena datos
                          // que llegan al Shield

    if(d=='P'){
        digitalWrite(ONOFFOMRON,HIGH); // encendido del Omron cuando
                                      // detecta la letra P con un SMS
        delay(2000);                  // equivale a presionar 2 segundos
                                      // pulsador del Omron
        digitalWrite(ONOFFOMRON,LOW); // y apagamos
    }

}

//---final recepcion celular
```

A continuación se presenta el código para la recepción de datos desde el Omron por el puerto serie1 del Arduino, datos que vienen en formato hexadecimal que son transformados a datos decimales tanto de la presión sistólica, diastólica y pulso. Como se explicó anteriormente solo una parte de la trama tiene los datos que se necesita, es por aquello que después de leer la parte buscada se presentara la letra 'W' y después de transformar todos los datos enviara a la red GSM.

```
//*****Envio lectura omron
if (Serial1.available()>0){

    char c= Serial1.read();
    var[i]=c;    // almacena datos que viene del ormon

    if(c=='W'){ // condicion para que se envie datos
                // medidos a la red GSM
//Calculo Sistolica
    // Sistolica 1
        if((var[28]>=65)&&(var[28]<=70) )
        {
            a=(var[28]-55)*16;
        }else{
            a=(var[28]-48)*16;
        }

    // Sistolica 2
        if((var[29]>=65)&&(var[29]<=70) )
        {
            b=(var[29]-55);
        }else{
            b=(var[29]-48);
        }

    c=(var[30]-48)*0.0625;
    S=(a+b+c)*2;
//final sistolica
```

A continuación se presenta un código más reducido que el presentado anteriormente que hace la misma función pero menos sentencias del condicional if.

Esta parte mostraremos la transformación de hexadecimal a decimal de la medida sistólica de la presión arterial.

```
// Calculo Sistolica
// Sistolica 1
    if((var[i-12]>=65)&&(var[i-12]<=70) )
    {
        a=(var[i-12]-55)*16;
    }else{
        a=(var[i-12]-48)*16;
    }

// Sistolica 2
    if((var[i-11]>=65)&&(var[i-11]<=70) )
    {
        b=(var[i-11]-55);
    }else{
        b=(var[i-11]-48);
    }

    c=(var[i-10]-48)*0.0625;
    S=(a+b+c)*2;
// final sistolica
```

A continuación se presenta el código que nos ayudara con la transformación de hexadecimal a decimal de la medida diastólica de la presión arterial.

```
// calculo Diastolica

//Diastolical
    if((var[i-8]>=65)&&(var[i-8]<=70) )
    {
        al=(var[i-8]-55)*16;
    }else{
        al=(var[i-8]-48)*16;
    }

//Diastolica 2
    if((var[i-7]>=65)&&(var[i-7]<=70) )
    {
        bl=(var[i-7]-55);
    }else{
        bl=(var[i-7]-48);
    }

    cl=(var[i-6]-48)*0.0625;
    D=(al+bl+cl)*2;

//final diastolica
```

A continuación se presenta el código que nos ayudara con la trasformación de hexadecimal a decimal de la medida del pulso algo adicional del proyecto.

```
//calculo pulso
//pulso 1
if((var[i-4]>=65)&&(var[i-4]<=70) )
{
    a2=(var[i-4]-55)*16;
}else{
    a2=(var[i-4]-48)*16;
}

//pulso 2
if((var[i-3]>=65)&&(var[i-3]<=70) )
{
    b2=(var[i-3]-55);
}else{
    b2=(var[i-3]-48);
}

P=(a2+b2);
//final pulso
```

Ya con los datos listos es tiempo de enviarlos mediante mensajes de texto a un destinatario establecido, para lo cual se usa comandos AT como se muestra en la siguiente parte del código. Primero hay que asegurarse de que las medidas que presentan errores den aviso de las mismas mediante SMS a dos destinatarios (medico, familiar cercano) como se ve a continuación

```
// Condicion cuando se produce un error en la medicion
// o no esta siendo usado
if ( S<=0 || D<=0 || P<=0)
{
    //-----Se configura el primer numero de envio
    Serial.println("AT"); // vemos si el modulo GSM responde
    delay(1000);
    Serial.println("AT+CMGF=1"); // establecemos en modo texto el sms
    delay(1000);
    Serial.print("AT+CMGS=\"0981767097\""); // fijamos destinatario
    Serial.println();
    delay(1000);
    Serial.println("ERROR"); // mensaje a enviar
    delay(200);
    Serial.println("MAL COLOCADO O NO ESTA EN USO"); // mensaje a enviar
    delay(200);
    Serial.println((char)26); // se confirma el envio del sms
    Serial.println();
    delay(3000);
}
```

En la siguiente parte del código se muestra la configuración de aviso de error para el segundo número.

```
//-----Se configura el segundo numero de envio
Serial.println("AT"); // vemos si el modulo GSM responde
delay(1000);
Serial.println("AT+CMGF=1"); // establecemos en modo texto el sms
delay(1000);
Serial.print("AT+CMGS=\"0987434850\""); // fijamos destinatario
Serial.println();
delay(1000);
Serial.println("NO ESTA EN USO O ESTA MAL COLOCADO"); // mensaje a enviar
delay(200);
Serial.println((char)26); // se confirma el envio del sms
Serial.println();
delay(1000);
```

Cuando en la medición no haya errores se presenta la siguiente condición con valores de la presión arterial que mediante la red GSM llegará a dos destinatarios como se ve a continuación.

```
}else {

  if (S>0 && D>0 && P>0)
  {
    //-----Envio de datos primer destinatario
    Serial.println("AT"); // vemos si modulo GSM responde
    delay(1000);
    Serial.println("AT+CMGF=1"); // establecemos en modo texto el sms
    delay(1000);
    Serial.print("AT+CMGS=\"0981767097\""); // fijamos destinatario
    Serial.println();
    delay(1000);
    Serial.println("PRESION ARTERIAL"); // mensaje a enviar
    delay(200);
    Serial.print("Sistolica="); // mensaja a enviar
    delay(200);
    Serial.println(S); // mensaje con la medida sistolica
    delay(200);
    Serial.print("Diastolica="); // mensaje a enviar
    delay(200);
    Serial.println(D); // mensaje con la medida diastolica
    delay(200);
    Serial.print("Pulso="); // mensaja a enviar
    delay(200);
    Serial.println(P); // mensaje con la medida del pulso
    delay(200);
    Serial.println((char)26); // se confia el envio del sms
    Serial.println();
    delay(5000);
```

Configuración del segundo número con los datos de la medición de presión arterial.

```
//envio datos a celular 2
  Serial.println("AT");
  delay(1000);
  Serial.println("AT+CMGF=1");
  delay(1000);
  Serial.print("AT+CMGS=\"0987434850\"");
  Serial.println();
  delay(1000);
  Serial.println("PRESION ARTERIAL");
  delay(200);
  Serial.print("Sistolica=");
  delay(200);
  Serial.println(S);
  delay(200);
  Serial.print("Diastolica=");
  delay(200);
  Serial.println(D);
  delay(200);
  Serial.print("Pulso=");
  delay(200);
  Serial.println(P);
  delay(200);
  Serial.println((char)26);
  Serial.println();
  delay(1000);
```

En la siguiente parte del código mostramos el incremento del contador para poder reiniciar la variable que almacena datos y poder utilizarla para una nueva medición.

