



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

DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DE ALERTA
TEMPRANA DE SISMOS MEDIANTE REDES SWAP CON NODOS
PANSTAMP NRG PARA LA ESPOCH

Trabajo de titulación presentado para optar al grado académico de:

INGENIERO ELECTRÓNICO EN TELECOMUNICACIONES Y REDES

AUTORES: ALAN GUSTAVO GARCES DEMERA
SERGIO DANIEL DEMERA CHARCOPA

TUTOR: ING. OSWALDO MARTINEZ GUASHIMA MSc.

Riobamba-Ecuador

2016

@2016, Alan Gustavo Garcés Demera y Sergio Daniel Demera Charcopa

Se autoriza la reproducción total o parcial, con fines académicos, por cualquier medio o procedimiento, incluyendo la cita bibliográfica del documento, siempre y cuando se reconozca el Derecho de Autor.

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

El Tribunal del Trabajo de Titulación certifica: “DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DE ALERTA TEMPRANA DE SISMOS MEDIANTE REDES SWAP CON NODOS PANSTAMP NRG PARA LA ESPOCH”, de responsabilidad de los señores Alan Gustavo Garcés Demera y Sergio Daniel Demera Charcopa, ha sido minuciosamente revisado por los miembros del Tribunal del Trabajo de Titulación, quedando autorizada su presentación.

NOMBRE	FIRMA	FECHA
Ing. Washington Luna MSc.		
DECANO FACULTAD DE INFORMÁTICA Y ELECTRÓNICA	_____	_____
Ing. Franklin Moreno MSc.		
DIRECTOR DE ESCUELA DE INGENIERÍA ELECTRÓNICA EN TELECOMUNICACIONES Y REDES	_____	_____
Ing. Oswaldo Martínez MSc.		
DIRECTOR DEL TRABAJO DE TITULACIÓN	_____	_____
Ing. José Guerra MSc.		
MIEMBRO DEL TRIBUNAL	_____	_____

NOTA DEL TRABAJO DE TITULACIÓN ESCRITA: _____

“Nosotros, Alan Gustavo Garcés Demera y Sergio Daniel Demera Charcopa, somos los responsables de las ideas, doctrinas y resultados expuestos en este Trabajo de Titulación y el patrimonio intelectual del Trabajo de Titulación pertenecen a la Escuela Superior Politécnica de Chimborazo”.

Alan Gustavo Garcés Demera

Sergio Daniel Demera Charcopa

DEDICATORIA

Dedico el presente trabajo primero a Dios que supo guiarme por el buen camino, a mi madre por el esfuerzo realizado para que pueda culminar mi carrera universitaria sin que me falte algo, quien ha sido ente fundamental en este logro que con sus consejos y enseñanza de valores han hecho que sea una persona de bien, demostrándome siempre que puedo contar con ella incondicionalmente.

A mis tíos Edgar y Carmen por brindarme su apoyo y la fuerza para seguir adelante, que con su ejemplo me demostraron que nada es imposible si las cosas se las hacen con esfuerzo y dedicación, finalmente a mis amigos con quienes al compartir las aulas de clase llenas de conocimiento y también risas, aprendimos a conocernos y apoyarnos unos a los otros demostrándonos siempre una verdadera amistad.

Alan

El presente trabajo está dedicado a Dios por guiar mi camino y darme fuerzas para seguir adelante y encarar las adversidades.

A mis padres, por sus consejos y ayuda en los momentos difíciles, por el apoyo moral y económico, ya que en el transcurso de mi vida han estado pendientes de mí, corrigiéndome, celebrando mis triunfos y reanimándome en mis fracasos, a ellos que siempre están dispuestos a extenderme su mano sin pedir nada a cambio.

A mis hermanos, por ser ejemplo y motivación, por estar a mi lado siempre incondicionalmente a pesar de la distancia.