```
  //*****
  }
  }
  J=0;
  }
  i=i+1;
  if(i==51){

  i=0;
  }

  }
  //*****final de envio lectura omron a movil
  }
```


CAPÍTULO V

PRUEBAS Y RESULTADOS

Evaluación del sistema de gestión local

En este apartado se evaluará paso a paso el funcionamiento del programa correspondiente a la medición de la presión arterial cuando el paciente acciona el dispositivo y el que receptara esas mediciones será el doctor que lleva el tratamiento y un familiar cercano.

El paciente se colocará la manguita en forma correcta para evitar medidas erróneas, se prende el tensiómetro y se espera a que el proceso de medición culmine, entonces se puede ver las medidas de la presión arterial que se muestran en el display del Omron, de esta forma se envía al móvil programado, después el sistema se apagará automáticamente y estará listo para la siguiente medición.

Es así que luego de haber ya implementado el sistema de comunicación omrom .- celular, se procedió a realizar mediciones a diferentes personas con el propósito de comprobar que los resultados de la medición que nos arroja el omrom sean los mismos que llegan al celular vinculado para obtener dichos datos.

Los resultados que se obtuvieron fueron los siguientes.

N ^a	Nombre	Cedula	Datos Omrom	Datos que llegan al celular	Datos correctos	Datos Incorrectos
1	Gabriela Peñafiel	060452398-5	Sistólica: 110 Diastólica: 62 Pulso: 55	Sistólica: 110 Diastólica: 62 Pulso: 55	✓	
2	Sergio Guaraca	060228081-0	Sistólica: 120 Diastólica: 71 Pulso: 62	Sistólica: 120 Diastólica: 71 Pulso: 62	✓	
3	Fabian Uquillas	060343596-7	Sistólica: 118 Diastólica: 76 Pulso: 71	Sistólica: 118 Diastólica: 76 Pulso: 71	✓	
4	Jorge Ocaña	060368368-1	Sistólica: 112 Diastólica: 68 Pulso: 56	Sistólica: 112 Diastólica: 68 Pulso: 56	✓	
5	Cesar Maigua	060410152-7	Sistólica: 116 Diastólica: 71 Pulso: 64	Sistólica: 116 Diastólica: 71 Pulso: 64	✓	
6	María broncano	060281445-1	Sistólica: 119 Diastólica: 69 Pulso: 56	Sistólica: 119 Diastólica: 69 Pulso: 56	✓	
7	Jhony Vasco	060345527-0	Sistólica: 114 Diastólica: 79 Pulso: 72	Sistólica: 114 Diastólica: 79 Pulso: 72	✓	
8	Laura Ortiz	060257760-3	Sistólica: 112 Diastólica: 71 Pulso: 68	Sistólica: 112 Diastólica: 71 Pulso: 68	✓	
9	Juan Marquez	060335126-3	Sistólica: 118 Diastólica: 69 Pulso: 58	Sistólica: 118 Diastólica: 69 Pulso: 58	✓	
10	Marlene Morocho	060136504-2	Sistólica: 115 Diastólica: 73	Sistólica: 115 Diastólica: 73	✓	

			Pulso: 60	Pulso: 60		
11	Abraham Paguay	060327902-7	Sistólica: 118 Diastólica: 77 Pulso: 56	Sistólica: 118 Diastólica: 77 Pulso: 56	✓	
12	Wilson Ordoñez	060313745-6	Sistólica: 120 Diastólica: 75 Pulso: 61	Sistólica: 120 Diastólica: 75 Pulso: 61	✓	
13	Luz Montesdeoca	171487880-6	Sistólica: 116 Diastólica: 79 Pulso: 59	Sistólica: 116 Diastólica: 79 Pulso: 59	✓	
14	Zoila Miranda	060317287-5	Sistólica: 121 Diastólica: 78 Pulso: 52	Sistólica: 121 Diastólica: 78 Pulso: 52	✓	
15	Patricia Rosero	060414071-5	Sistólica: 112 Diastólica: 73 Pulso: 61	Sistólica: 112 Diastólica: 73 Pulso: 61	✓	
16	María Toaquiza	060467034-9	Sistólica: 110 Diastólica: 68 Pulso: 65	Sistólica: 110 Diastólica: 68 Pulso: 65	✓	
17	Ximena Tobando	180294076-5	Sistólica: 116 Diastólica: 75 Pulso: 60	Sistólica: 116 Diastólica: 75 Pulso: 60	✓	
18	Boris Camacho	210092786-8	Sistólica: 111 Diastólica: 70 Pulso: 66	Sistólica: 111 Diastólica: 70 Pulso: 66	✓	
19	Carmen Cruz	060361511-3	Sistólica: 125 Diastólica: 79 Pulso: 59	Sistólica: 125 Diastólica: 79 Pulso: 59	✓	
20	Irma Concha	060302028-0	Sistólica: 114 Diastólica: 72 Pulso: 56	Sistólica: 114 Diastólica: 72 Pulso: 56	✓	

Tabla V: Resultados de mediciones realizadas a diferentes personas con el sistema implementado

Al finalizar con las mediciones de la presión sistólica, diastólica y pulso que nos permite obtener el omrom, se logró obtener de 20 mediciones realizadas a diferentes

personas los datos correctos los mismos que llegaron al celular asignado para la recepción de datos sin ninguna alteración tal y como se mostraba en la pantalla del omrom.

Comparación del sistema de comunicación omrom – celular con un medidor de presión manual

El propósito de esta comparación es saber el tiempo que se demora en mostrar los datos de las mediciones de la presión arterial sistólico, diastólico y el pulso tanto del sistema implementado como del medidor manual obteniendo los siguientes resultados:

Tipo de Medición	Tiempo Medidor Manual	Tiempo Sistema Implementado
Presión diastólica	1 min	0,35 seg
Presión Sistólica	1min	
Pulso	1 min	

Tabla VI: Comparación del sistema implementado con un medidor de presión manual

Al realizar varias pruebas el tiempo en obtenerse los datos con el medidor de presión manual se tuvo un tiempo estimado de 1minuto tanto para la presión sistólica y diastólica, añadiendo además un minuto extra que se demora en obtener el valor del pulso dando como resultado un tiempo aproximado de 3minutos en obtener los datos, mientras que en el sistema implementado todos los datos de la medición llegan al celular de recepción en un tiempo aproximado de 0,35 segundos.

Comprobación de Hipótesis

La hipótesis que se planteó al inicio del proyecto investigativo cita “El diseño e implementación de un sistema de comunicación omrom - celular brindará confiabilidad, seguridad y eficiencia en la recepción de datos”

Al realizar las pruebas pertinentes se obtuvo como resultado que el sistema de comunicación diseñado para medir la presión arterial es capaz de traducir los datos que se obtienen del omrom, enviarlos a la red gsm para lograr llegar al receptor que es el teléfono celular sin ningún error, de esta forma los datos pueden llegar a su

destino mostrando los mismos valores de medición que se obtuvieron en el omrom antes de pasar por la red gsm.

Otra de las pruebas que se realizó con el sistema implementado de medición de presión arterial, permitió comprobar la rapidez que tiene nuestro sistema en mostrar los valores de medición comparado con un medidor de presión manual que se demora más tiempo en dar los mismo valores.

Por lo tanto “DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DE COMUNICACIÓN OMROM – CELULAR PARA MEDIR LA PRESIÓN ARTERIAL” es confiable y seguro porque cuando se envía los datos llegan al receptor exactamente igual a los datos que se obtuvieron en la medición en el omrom al inicio, tomando en cuenta que la red celular nos ayuda brindando seguridad a los datos para ser transportados a su destino, y además es eficiente ya que permite obtener los mismos datos que nos daría un medidor de presión manual pero utilizando menos recursos que en el proyecto es en menos tiempo y logrando un envío de datos a lugares remotos.

CONCLUSIONES

- ✓ Al diseñar e implementar el sistema de comunicación para medir la presión arterial se utilizó el módulo shield gsm SIM 900 y la placa arduino mega 2560 para poder enviar los datos a través de un sms obtenidos del medidor de presión omron a un dispositivo móvil.
- ✓ El sistema de comunicación omron celular facilita el envío de datos de la presión arterial desde lugares remotos a través de la red gsm.
- ✓ El sistema de comunicación omron – celular es más eficiente que un medidor de presión manual ya que nos permite obtener los datos de la medición en menor tiempo.
- ✓ Como el sistema de comunicación está trabajando bajo la red gsm, es factible activarlo desde cualquier región del país donde exista cobertura GSM.
- ✓ Tomando en cuenta que la plataforma de programación de arduino es de acceso libre las librerías utilizadas para la construcción del sistema pueden ser descargadas directamente del internet.
- ✓ El tiempo estimado en que tarda en llegar un SMS con los datos de la medición obtenidos en el omron al celular es de aproximadamente 5 segundos que es considerado un retardo aceptable para nuestro sistema.

RECOMENDACIONES

- Se recomienda al momento de realizarse mediciones de la presión colocarse de forma correcta la manga para evitar obtener como resultado de la medición un mensaje de error.
- Se recomienda tener en cuenta los valores de los voltajes necesarios para que el sistema pueda funcionar correctamente y el shiled gsm SIM 900 pueda acceder a la red.
- Capacitar a los usuarios sobre la utilización y funcionamiento del sistema para que no existan errores en la medición
- No se debe olvidar que el sistema funciona a través de mensajes por lo que se recomienda consultar el saldo de mensajes del chip que se encuentra en el módulo para lograr que la activación del sistema sea exitosa y a la vez el envío de la información pueda llegar a su receptor.
- Consultar las hojas de datos técnicos de los dispositivos electrónicos a utilizar para poder darles un uso adecuado evitando posibles daños a los mismos.
- Es preferible usar un chip de la operadora claro ya que tiene mayor cobertura de red a nivel del país.

RESÚMEN

Diseñar e implementar un sistema de comunicación que permite medir la presión arterial y el pulso cuyos datos son enviados mediante un sms de texto a un teléfono celular, tiene como propósito informar de forma veraz y oportuna al paciente que padece esta enfermedad permitiéndole prevenir una complicación mayor y además permitir que el doctor lleve un control de sus mediciones periódicas.

El sistema consta de un medidor de presión arterial omron, una placa arduino mega 2560 el cual contiene un programa que permite el funcionamiento del sistema y el envío de los datos a un dispositivo móvil mediante un sms, un shield GSM SIM900 que utiliza un chip de cualquier operadora móvil, un circuito de relé y un cargador regulable a 9 V necesario para encender el sistema.

Las técnicas utilizadas fueron la observación y testing, con ayuda de una investigación científica no experimental de campo necesario para evaluar el desempeño del sistema, implementar el prototipo acoplado el medidor de presión al modem arduino y comprobar los resultados obtenidos.

Con la implementación del sistema se logró cumplir con el objetivo ya que los valores de las mediciones de la presión arterial y el pulso llegan al celular mediante la red GSM en tiempo real y sin alteración alguna.

Palabras Clave: /OMRON//ARDUINO//SHIELD//GSM//SIM 900/

SUMMARY

Design and implement a communication system that allows to measure the blood pressure and pulse whose data are sending by a text or cell phone message, has as purpose to report the patient illness in a verified way to prevent further complication also allow the doctor keep track of the periodic measurements.