Sergio

AGRADECIMIENTO

Este trabajo de titulación se debe al esfuerzo y esmero que hemos dedicado en su desarrollo quienes somos integrantes del mismo, primeramente, queremos agradecer a Dios ser maravilloso que nos dio fuerza y fe para creer lo que nos parecía imposible de terminar.

A nuestros padres porque día a día con su amor y confianza han sido los cimientos para este logro, dándonos la fortaleza para seguir adelante.

A la Lic. Mariana Benítez por su apoyo total y su amistad, a nuestro Tutor, Asesor y maestros que, con sus conocimientos, orientaciones y sobre todo paciencia ayudaron a que sea posible el desarrollo del presente trabajo.

Alan y Sergio

TABLA DE CONTENIDO

	Páginas
PORTADA	
DERECHO DE AUTOR.....	i
CERTIFICACIÓN.....	ii
DECLARACIÓN DE RESPONSABILIDAD.....	iii
DEDICATORIA.....	iv
AGRADECIMIENTO	vi
TABLA DE CONTENIDO	vii
ÍNDICE DE TABLAS.....	x
ÍNDICE DE FIGURAS	xi
INDICE DE ANEXOS.....	xiii
RESUMEN.....	xiv
SUMMARY.....	xv
INTRODUCCIÓN.....	1
1. MARCO TEORICO	4
1.1 Tectónica de placas y Sismología.....	4
<i>1.1.1 Características de las Redes de Sensores.....</i>	<i>4</i>
<i>1.1.2 Sismología.....</i>	<i>5</i>
<i>1.1.2.1 Sismos.....</i>	<i>6</i>
1.2 Ondas Sísmicas y Tipos.....	6
<i>1.2.1 Ondas de cuerpo</i>	<i>7</i>
<i>1.2.1.1 Ondas P</i>	<i>7</i>
<i>1.2.1.2 Ondas S.....</i>	<i>7</i>
<i>1.2.2 Ondas Superficiales</i>	<i>8</i>
<i>1.2.2.1 Ondas Rayleigh.....</i>	<i>8</i>
<i>1.2.2.2 Ondas Love.....</i>	<i>9</i>
1.3 Magnitud e Intensidad sísmica.....	9
<i>1.3.1 Magnitud Sísmica</i>	<i>9</i>
<i>1.3.2 Intensidad Sísmica.....</i>	<i>10</i>
1.4 Actividad Sísmica en el Ecuador.....	11
1.5 Redes de sensores inalámbricos	12
<i>1.5.1 Características de las redes de sensores.....</i>	<i>12</i>

1.5.2	<i>Arquitectura de las redes de sensores</i>	13
1.5.3	<i>Arquitectura de los nodos</i>	17
1.5.4	<i>Comunicaciones M2M</i>	18
1.5.5	<i>Comunicaciones SWAP</i>	18
1.6	Panstamp	18
1.6.1	<i>Panstamp NRG 2.0</i>	19
1.6.2	<i>Panstick 4.0</i>	20
1.6.3	<i>Tablero de minibat ntc</i>	21
1.7	Sensor Acelerómetro MPU6050	22
1.8	GPRS Shield V2.0	22
1.9	Arduino Mega	23
2.	MARCO METODOLÓGICO	24
2.1	Comparación de sensores	24
2.1.1	<i>Sensores para medir aceleración</i>	25
2.1.2	<i>Hardware para desarrollar la comunicación</i>	27
3.	MARCO DE RESULTADOS	31
3.1	Diseño del Prototipo de la Red Inalámbrica	31
3.1.1	<i>Diagrama de bloques de la red</i>	32
3.1.2	<i>Diagrama de flujo de la red</i>	33
3.2	Configuración de la Red Inalámbrica	33
3.2.1	<i>Configuración de hardware</i>	33
3.2.1.1	<i>Configuración del Gateway Panstick</i>	33
3.2.1.2	<i>Configuración del módulo Panstamp</i>	35
3.2.1.3	<i>Configuración del módulo Arduino Mega</i>	37
3.2.1.4	<i>Configuración de los sensores MPU6050</i>	38
3.2.1.5	<i>Estructura de conexiones</i>	40
3.2.2	<i>Configuración del software</i>	42
3.2.2.1	<i>Configuración del software Realterm</i>	44
3.2.2.2	<i>Configuración del software Kst</i>	45
3.3	Pruebas de Funcionamiento del prototipo de la WSN	46
3.3.1	<i>Pruebas de alcance de la red</i>	48
3.3.2	<i>Datos obtenidos con el prototipo de WSN para la alerta temprana de sismos</i>	51
3.3.3	<i>Análisis estadístico</i>	52
3.3.4	<i>Cálculo de Pérdidas (db) en el espacio libre</i>	56
3.3.5	<i>Cálculo de la potencia recibida en el enlace inalámbrico</i>	57