The system is about a meter Omron blood pressure and an Arduino Mega plate 2560 which contains a program that allows the operating system and sending the data to a mobile device by a text message, a Shield GSM SIM 900 which uses any mobile operator chip, a relay circuit and an 9V adjustable charger need to turn on the system.

The techniques applied were observation and testing, using a non-experimental scientific research necessary to evaluate the system performance, implement the prototype coupling the pressure gauge to Arduino modem and check the results.

With the system implementation was achieved the objective because the blood pressure values measurements and pulse get in the cell phone through the GSM network in real time without any change.

Clue Words: /OMRON//ARDUINO//SHIELD//GSM//SIM 900/

BIBLIOGRAFIA

1. **BARRETT, S.**, Arduino microcontroller processing for everyone., 3ra ed., s.l., Mitchell A., 2013., Pp. 150p.
2. **FERRER, O.**, Telemedicina., Madrid - España., Médica Panamericana., 2001., Pp. 1 – 16 – 54 – 55 – 71 – 92.
3. **KUZMAR, I.**, Como crear un servicio de telemedicina: guía para su implantación., 2da ed., Madrid - España., IBD., 2013., Pp. 15-49.
4. **LAPORTA, J., y MIRALLES, M.**, Fundamentos de Telemática., s.ed., Valencia – España., Universidad Politécnica de Valencia., s.f. Pp. 1 – 21 – 82 – 134.
5. **REYES, A.**, Microcontroladores programación en basic., 3ra ed., Quito - Ecuador., RISPGRAF., 2008., Pp. 25 – 31.
6. **SEDIN, A.**, Fundamentos de los sistemas de comunicaciones móviles., Madrid - España., Antonio Garcia., 2004., Pp. 335 – 363.
7. **TORRENTE, O.**, Arduino curso práctico de formación., s.ed., Madrid España., San Fernando de Heanares., 2013., Pp. 30 - 70

INTERNET

1. INFORMATICA Y SALUD ESPECIAL TELEMEDICINA P

<http://www.seis.es/documentos/revistas/revistacompleta/103.pdf>

2014-05-25

2. CLASIFICACIÓN DE LA TELEMEDICINA

<http://www.cenetec.salud.gob.mx/descargas/telemedicina/publicaciones/Volumen32daEdicion.pdf>

2014-05-26

3. CLASIFICACIÓN Y DEFINICIÓN DE LOS MÉTODOS DE MEDICIÓN DE LA PRESIÓN ARTERIAL

http://www.bibliotecadigitalcecova.com/contenido/revistas/cat7/pdf/manual_2.pdf

2014-06-05

4. DEFINICIÓN DE GSM

<http://186.42.96.211:8080/jspui/bitstream/123456789/1650/1/Sandoval%20Perugachi%20Juan%20Roberto.pdf>

2014-06-08

5. ARQUITECTURA DE RED GSM

<http://www.learobotics.com/personal/juan/doctorado/sms/sms.pdf>

2014-06-11

6. COMANDOS AT

http://elec Freaks.com/store/download/datasheet/rf/SIM900/SIM900_AT%20Command%20Manual_V1.03.pdf

2014-06-11

7. TECNOLOGÍA ARDUINO

http://www.tiendaderobotica.com/download/Libro_kit_Basico.pdf

2014-06-15

8. EFCOM/GPRS SHIELD

http://www.elec Freaks.com/store/download/product/EFcom/EFcom_Datasheet.pdf

2014-07-05

9. MANUAL DE PROGRAMACIÓN ARDUINO

<https://www.google.com.ec/url?sa=t&rct=j&q=&esrc=s&source=web&cd=1&cad=rja&6A&bvm=bv.79189006,d.eXY>

2014-07-14

10. DESCRIPCIÓN DE ARDUINO MEGA 2560

<http://arduino.cc/en/Main/arduinoBoardMega2560>

2014-07-21

11. SENSORES DE PRESIÓN

http://www.ele.uva.es/~lourdes/docencia/Master_Biosensores/Sensores_Presion.pdf

2014-08-02

12. OMRON INSTRUCTION MANUAL

<http://ecx.images-amazon.com/images/I/A1WZgdKS8US.pdf>

2014-08-03

13. TUTORIAL ARDUINO AND SIM900 GSM MODULES

<http://tronixstuff.com/2014/01/08/tutorial-arduino-and-sim900-gsm-modules/>

2014-08-15

14. EFCOM GPRS/GSM SHIELD

http://www.electronicsfor.com/wiki/index.php?title=EFCOM_GPRS/GSM_Shield

2014-09-01

ANEXOS

ANEXO 1

Manual de Usuario

ANEXO 2

Telemedicina

ANEXO 3

Medidor de Presión Arterial OMRON

ANEXO 4

Shield GSM

ANEXO 5

Arduino

ANEXO 6

Pruebas

ANEXO 1

Manual de Usuario

En el presente manual se da a conocer todos los procedimientos necesarios que se deben llevar a cabo para que el medidor de presión arterial funcione correctamente, siguiendo las recomendaciones se podrá manipular de forma correcta y segura, por esta razón es muy importante estudiar y conocer el manual de usuario.

GENERALIDADES DEL SISTEMA

El monitor de presión arterial omron acoplado para que funciona en conjunto con la red GSM utiliza el método oscilométrico de medición, significa que el monitor detecta el movimiento de la sangre mediante su arteria braquial y convierte los movimientos en una lectura digital. El sistema medidor de presión arterial inteligente tiene la capacidad de enviar la información procesada mediante mensajes de texto basado en las redes móviles de la tecnología GSM. Un monitor oscilométrico no necesita estetoscopio, por lo tanto, el monitor es fácil de usar.

El medidor de presión arterial inteligente incluye los siguientes componentes:

- Monitor
- Brazaletes estándar para brazo adulto
- Servicio de envío y recepción de mensajes
- Manual de usuario.

El Monitor de Presión Arterial inteligente está diseñado para ser usado en el hogar o lugares remotos en el que los médicos no pueden estar presentes.

SEGURIDADES

Es fundamental antes de empezar a utilizar el sistema seguir el instructivo de manejo y precaución. Teniendo en cuenta cómo y bajo qué parámetros y condiciones se utilizará el dispositivo, no olvidarse dispone de una ayuda en el control de la salud de las personas hipertensas y las que empiezan este tratamiento.

- Consulte a su médico para obtener información específica acerca de su presión arterial. El autodiagnóstico y la automedicación realizados sobre la base de los resultados de la medición pueden ser peligrosos. Siga las instrucciones de su médico o profesional autorizado.

- Este dispositivo está diseñado para medir la presión arterial y el pulso en adultos. Este dispositivo no debe utilizarse con bebés o con otras personas que no puedan expresar su consentimiento.

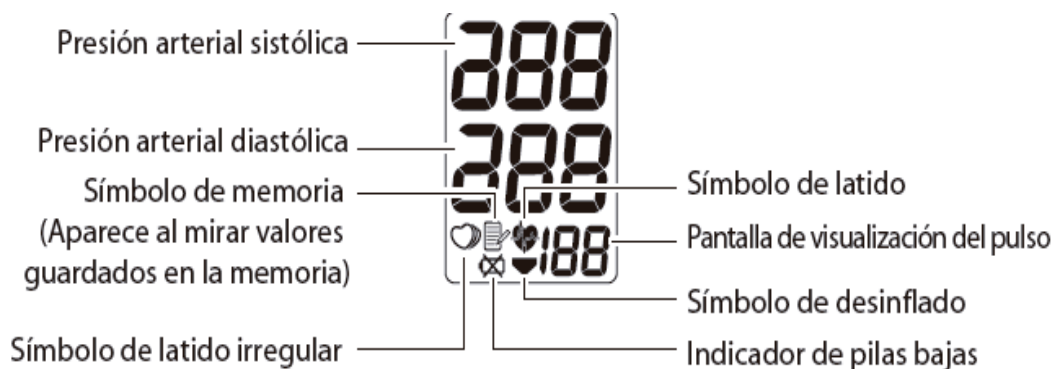
Antes de tomar una Medición

A fin de garantizar una lectura confiable, siga las recomendaciones que se enumeran a continuación:

- Evite comer, tomar bebidas alcohólicas, fumar, hacer ejercicio y bañarse durante 30 minutos antes de tomar una medición. Descanse durante al menos 15 minutos antes de tomar la medición.
- El estrés eleva la presión arterial. Evite tomar mediciones en momentos de estrés.
- Las mediciones se deben llevar a cabo en un lugar tranquilo.
- Quítese la vestimenta ajustada del brazo izquierdo.
- Siéntese en una silla con los pies apoyados sobre el piso. Coloque su brazo izquierdo en una mesa de modo que el brazalete quede al mismo nivel que su corazón.
- Quédese quieto y no hable durante la medición.
- Espere 2-3 minutos entre cada medición. El tiempo de espera permite que las arterias vuelvan a la condición en la que estaban antes de tomar la medición de la presión. Es posible que sea necesario aumentar el tiempo de espera según sus características fisiológicas particulares.
- Lleve un registro de las lecturas de presión arterial y del pulso para que las vea su médico. Una sola medición no brinda una indicación precisa de su verdadera presión arterial. Es necesario tomar varias mediciones y registrarlas durante un período de tiempo. Trate de medir su presión arterial todos los días a la misma hora para obtener medidas consistentes.

DESCRIPCION DEL SISTEMA

El sistema consta de tres dispositivos acoplados para funcionar en conjunto y con un solo pulsador ya sea accionado manualmente o de forma remotamente o a la vez mediante un mensaje de texto desde un teléfono móvil remoto.



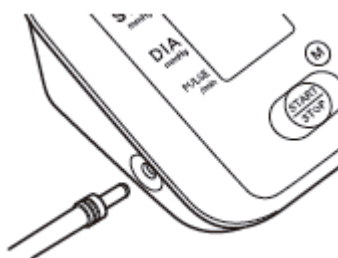
Pantalla del medidor de la presión Arterial Omron

FORMA MANUAL

La forma manual se utiliza accionando el botón de encendido, se espera el tiempo que sea necesario para obtener las mediciones, para ello se sigue los siguientes pasos:

Colocación del brazalete en el brazo izquierdo

1. Asegurarse de que el enchufe macho del aire esté correctamente colocado en la unidad principal.

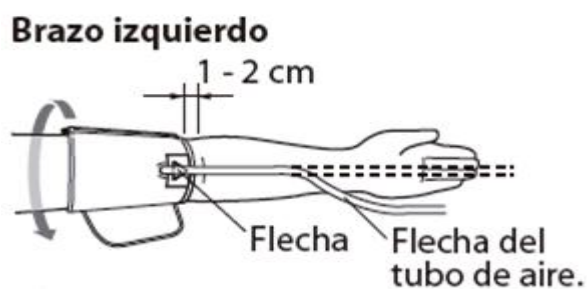


2. Retirar la vestimenta ajustada de la parte superior del brazo.



3. Colocar el brazalete en la parte superior del brazo izquierdo de modo que la flecha quede centrada en la parte interna del brazo y alineada con el dedo medio. El tubo de aire baja por la parte interna del brazo.

La parte inferior del brazalete debe quedar aproximadamente a 1/2" (1 - 2 cm) por encima del codo.



4. Colocar el brazalete de modo que quede firmemente envuelto alrededor del brazo usando la correa de cierre.



Colocación del brazalete en el brazo derecho

Cuando se toma una medición en el brazo derecho, se debe usar esta instrucción para el Paso 3. Colocar el brazalete en la parte superior del brazo derecho de modo que la flecha y el tubo de aire corran por la parte interna del brazo.

La parte inferior del brazalete debe quedar aproximadamente a 1/2" (1 - 2 cm) por encima del codo.

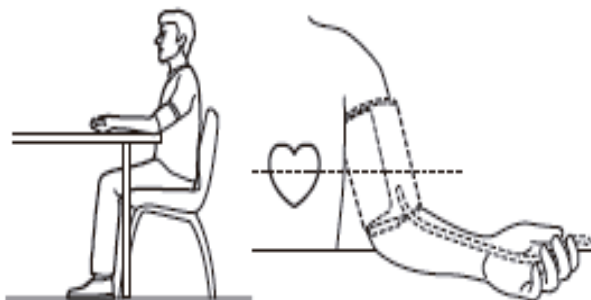
Brazo derecho



NOTA: Tener cuidado de no apoyar el brazo sobre el tubo de aire. Esto obstruiría el flujo de aire hacia el brazalete.

COMO TOMAR UNA MEDICIÓN

1. Sentarse en una silla con los pies apoyados sobre el piso.
Colocar el brazo en una mesa de modo que el brazalete quede en el mismo nivel que su corazón.
Quedarse quieto y no hablar durante la medición.

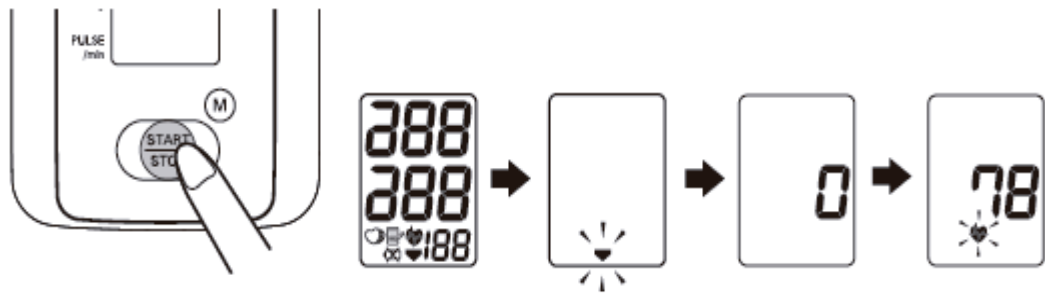


2. Presionar el botón START/STOP (Iniciar/Detener).

Todos los símbolos aparecen en la pantalla.

El brazalete comienza a inflarse automáticamente. A medida que el brazalete empieza a inflarse, el monitor determinará automáticamente el nivel óptimo de inflación.

Se debe quedar quieto y no mover el brazo hasta que se complete la medición.



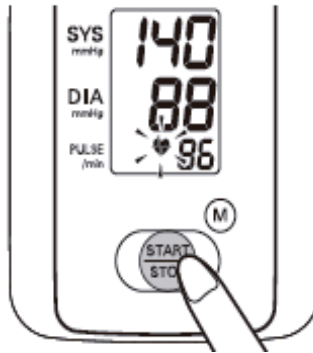
3. La inflación se detiene automáticamente y se inicia la medición.
 A medida que el brazalete se desinfla, aparecen números en la pantalla en orden decreciente y el símbolo de latido (🫀) parpadea.



4. Una vez que se haya completado la medición, el brazalete se desinfla por completo. Aparecen la presión arterial y el pulso.





5. Presionar el botón START/STOP (Iniciar/Detener) para apagar el monitor.



NOTA: El monitor se apagará automáticamente después de cinco minutos.

INDICADORES DE ERROR

SÍMBOLO	CAUSA	CORRECCIÓN
E	El monitor no pudo detectar el pulso.	Tomar otra medición y quedarse quieto hasta que se complete la medición.
EE	El brazalete no está lo suficientemente inflado. El brazalete no está colocado correctamente.	Quitar el brazalete del brazo. Lea "Como tomar una medición" Espere 2 o 3 minutos. Tomar otra medición.
EE	El brazalete está demasiado inflado.	
	La batería está agotada.	Cambiar la batería que alimenta el sistema con 9V.

	Error en el dispositivo	Reinicie el sistema pulsando el botón reset.
---	-------------------------	--

FORMA REMOTA