3.3.6	<i>Cálculo de la distancia límite</i>	58
3.3.7	<i>Análisis económico</i>	59
3.3.8	<i>Evaluación de la fuente de alimentación</i>	62
	CONCLUSIONES	65
	RECOMENDACIONES	66
	GLOSARIO	
	BIBLIOGRAFÍA	
	ANEXOS	

ÍNDICE DE TABLAS

Tabla 2-1	Escala cuantitativa	25
Tabla 2-2	Escala cualitativa	25
Tabla 2-3	Sensores de aceleración disponibles en el mercado.....	25
Tabla 2-4	Valoración de las características para los sensores de aceleración.....	26
Tabla 2-5	Análisis de las alternativas de sensores de aceleración.....	26
Tabla 2-6	Placas de desarrollo disponibles en el mercado.....	27
Tabla 2-7	Valoración de las características de las placas de desarrollo	27
Tabla 2-8	Porcentaje de las alternativas para la comunicación.....	28
Tabla 3-1	Alcance de la comunicación para cada nodo.	48
Tabla 3-2	Datos obtenidos con el prototipo funcional	51
Tabla 3-3	Inversión en equipos para movimiento sísmico.....	60
Tabla 3-4	Valor de equipos utilizados para el prototipo	61
Tabla 3-5	Análisis comparativo de inversiones	62

ÍNDICE DE FIGURAS

Figura 1-1	Placas Tectónicas continentales	5
Figura 1-2	Propagación de la onda P	7
Figura 1-3	Propagación de la onda S	8
Figura 1-4	Propagación de la onda R.....	8
Figura 1-5	Propagación de la onda Love	9
Figura 1-6	Escala de Richter	10
Figura 1-7	Escala de Mercalli	10
Figura 1-8	Topología WSN en estrella	15
Figura 1-9	Topología WSN en árbol.....	16
Figura 1-10	Topología WSN en malla.....	17
Figura 1-11	Arquitectura del nodo.....	17
Figura 1-12	Estructura de paquetes SWAP.....	18
Figura 1-13	Modulo Panstamp NRG 2.0	19
Figura 1-14	Modulo Panstick 4.0.....	20
Figura 1-15	Minibat	21
Figura 1-16	Minibat	22
Figura 1-17	Shield GPRS.....	23
Figura 1-18	Modulo Arduino Mega 2560.....	23
Figura 2-1	Vista frontal del case	28
Figura 2-2	Conector SMA.....	29
Figura 2-3	SMA Articulada	29
Figura 2-4	Antena SMA de base magnetica	30
Figura 2-5	Battery holder.....	30
Figura 3-1	Topología Estrella en base a la ubicación de los tanques.....	31
Figura 3-2	Diagrama de bloques del prototipo de la WSN	32
Figura 3-3	Diagrama de flujo del prototipo de la WSN	33
Figura 3-4	Diagrama de flujo del Panstick	34
Figura 3-5	Instalación de librería panstamp en Arduino.....	35
Figura 3-6	Configuración de panstamp.....	36
Figura 3-7	Diagrama de flujo de los nodos.....	36
Figura 3-8	Diagrama de flujo del módulo Arduino	37
Figura 3-9	Módulo Arduino conectado al shield GPRS	38
Figura 3-10	Calibración del sensor MPU6050.....	39
Figura 3-11	Sistema prototipo final	40