Siguiendo el manual instructivo de la forma de accionamiento manual y tomando las mismas recomendaciones, se explica que en la forma de accionamiento remota el sistema de medición arterial inteligente se acciona de forma automática y remota desde un teléfono móvil en cualquier parte, se debe tener en cuenta la letra clave y el número con el que está registrado el sistema, de esta manera con solo enviar un sencillo mensaje en este caso la letra “P” e indicando el destinatario, el número que este registrado en el sistema empezará a funcionar de la misma manera que en la forma manual.

CONSEJOS PARA LA SOLUCIÓN DE PROBLEMAS

PROBLEMA	CAUSAS Y SOLUCIONES
No hay alimentación eléctrica. No aparece ningún símbolo en la pantalla de la unidad.	Cambiar la batería por una nueva. Verificar si la fuente de alimentación está bien conectada.
Los valores de medición parecen ser demasiado altos o demasiado bajos.	La presión arterial varía constantemente. Muchos factores, incluyendo el estrés, la hora del día, la forma en que se coloca el brazalete, pueden afectar su presión arterial. Repasar las secciones “Antes de tomar una medición” y “Cómo tomar una medición”.
El sistema no enciende remotamente	Verificar el número del sistema. Verificar si el sistema contiene mensajes o saldo activo.

CUIDADO Y MANTENIMIENTO

- No someter el monitor a golpes fuertes, como dejar caer la unidad al suelo.
- No sumergir el dispositivo ni ninguno de sus componentes en el agua.
- Guardar el dispositivo y los componentes en un lugar limpio y seguro.
- Los cambios o las modificaciones que no hayan sido aprobados por el manual instructivo ocasionarán fallos en el sistema.
- No doblar el brazaletes ni el tubo de aire por la fuerza. No doblar los elementos de modo que queden demasiado apretados.
- Limpiar el monitor con un paño seco y suave.
- Utilizar la unidad de acuerdo con las instrucciones provistas en este manual.

RECOMENDACIONES

- Leer el manual de usuario para evitar mediciones erróneas y fallos en el sistema.
- Recargar periódicamente el sistema con mensajes o saldo para que pueda responder las peticiones remotas.
- Verificar que la antena este bien ajustada y colocada para que no haya problemas de conexión con la red.
- De preferencia utilizar la telefonía móvil de la operadora Claro.
- Contratar un paquete de mensajes establecido para reducir costos.

ANEXO 2

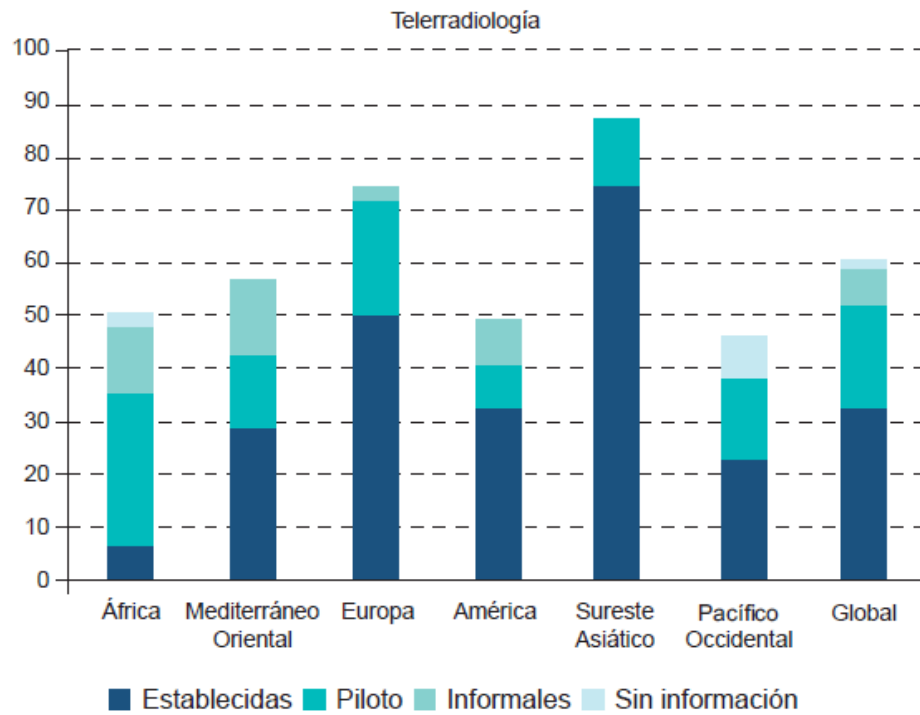
Telemedicina

Instituciones que han desarrollado actividades de telemedicina en el Ecuador

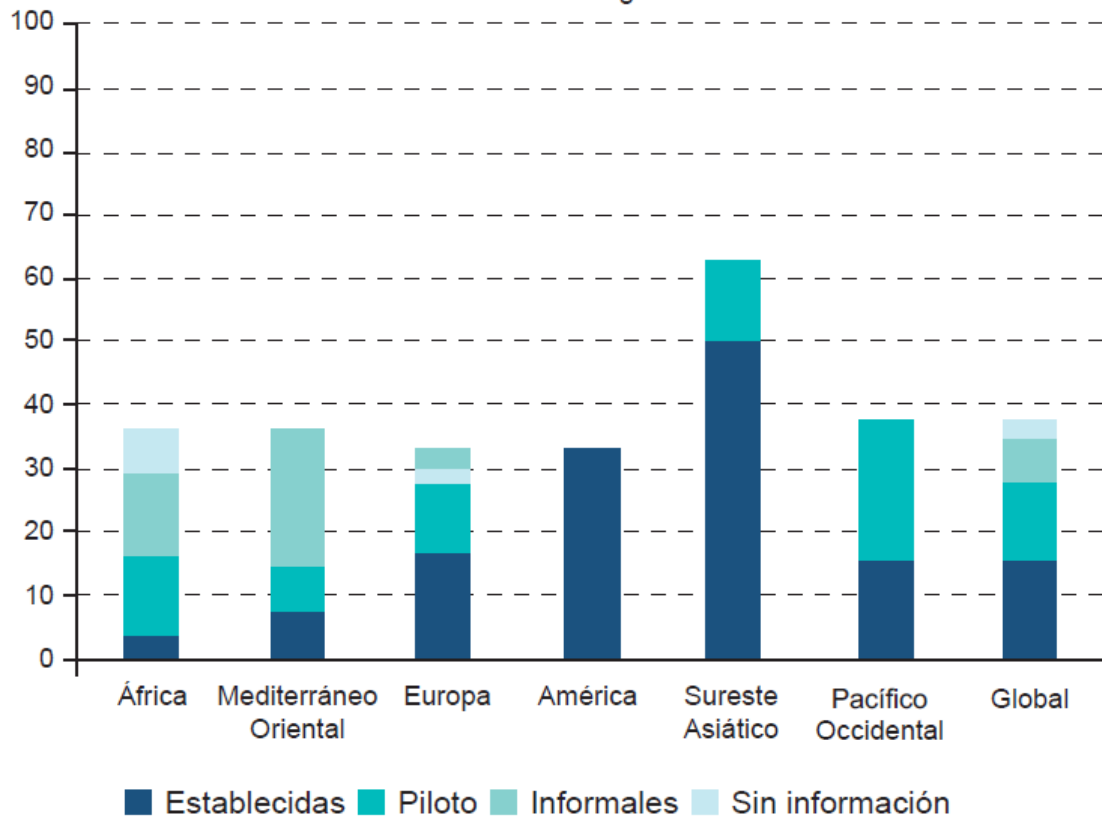
Ministerio de Salud Pública

- **Universidad Tecnológica Equinoccial (UTE),** 
- **Universidad del Azuay** 
- **Universidad San Francisco De Quito** 
- **Universidad Técnica Particular de Loja (UTPL)** 
- **Fuerza Aérea Ecuatoriana (FAE)** 
- **Escuela Politécnica del Litoral** 
- **Universidad Central del Ecuador**
– FACULTAD DE CIENCIAS MEDICAS (2011) 

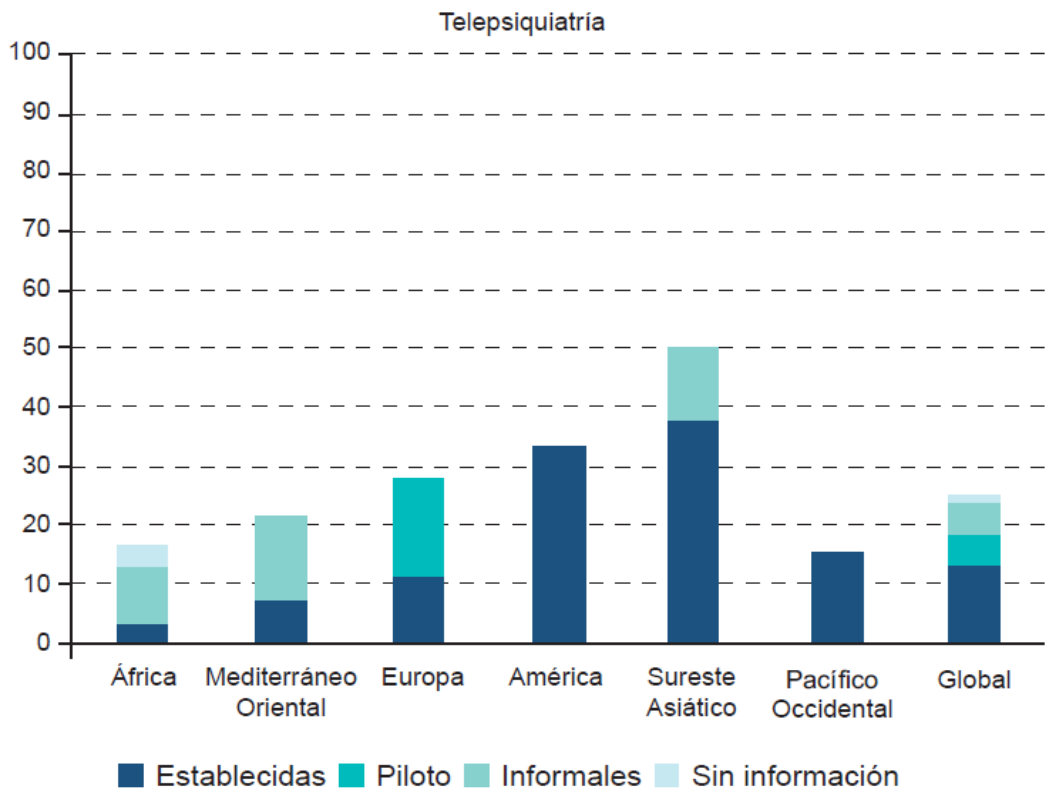
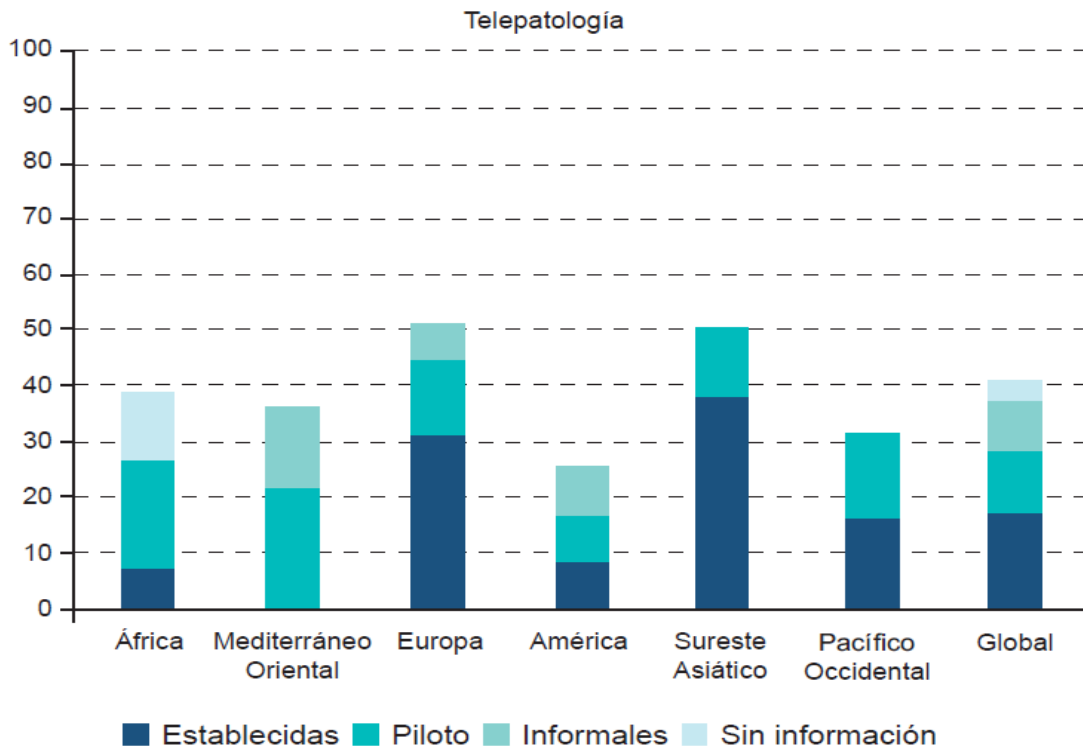
Iniciativas en Telerradiología y Tele dermatología por Región (En porcentajes)



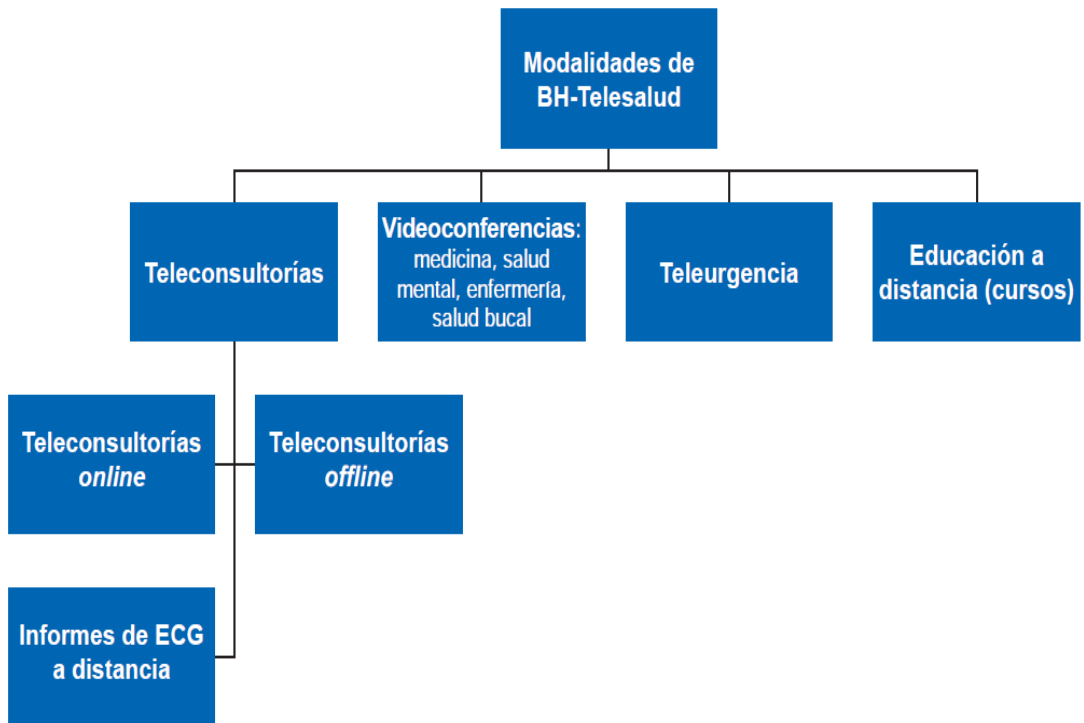
Tele dermatología



Iniciativas en Telepatología y Telepsiquiatría por Región (En porcentajes)

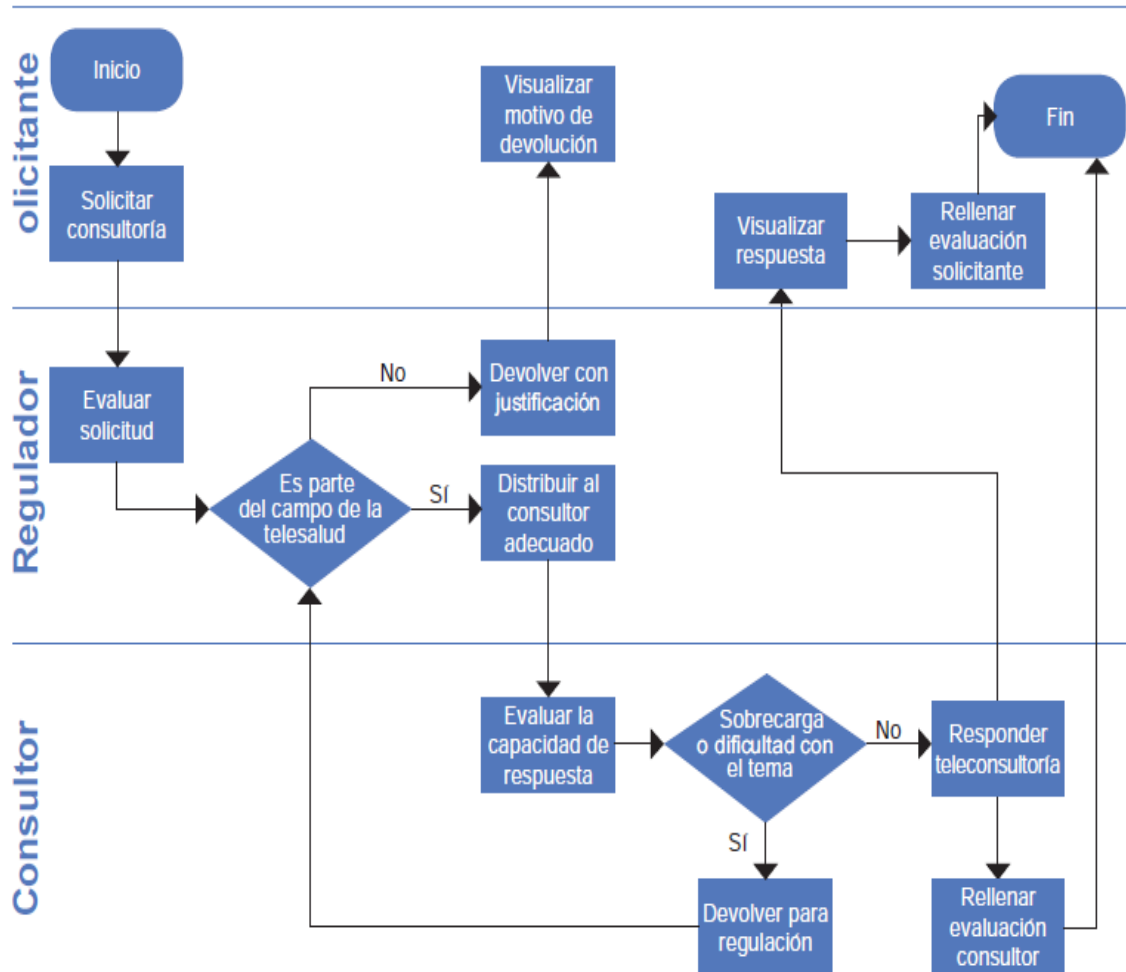


Campos de la Telemedicina

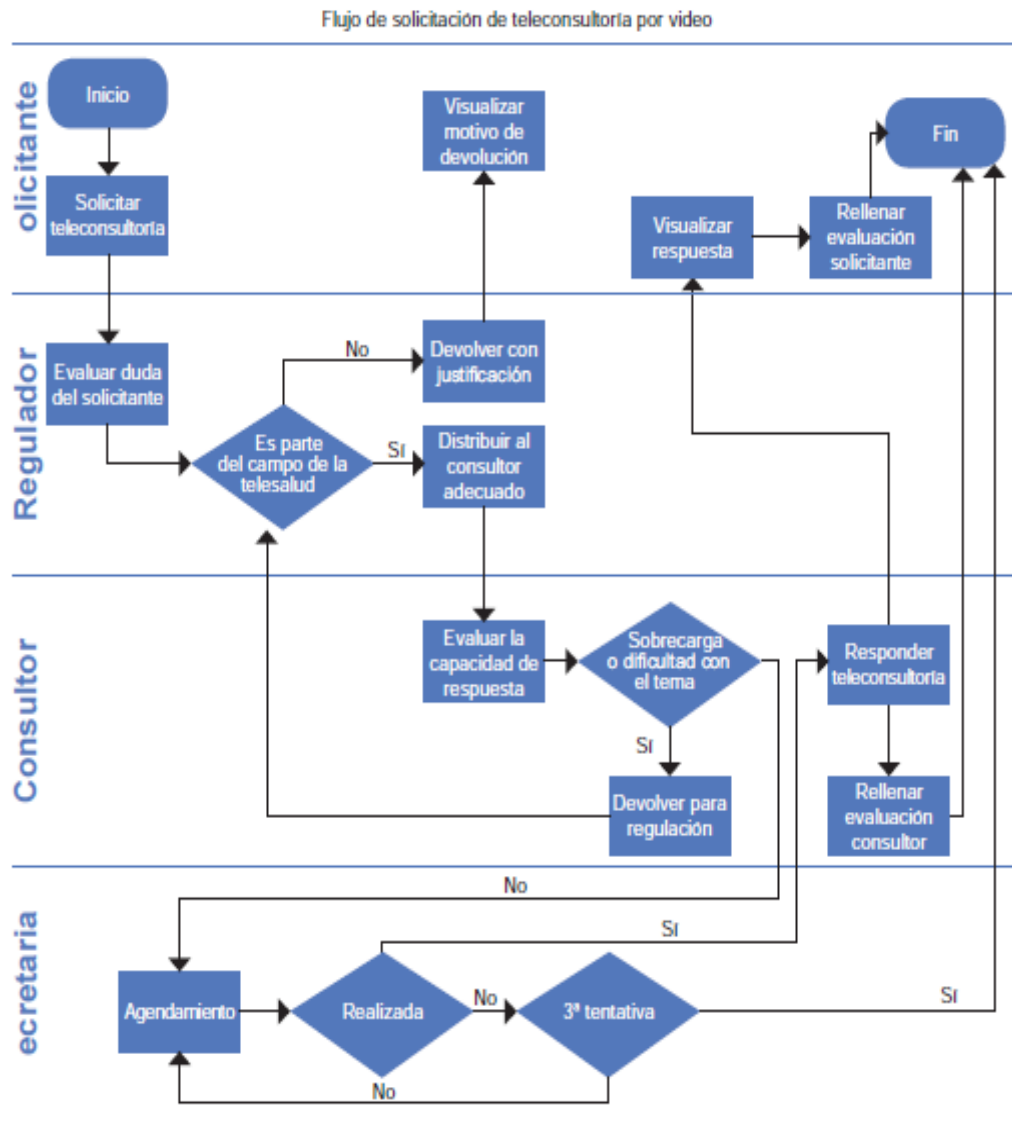


Flujo de Teleconsultorías asincrónicas, telesalud RS, 2011

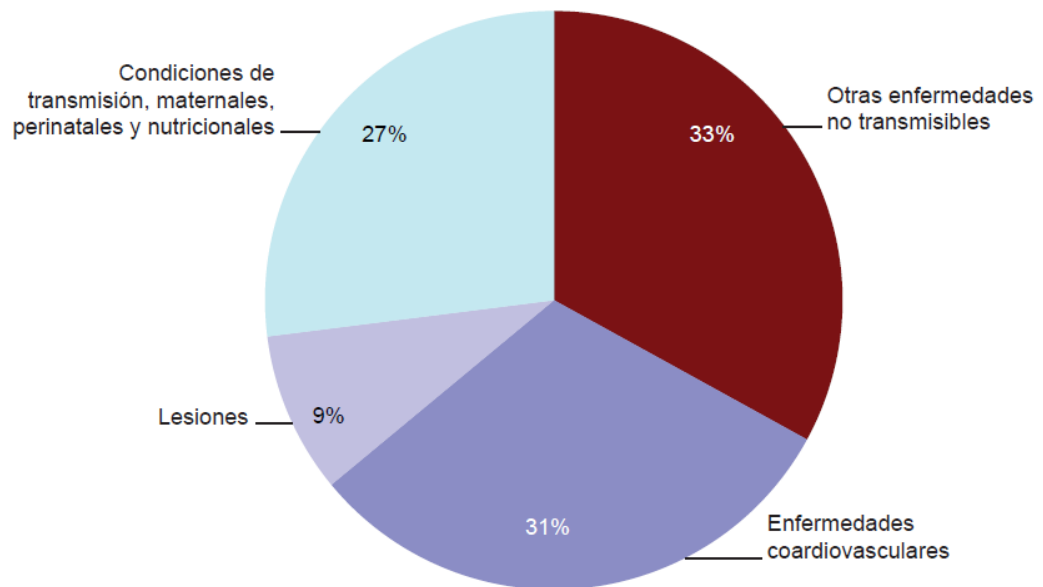
Flujo de teleconsulta de texto y material (asincrónica)



Flujo de Teleconsultorías Sincrónicas, Telesalud RS, 2011



Distribución de las Principales Causas de Muerte en el Mundo



Ecuador: Diez Principales Causas de Muerte (Genera), 2009

Número	Causa de muerte	Número	Porcentaje	Tasa ^a
1	Diabetes mellitus	4 067	6,8	29,0
2	Enfermedades cerebro vasculares	3 789	6,3	27,1
3	Accidentes de transporte terrestre	3 176	5,3	22,7
4	Enfermedades hipertensivas	3 158	5,3	22,5
5	Influenza y neumonia	3 099	5,2	22,1
6	Enfermedades isquémicas del corazón	2 293	3,8	16,4
7	Agresiones (homicidios)	2 187	3,7	15,6
8	Insuficiencia cardíaca. Complicaciones y enfermedades mal definidas	1 907	3,2	13,6
9	Cirrosis y otras enfermedades del hígado	1 902	3,2	13,6
10	Neoplasia maligna del estómago	1 626	2,7	11,6

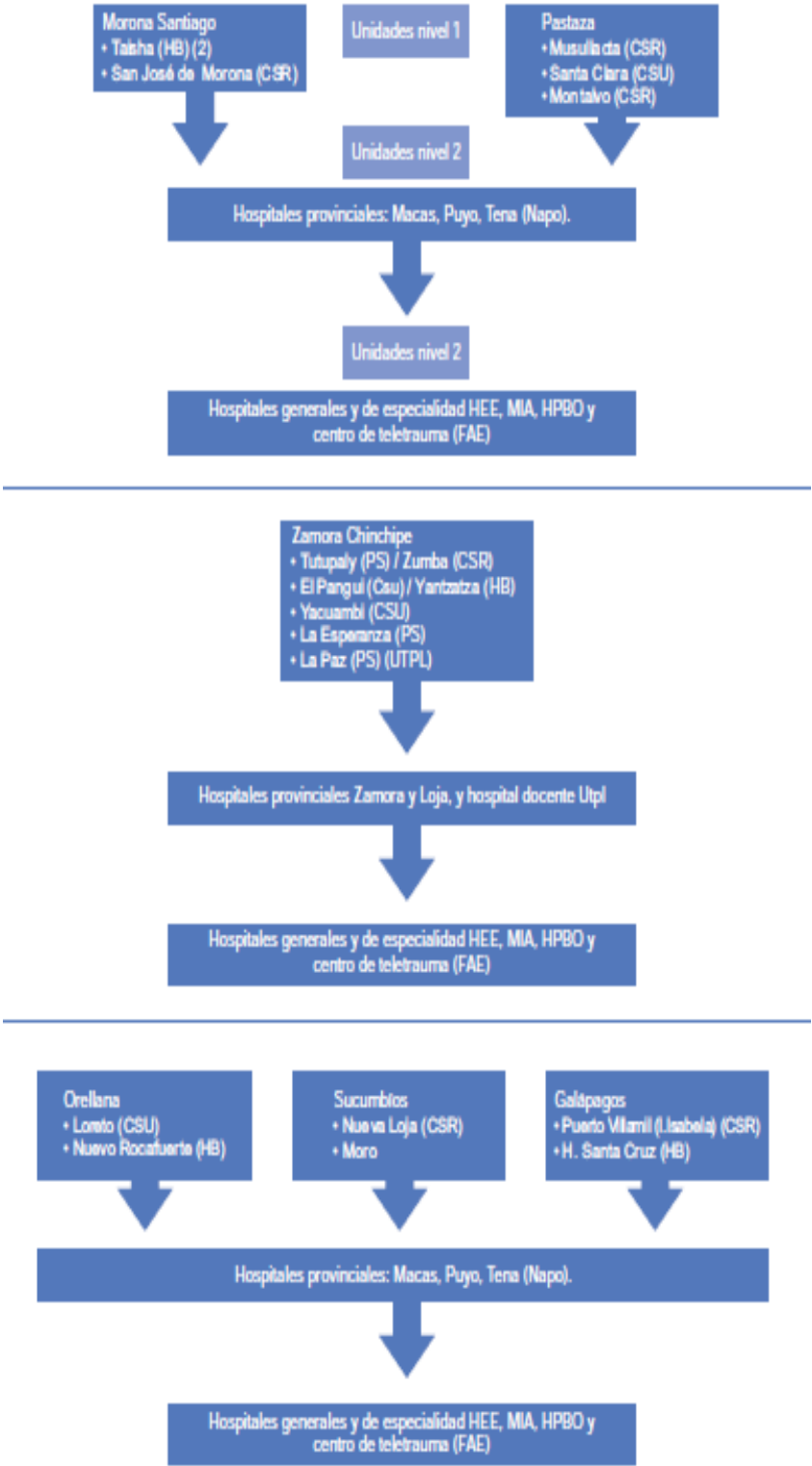
Fuente: Instituto Ecuatoriano de Estadísticas y Censos (INEC), *Anuario de estadísticas vitales, nacimientos y defunciones*, 2009.

Prioridades de Investigación del MSP Año 2009

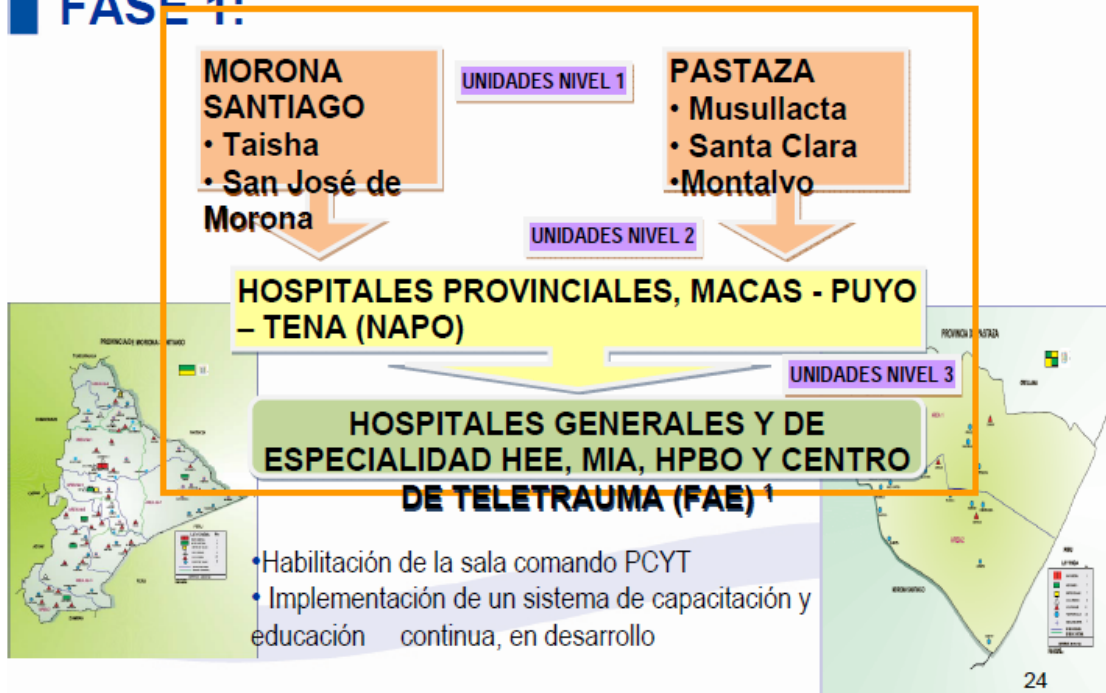
1.-SANEAMIENTO-CONTAMINACION AMBIENTAL	
2.-MEDICINA PREVENTIVA/ EDUCACION PARA LA SALUD	
3.-VIOLENCIA INFANTIL	
4.-ACCIDENTES DE TRANSITO	
5.-ENFERMEDADES INFECCIOSAS	
6.-ENFERMEDADES CRONICO DEGENERATIVAS: CARDIO VASCULARES/ GENETICAS/CANCER	
7.-PROBLEMAS DE LA ADOLESCENCIA	
8.-ENFERMEDADES DE TRANSMISION SEXUAL	
9.-TRAUMA	
10.-SALUD SEXUAL Y REPRODUCTIVA	
11.-SALUD PUBLICA Y EPIDEMIOLOGICA	
12.-TRANSTORNOS DE LA NUTRICION Y ENFERMEDADES METABOLICAS	
13.-POLITICAS PUBLICAS EN SALUD	
14.-RECURSOS HUMANOS	
15.-DESARROLLO SOCIO ECONOMICO	
16.-METODOLOGIA DE LA INVESTIGACION	
17.-SICOLOGIA	
18.-ENFERMEDADES GASTRICAS	
19.-INFRAESTRUCTURA	
20.-TELEMEDICINA	



Estructura de la Red Nacional De Telemedicina Fases 1 y 2 Amazonía, 2009 – 2012



FASE 1:



FASE 2 (cont.):

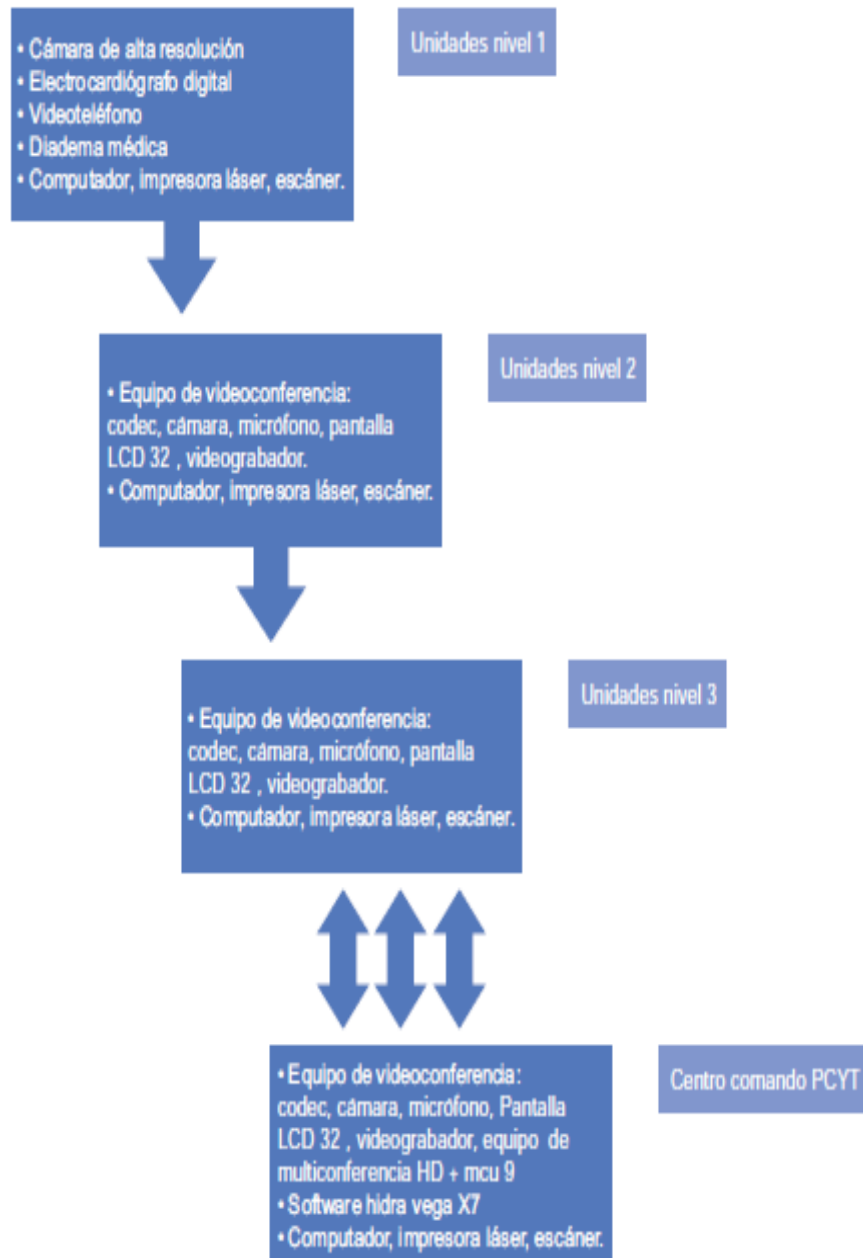




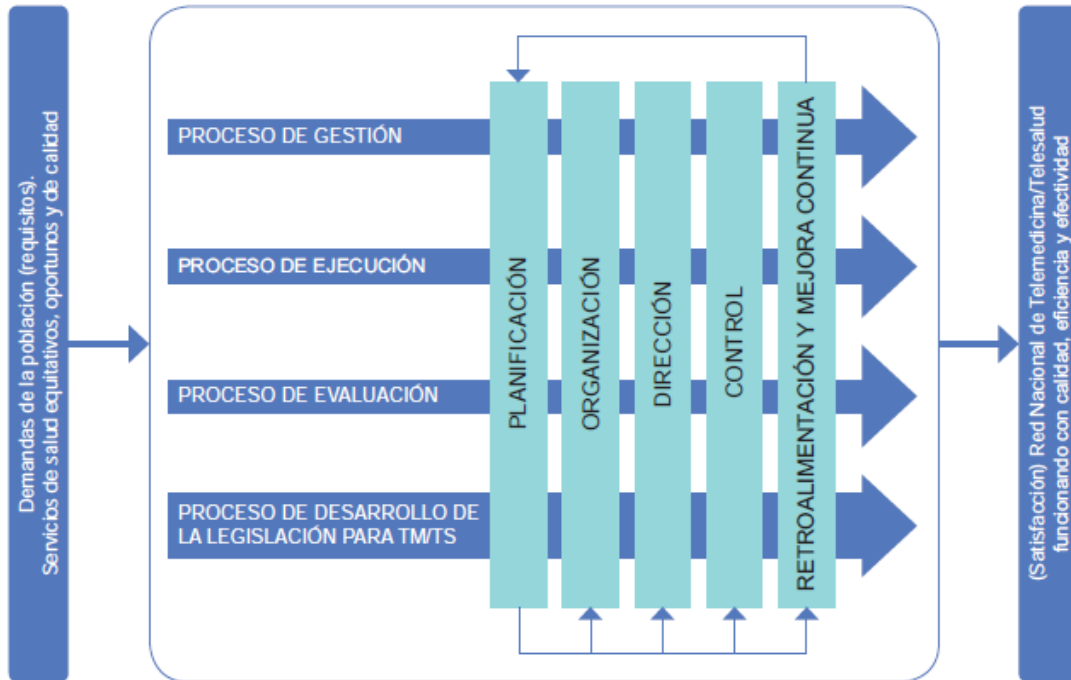
Fase Regional



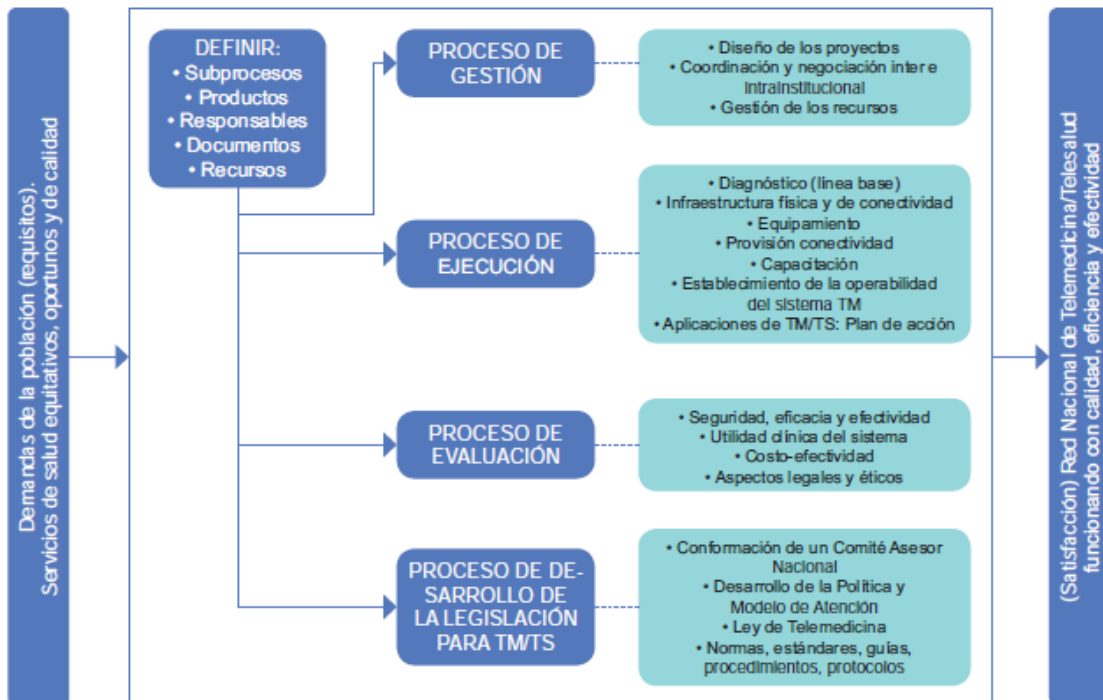
Equipamiento: Equipos médicos, computación y comunicación



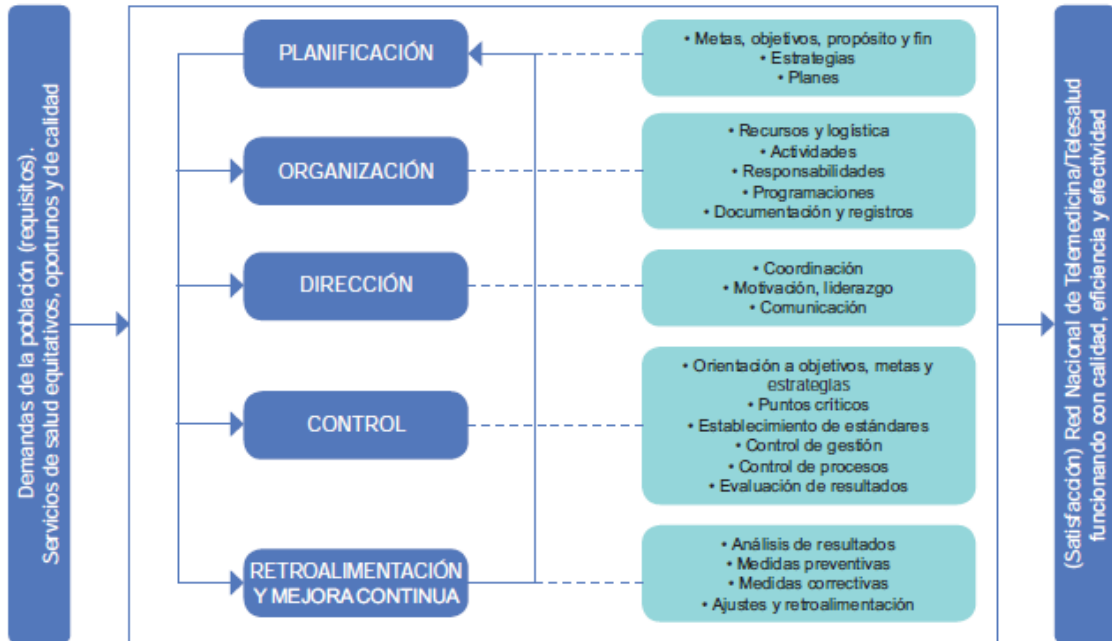
Gestión de Calidad en la Implementación de la Telemedicina



Proyectos de Telemedicina, estructura General del proceso de Implementación



Gestión en cada proceso de implementación



Anexo 3

Medidor de Presión Arterial Omron

Especificaciones del Omron

Monitor de presión arterial automático



ESPECIFICACIONES DE EMPAQUE:

CAJA:	CANTIDAD:	PESO APROXIMADO:	DIMENSIONES APROXIMADAS: (ancho x altura x profundidad)
Unidad:	1 unidad	710 g	115mm x 162mm x 95mm
Caja máster:	10 unidades	8.1 kg	499mm x 195mm x 245mm
Código UPC:	073796711399		

ESPECIFICACIONES:

Modelo	HEM-7113
Pantalla	Pantalla digital LCD
Rango de medición	Presión: 0 a 299 mmHg, Pulso: 40 a 180/min.
Precisión/Calibración	Presión: ± 3 mmHg o 2% de lectura, Pulso: $\pm 5\%$ de lectura
Inflado	Automático por bomba eléctrica
Desinflado	Válvula de liberación automática de presión
Liberación rápida de presión	Válvula de liberación automática
Detección de presión	Sensor de presión capacitivo
Método de medición	Método oscilométrico
Detección del pulso	Sensor de presión de tipo de capacitancia
Fuente de alimentación	4 pilas "AAA" de 1.5V
Vida útil de las pilas	Aproximadamente 300 usos con 4 pilas nuevas
Temperatura de operación/Humedad	10°C a 40°C / 15 a 90% de humedad relativa
Temperatura de almacenamiento/Humedad/Presión de aire	-20°C a 60°C / 10 a 95% de humedad relativa / 700-1060 hPa
Peso de la unidad principal	Aproximadamente 245g (sin incluir las pilas)
Dimensiones de la unidad principal	Aproximadamente 128 mm x 104 mm x 64 mm
Tamaño del brazalete	Aproximadamente 146 mm x 446 mm, Tubo del brazalete 600 mm
Circunferencia del brazalete	Se adapta a una circunferencia de brazo de 220 mm a 320 mm
Memoria	Hasta 30 lecturas
Contenidos	Unidad principal, brazalete estándar para brazo adulto, manual de instrucciones, Guía rápida, Libro de Registro, Estuche, 4 Pilas AAA
Accesorios opcionales	Brazalete grande 320mm - 420mm



= Tipo B

NOTAS: Estas especificaciones están sujetas a cambio sin previo aviso. La eliminación de este producto y de pilas usadas deben realizarse de acuerdo con las normas nacionales para la eliminación de productos electrónicos.

Distribuido por:
Omron Healthcare, Inc.
 1200 Lakeside Drive
 Bannockburn, IL 60015, USA
www.omronhealthcare.com

OMRON

© 7/2010 Omron Healthcare, Inc.

Símbolos de la Pantalla del Omron

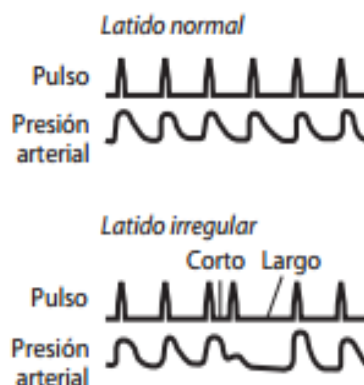
SÍMBOLOS DE LA PANTALLA

SÍMBOLO DE LATIDO IRREGULAR (🫀)

Cuando el monitor detecta un ritmo irregular dos o más veces durante la medición, aparece en la pantalla el símbolo de latido irregular (🫀) con los resultados de su medición.

El latido irregular se define como un ritmo que es más del 25% más lento o más del 25% más rápido que el ritmo promedio que se detecta mientras el monitor mide la presión arterial sistólica y diastólica.

Si se muestra el símbolo de latido irregular (🫀) con los resultados de su medición, le recomendamos que consulte a su médico y siga sus instrucciones.



SÍMBOLO DE LATIDO / INDICADOR DE HIPERTENSIÓN (🫀)

TOMAR UNA MEDICIÓN

El símbolo de latido (🫀) parpadea en la pantalla durante la medición. El símbolo de latido (🫀) parpadea ante cada latido del corazón.

Una vez que se completa la medición, el símbolo de latido (🫀) parpadea en la pantalla junto con la presión arterial y el pulso, si la lectura es superior a 135 para el valor de la presión arterial sistólica y/o superior a 85 para el valor de la presión arterial diastólica.

USO DE LA FUNCIÓN MEMORIA

El símbolo de latido (🫀) parpadea cuando la lectura de la presión arterial almacenada en la memoria es superior a 135 para el valor de la presión arterial sistólica y/o superior a 85 para el valor de la presión arterial diastólica.

La Asociación Americana del Corazón (AHA, por sus siglas en inglés) recomienda los siguientes valores como límite máximo de una lectura normal de presión arterial tomada en el hogar:

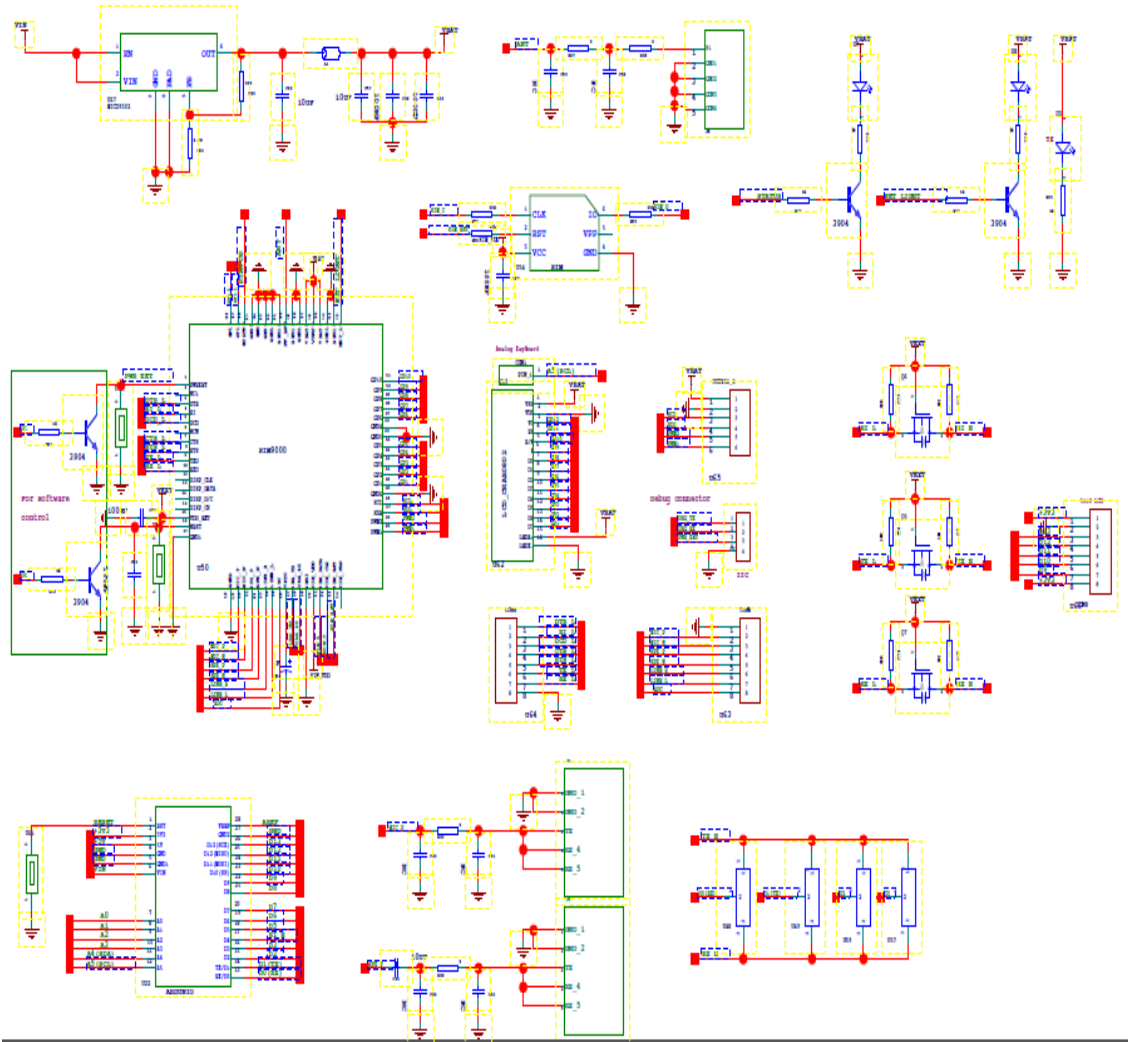
Pautas de la AHA para la medición de la presión arterial normal máxima en el hogar	
Presión arterial sistólica	135 mmHg
Presión arterial diastólica	85 mmHg

Esta es una pauta general dado que la presión arterial depende de la edad y del estado de salud de la persona. La AHA recomienda un objetivo de presión arterial medida en el hogar más bajo para ciertos pacientes, como los pacientes diabéticos, mujeres embarazadas y pacientes con insuficiencia renal¹.

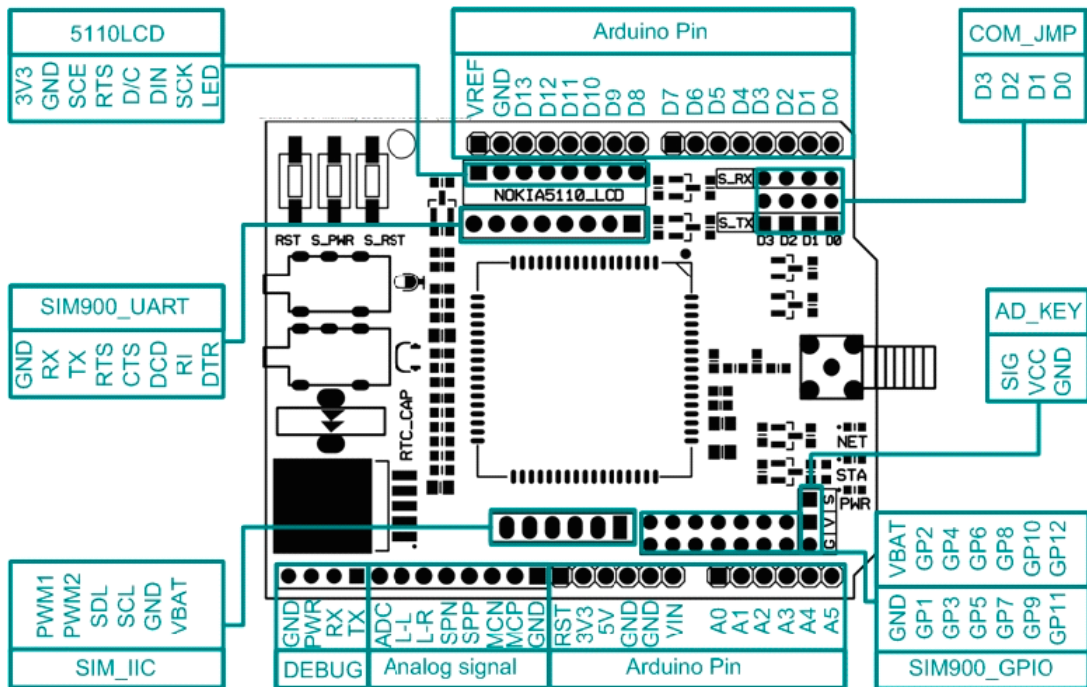
Anexo 4

Shield GSM

Datasheet Ecom Sim 900



Módulo SIM 900



GSM/GPRS SHIELD EFCOM - Módulo SIM900 - Comunicación GSM/GPRS/SMS

Actualizado hace más de un año

La placa GSM SHIELD EFCOM nos proporciona conectividad GSM/GPRS a nuestro proyecto ARDUINO de una forma muy sencilla. Esta placa Shield es compatible con todas las placas que tengan el mismo factor de forma y pinout de la Arduino Duemillanove/UNO. La placa esta basada en el módulo GSM SIM900 de la firma SIMCOM. Este módulo proporciona una completa solución GSM/GPRS cuatribanda 850/900/1800/1900MHz con cobertura mundial, apto para todo tipo de aplicaciones: Monitoreo y control remoto, robótica, vehículos autónomos, seguimiento de objetos o vehiculos, etc.