Figura 3-12	Diagrama de conexiones del nodo Panstamp	41
Figura 3-13	Diagrama de conexiones del Gateway	41
Figura 3-14	Diagrama de conexiones del Gateway	42
Figura 3-15	Diagrama de flujo de supervisión.....	42
Figura 3-16	Supervisión en el PC	43
Figura 3-17	Página de inicio de Realterm.....	44
Figura 3-18	Archivo de texto creado	44
Figura 3-19	Página de inicio de Kst.....	45
Figura 3-20	Visualización gráfica de los valores.....	45
Figura 3-21	Monitorización en el PC.....	46
Figura 3-22	Monitorización en el PC.....	46
Figura 3-23	Ubicación del nodo1 cabaña del parque del estudiante.....	47
Figura 3-24	Ubicación del nodo 2 estructura de los baños parque del estudiante	47
Figura 3-25	Ubicación del nodo 3 parte posterior de la biblioteca	48
Figura 3-26	Ubicación del nodo 3 parte posterior de la biblioteca	49
Figura 3-27	Ubicación del nodo 3 parte posterior de la biblioteca	50
Figura 3-28	Ubicación del nodo 3 parte posterior de la biblioteca	50
Figura 3-29	Histograma del nodo 1	53
Figura 3-30	Histograma del nodo 2	54
Figura 3-31	Histograma del nodo 3	55
Figura 3-32	Histograma del nodo 4	56

ÍNDICE DE ANEXOS

Anexo A	Datasheet módulo Panstamp
Anexo B	Datasheet Gateway panstick
Anexo C	Datasheet Minibat
Anexo D	Datasheet Acelerometro MPU6050
Anexo E	Datasheet módulo GSM/GPRS SIM900
Anexo F	Datasheet módulo Arduino Mega
Anexo G	Código de los nodos panstamp
Anexo H	Código en Arduino del Sield GPRS SIM900
Anexo I	Configuración de Realterm
Anexo J	Configuración de Kst
Anexo K	Programa para calibrar acelerómetro MPU6050

RESUMEN

El presente trabajo de titulación consistió en el diseño en implementación de un sistema de alerta temprana de sismos mediante una red de sensores inalámbrica con nodos Panstamp mediante el protocolo SWAP dentro de la banda ISM de 915 MHz en la Escuela Superior Politécnica de Chimborazo, ubicados en un radio de 135m de la Asociación de Estudiantes Telecomunicaciones y Redes en el parque del estudiante, para brindar una alerta a través de mensajes de texto vía celular. Este trabajo busca obtener datos en tiempo real de las aceleraciones apoyándonos en la escala de Mercalli, esta escala en sus niveles bajos se basa en la percepción de las personas ante un evento sísmico y en los niveles altos en el daño estructural que se observa. Para el desarrollo del prototipo funcional se usó el método comparativo en la elección de los sensores de aceleración y las tarjetas de desarrollo, por medio del análisis de sus características como sensibilidad, inversión y alcance se realizó la elección. Además, se demostró que el tiempo de autonomía de los nodos es de 6.13 meses al ser alimentados con pilas AAA de 1150mAh, las pruebas de alcance demostraron que los nodos y el Gateway se pueden comunicar hasta una distancia de 191.58m en línea de vista. Al tomar el registro de las aceleraciones obtenidas por los nodos con un total de 3499424 muestras se calculó una varianza de $1.98 \cdot 10^{-5}$, y que sus ejes horizontales son independientes al tener una correlación entre ellos de -0.01345 demostrando la estabilidad del sistema, en cuanto a la inversión, representó un ahorro de \$26428,58 con respecto al sistema profesional del Instituto Geofísico de la Escuela Politécnica Nacional. Se recomienda a los integrantes de la Asociación de Escuela realizar el mantenimiento de los nodos cada 5 meses.