La GPRS Shield se configura y controla a través de una conexión serial (UART) y empleando comandos AT. Solo es necesario "enchufar" la placa shield sobre una placa arduino compatible y está lista para funcionar. La placa proporciona diferentes opciones de conexión a los pines arduino empleando jumpers, esto permite adaptar la placa al uso de

distintos shields o conexiones seriales disponibles por hardware o software.

én proporciona las conexiones de audio del módulo SIM900 y acceso a TODOS los pines del mismo.

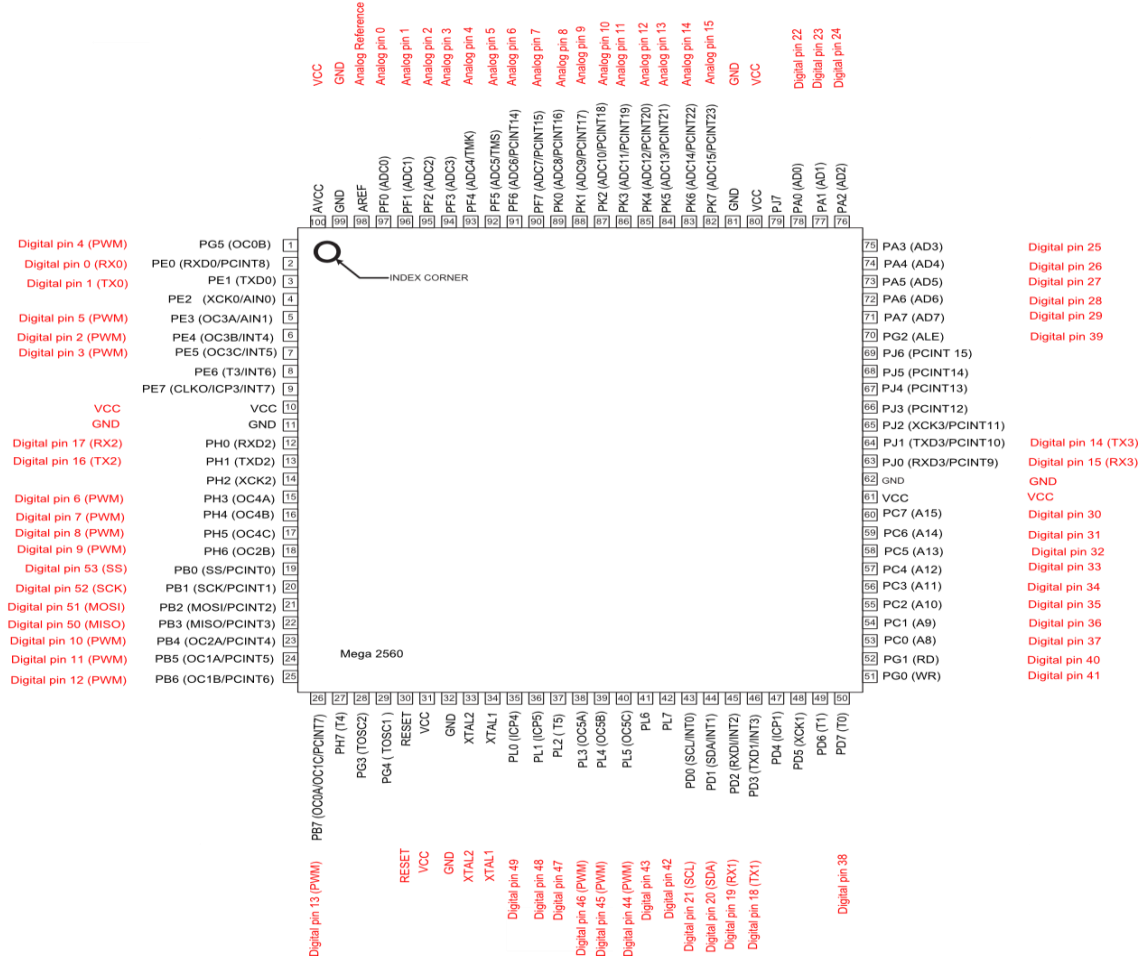
ESPECIFICACIONES

- Totalmente compatible con Arduino/Funduino/Mega.
- Conexión serial configurable mediante jumpers: Hardware serial (D0/D1) o Software serial (D2/D3).
- Todos los pines del módulo disponibles en pads de 2.54mm.
- Respaldo del RTC del módulo mediante Super Capacitor.
- CuatriBanda 850/900/1800/1900MHz.
- GPRS Multi Slot class 8/10.
- Control mediante comandos AT (GSM 07.07 ,07.05 y comandos AT EFCOM).
- Voltage de alimentación: 3,1-4,8V.

Anexo 5

Arduino

Arduino Mega 2560



Arduino Mega 2560 PIN Mapping Table

Pin Number	Pin Name	Mapped Pin Name
1	PG5 (OC0B)	Digital pin 4 (PWM)
2	PE0 (RXD0/PCINT8)	Digital pin 0 (RX0)
3	PE1 (TXD0)	Digital pin 1 (TX0)
4	PE2 (XCK0/AIN0)	
5	PE3 (OC3A/AIN1)	Digital pin 5 (PWM)
6	PE4 (OC3B/INT4)	Digital pin 2 (PWM)
7	PE5 (OC3C/INT5)	Digital pin 3 (PWM)
8	PE6 (T3/INT6)	
9	PE7 (CLKO/ICP3/INT7)	
10	VCC	VCC
11	GND	GND
12	PH0 (RXD2)	Digital pin 17 (RX2)
13	PH1 (TXD2)	Digital pin 16 (TX2)
14	PH2 (XCK2)	
15	PH3 (OC4A)	Digital pin 6 (PWM)
16	PH4 (OC4B)	Digital pin 7 (PWM)
17	PH5 (OC4C)	Digital pin 8 (PWM)
18	PH6 (OC2B)	Digital pin 9 (PWM)
19	PB0 (SS/PCINT0)	Digital pin 53 (SS)
20	PB1 (SCK/PCINT1)	Digital pin 52 (SCK)
21	PB2 (MOSI/PCINT2)	Digital pin 51 (MOSI)
22	PB3 (MISO/PCINT3)	Digital pin 50 (MISO)
23	PB4 (OC2A/PCINT4)	Digital pin 10 (PWM)
24	PB5 (OC1A/PCINT5)	Digital pin 11 (PWM)
25	PB6 (OC1B/PCINT6)	Digital pin 12 (PWM)
26	PB7 (OC0A/OC1C/PCINT7)	Digital pin 13 (PWM)
27	PH7 (T4)	
28	PG3 (TOSC2)	
29	PG4 (TOSC1)	
30	RESET	RESET
31	VCC	VCC
32	GND	GND
33	XTAL2	XTAL2

34	XTAL1	XTAL1
35	PL0 (ICP4)	Digital pin 49
36	PL1 (ICP5)	Digital pin 48
37	PL2 (T5)	Digital pin 47
38	PL3 (OC5A)	Digital pin 46 (PWM)
39	PL4 (OC5B)	Digital pin 45 (PWM)
40	PL5 (OC5C)	Digital pin 44 (PWM)
41	PL6	Digital pin 43
42	PL7	Digital pin 42
43	PD0 (SCL/INT0)	Digital pin 21 (SCL)
44	PD1 (SDA/INT1)	Digital pin 20 (SDA)
45	PD2 (RXDI/INT2)	Digital pin 19 (RX1)
46	PD3 (TXD1/INT3)	Digital pin 18 (TX1)
47	PD4 (ICP1)	
48	PD5 (XCK1)	
49	PD6 (T1)	
50	PD7 (T0)	Digital pin 38
51	PG0 (WR)	Digital pin 41
52	PG1 (RD)	Digital pin 40
53	PC0 (A8)	Digital pin 37
54	PC1 (A9)	Digital pin 36
55	PC2 (A10)	Digital pin 35
56	PC3 (A11)	Digital pin 34
57	PC4 (A12)	Digital pin 33
58	PC5 (A13)	Digital pin 32
59	PC6 (A14)	Digital pin 31
60	PC7 (A15)	Digital pin 30
61	VCC	VCC
62	GND	GND
63	PJ0 (RXD3/PCINT9)	Digital pin 15 (RX3)
64	PJ1 (TXD3/PCINT10)	Digital pin 14 (TX3)
65	PJ2 (XCK3/PCINT11)	
66	PJ3 (PCINT12)	
67	PJ4 (PCINT13)	
68	PJ5 (PCINT14)	
69	PJ6 (PCINT 15)	
70	PG2 (ALE)	Digital pin 39
71	PA7 (AD7)	Digital pin 29

72	PA6 (AD6)	Digital pin 28
73	PA5 (AD5)	Digital pin 27
74	PA4 (AD4)	Digital pin 26
75	PA3 (AD3)	Digital pin 25
76	PA2 (AD2)	Digital pin 24
77	PA1 (AD1)	Digital pin 23
78	PA0 (AD0)	Digital pin 22
79	PJ7	
80	VCC	VCC
81	GND	GND
82	PK7 (ADC15/PCINT23)	Analog pin 15
83	PK6 (ADC14/PCINT22)	Analog pin 14
84	PK5 (ADC13/PCINT21)	Analog pin 13
85	PK4 (ADC12/PCINT20)	Analog pin 12
86	PK3 (ADC11/PCINT19)	Analog pin 11
87	PK2 (ADC10/PCINT18)	Analog pin 10
88	PK1 (ADC9/PCINT17)	Analog pin 9
89	PK0 (ADC8/PCINT16)	Analog pin 8
90	PF7 (ADC7)	Analog pin 7
91	PF6 (ADC6)	Analog pin 6
92	PF5 (ADC5/TMS)	Analog pin 5
93	PF4 (ADC4/TMK)	Analog pin 4
94	PF3 (ADC3)	Analog pin 3
95	PF2 (ADC2)	Analog pin 2
96	PF1 (ADC1)	Analog pin 1
97	PF0 (ADC0)	Analog pin 0
98	AREF	Analog Reference
99	GND	GND
100	AVCC	VCC

Modelos de Arduino

Familia Arduino

Como toda familia que se respete, tenemos de los integrantes más grandes a los más pequeños, te invitamos a conocerlos:

Arduino UNO



El más vendido,
todo en uno

Arduino Mega2560



Muchas entradas y
salidas

Arduino Pro



Ligero y práctico

Arduino Fio



Programación
XBee

Arduino Mini



Pequeño, ideal
para protoboard

Arduino Bluetooth



Programación vía
bluetooth

Arduino Mega ADK



Para conectar el
Android

Arduino LyliPad



Para artistas y
manejar e-textils

Arduino Nano



Un pequeño pode-
roso

Arduino Serial



El inicio del pro-
yecto Arduino

Arduino Leonardo



El más joven y
económico

Expandir Arduino con los shields

Un Shield o escudo es una placa que permite expandir funcionalidades a tu Arduino, con lo cual puedes conectar motores, o a la red celular, a una red WiFi, a una red Ethernet o tener un MP3 en el Arduino, entre muchos más solo mira:

Celular



Ethernet



Proto



GPS



XBee



WiFi



LCD a color



USB host



Joystick



Motores DC



SD Card



Anexo 6

Pruebas y Resultados

Diagrama Arduino, Omron, Celular

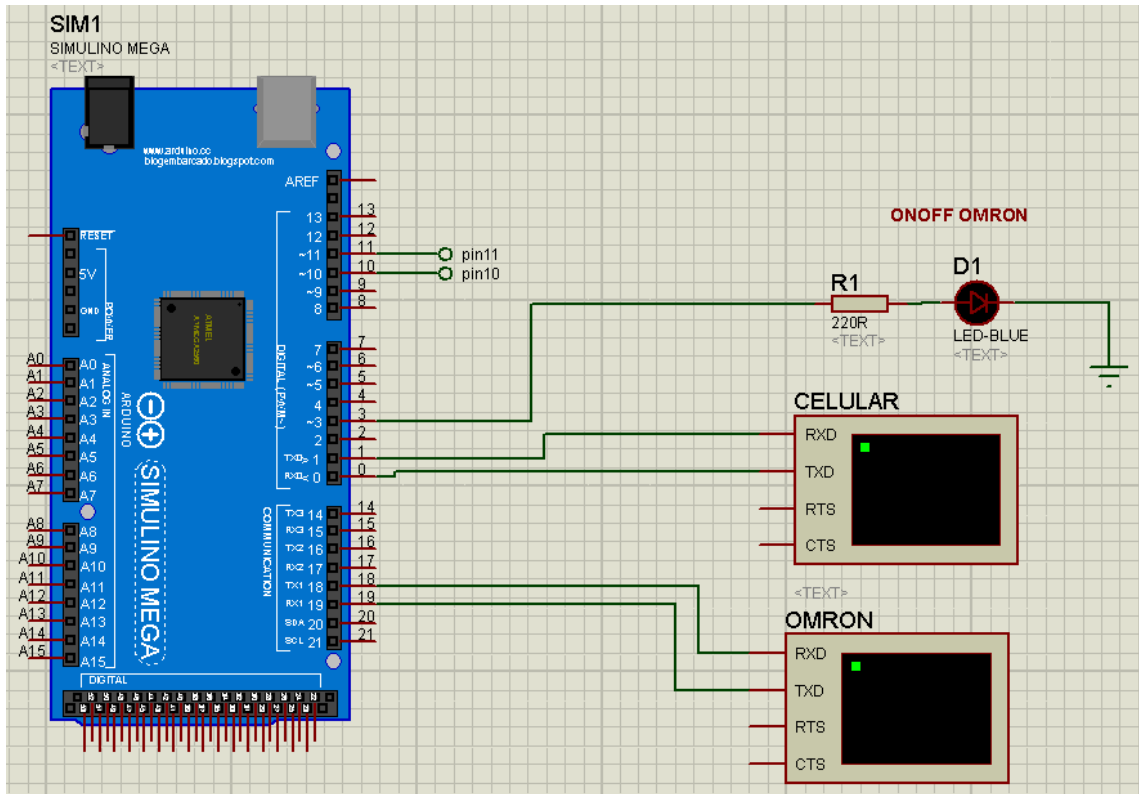
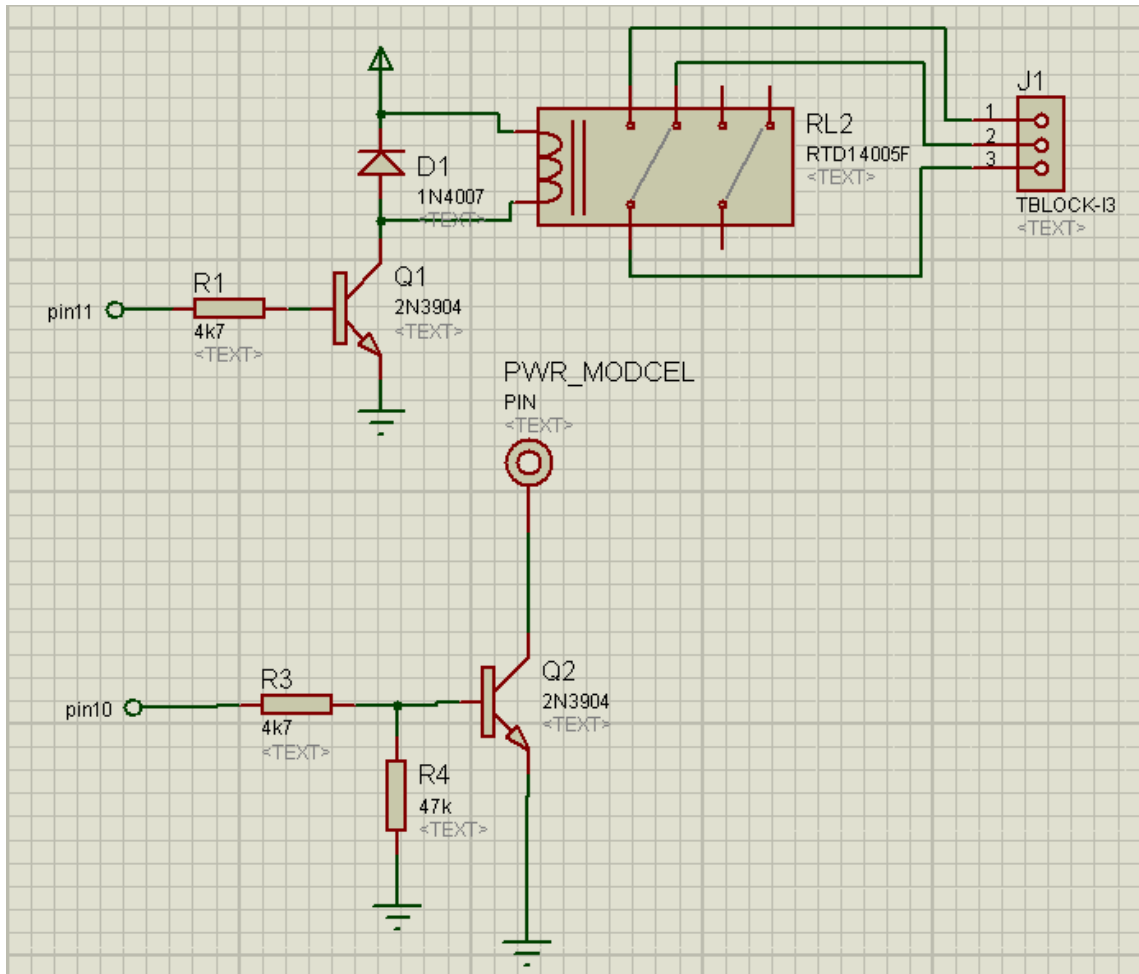
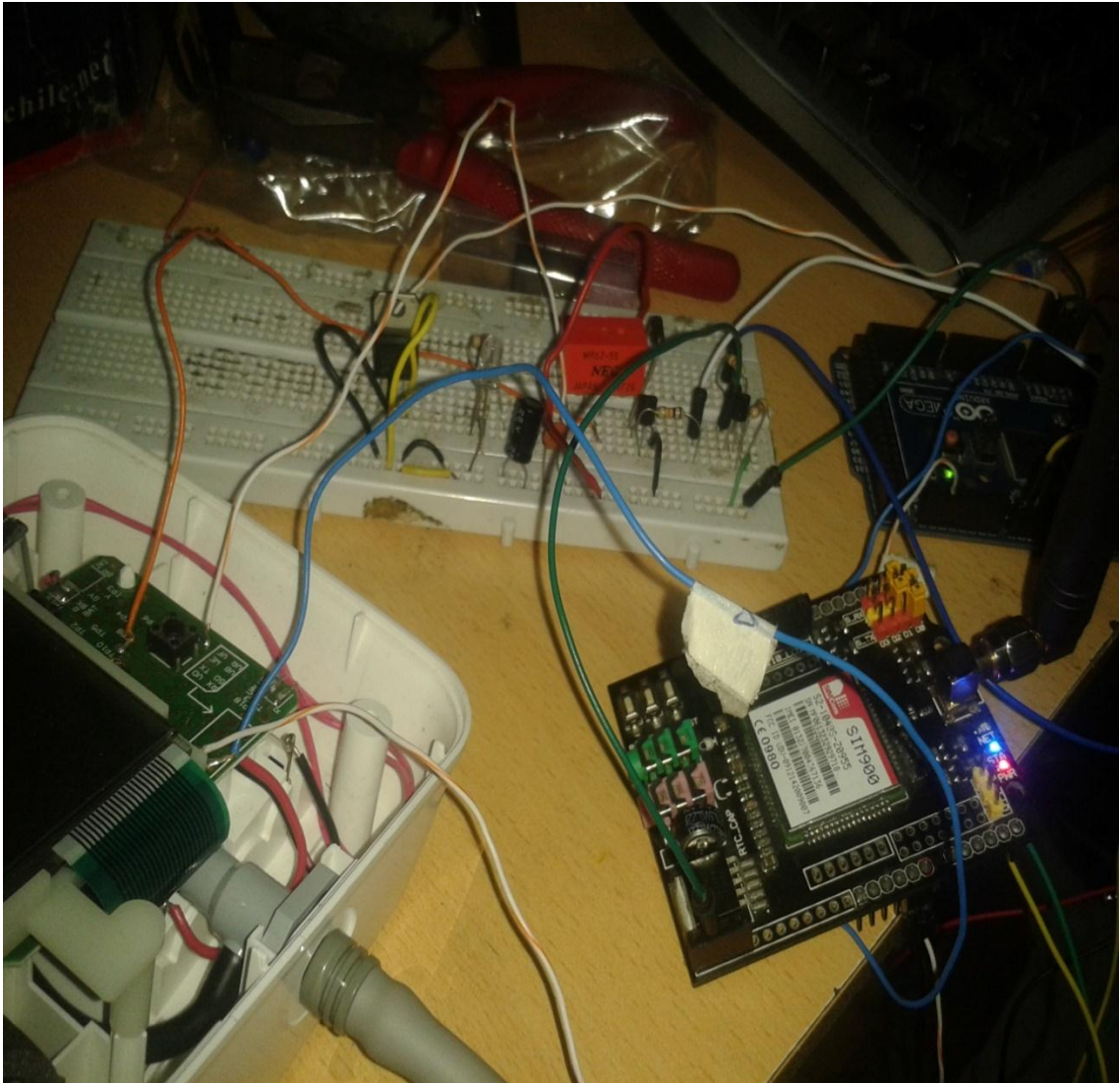


Diagrama de Relé



Sistema de Medidor inteligente



Datos de Medición Obtenidos en el Omron y Celular