PALABRAS CLAVES: <ALERTA TEMPRANA DE SISMOS> <TECNOLOGÍA PANSTAMP> <TELECOMUNICACIONES Y REDES> <PROTOCOLO SWAP> <RED DE SENSORES INALÁMBRICOS> <MENSAJES DE TEXTO VIA CELULAR > <ESCALA DE MERCALLI> <RIOBAMBA (CANTON) >

SUMMARY

This research is intended to design and implement a system for early earthquake alert-detection through wireless web sensor with nodes Panstamp through swap protocol in the band ISM of 915 MHz into the Escuela Superior Politécnica de Chimborazo, to be located at a 135m-radius from the settings of Telecommunication student Association to web of student Park in order to issue early earthquake alert through text messages to cellphones. This research aims to gather data of earthquake accelerations at the very time it occurs by supporting of the Mercalli scale, which can give low-levels earthquake alert based on human perception as well as high-levels earthquake alert based on the earth structure damages. To develop this functional system prototype for early earthquake alert, the comparative method was used in choosing the acceleration sensor devices and acceleration development cards by analyzing the features of these earthquake alert devices in regarding their sensibility-performance, investment and the device scope. The study result reveal that the node autonomy time is 6.13 months when they are being supplied by AAA 1150mAh power battery, the research testing demonstrates that the nodes and the Gateway can be intercommunicated at 191.58m-distance on see line. The acceleration register obtained by the nodes accounts for a total of 3499424 samples by which the variance was calculated accounting for $1.98 \cdot 10^{-5}$, and the horizontal axis are independent accounting for a correlation of 0.01354 between them, which allowed to demonstrates the system stability, on the other hand, it represent a saving of \$26428.58 in regarding the investment required for the professional system for early earthquake alert-detection of the Geophysical institute from Escuela Politécnica Nacional. Therefore, it is advisable for the head director members of the Telecommunications College Association from Politécnica de Chimborazo to give a regular maintenance to nodes every 5 months.

KEY WORDS: < EARLY WARNING EARTHQUAKE > < PANSTAMP TECHNOLOGY > <TELECOMMUNICATIONS AND NETWORK> <SWAP PROTOCOL> <WIRELESS SENSOR NETWORK> <TEXT MESSAGE> <MERCALLI SCALE> <RIOBAMBA> <CANTON>.

INTRODUCCIÓN

El Ecuador está situado en una zona de alto riesgo sísmico, a lo largo de su historia se han producido muchos terremotos. Se encuentra ubicado en el noroeste de Sudamérica, limita al norte con Colombia, al este y sur con Perú, y al oeste con el Océano Pacífico, toma este nombre gracias a la línea que divide el globo en dos hemisferios denominada ecuador, tiene una superficie total de 256.370 Km², una zona continental y una zona insular localizada a 1050 Km en el Océano Pacífico.

La Cadena de Fuego del Pacífico, es una de las amenazas más importante para el Ecuador, debido a su relieve muy accidentado como consecuencia del fenómeno de subducción de la Placa de Nazca bajo la de Sudamérica originando el zócalo continental que genera movimientos tectónicos a lo largo de la misma. El sismo es el movimiento brusco de la Tierra causado por la liberación de energía acumulada durante un largo tiempo. Habitualmente estos movimientos son lentos e imperceptibles, pero en algunos el desplazamiento libera una gran cantidad de energía, cuando una de las placas se mueve bruscamente contra la otra, rompiéndola y originando un terremoto o si se produce en el mar a una determinada profundidad un tsunami.

El instituto geofísico de la Universidad Politécnica Nacional constituye el principal centro de investigación en Ecuador para el diagnóstico y la vigilancia de los peligros sísmicos y volcánicos, a partir del 13 de enero de 2003, mediante Decreto Oficial No. 3593, el Instituto Geofísico recibe del Estado ecuatoriano el encargo oficial del diagnóstico y la vigilancia de los peligros sísmicos y volcánicos en todo el territorio nacional. (IGEPN,2014).

Hoy en día con el avance de la tecnología, podemos darnos cuenta que se pueden implementar nuevos sistemas de monitoreo y prevención de sismos, desarrollando redes wsn que puedan contribuir en la detección de estos eventos y a su vez nos proporcionen información necesaria para entender un poco más estos fenómenos naturales.

Los nodos Panstamp nos permitirán realizar un monitoreo de los movimientos que se producen en el suelo, mediante la comunicación inalámbrica al Gateway panstick el cual recibe los diferentes eventos q se presenten en los nodos. El aviso de alerta será establecido al final de la red a través de un módulo arduino con una shield GSM que de acuerdo a la magnitud del evento sísmico emitirá un mensaje de texto a determinados números de celulares.

Justificación:

Justificación Teórica:

El riesgo sísmico constituye un campo importante de estudio en nuestro país debido a su localización dentro del cinturón de fuego del Pacífico y la carencia de herramientas preventivas para este tipo de catástrofes. La presente investigación desarrollara un sistema que permitirá obtener información útil y oportuna que pueda ser de ayuda a las instituciones locales responsables de la protección civil en la ejecución de sus actividades.

El sistema de alerta temprana de sismos pretende brindar un aviso con la mayor anticipación posible, para que la población afectada pueda tomar las precauciones necesarias para salvaguardar su integridad. Su aplicación permitirá generar un registro estadístico de la sismicidad en la zona de estudio facilitando la realización de futuras investigaciones concernientes al tema que sirvan para generar conocimiento.

Justificación Aplicativa:

Para lograr lo propuesto se diseñará una red WSN basada en dispositivos PanStamp NRG 2.0 que son pequeños módulos inalámbricos programables desde el IDE Arduino, estos proporcionan la conectividad y la potencia de procesamiento necesaria utilizando un bajo consumo energético.

Se realizarán mediciones de vibraciones en el suelo mediante el uso de acelerómetros MPU6050 de 3 ejes que funciona con señales de interrupción de despertador inerciales que monitorean eventos y permanecen en un modo de bajo consumo durante los períodos de inactividad., ubicados estratégicamente e interconectados de forma inalámbrica mediante la aplicación de una red SWAP, un protocolo de código abierto diseñado especialmente para radios CC11XX TI.

El Gateway panstick es quien se encargará de enviar los datos obtenidos al módulo de alarma en el que se mostrará en tiempo real los eventos, y se podrá observar sus graficas correspondientes.

Objetivo General:

Implementar una red de nodos Panstamp NRG para la alerta temprana de sismos, capaz de informar oportunamente ante un posible evento sísmico.

Objetivos Específicos:

- ❖ Analizar los requerimientos y seleccionar los dispositivos para realizar el proyecto.
- ❖ Estudiar y seleccionar el sistema de alimentación de energía para los nodos.
- ❖ Diseñar la red inalámbrica de nodos y establecer el protocolo de comunicación inalámbrica para esta aplicación.
- ❖ Implementar la red de módulos Panstamp y establecer la comunicación.
- ❖ Realizar el grafico del histograma de la señal

CAPITULO I

1. MARCO TEORICO

Este capítulo tiene como finalidad abordar los conceptos básicos necesarios para la comprensión más clara del desarrollo del proyecto que se detalla más adelante.

1.1 Tectónica de placas y Sismología

Son teorías fuertemente ligadas y de gran importancia que por medio de su estudio permiten comprender el origen y por qué se producen los sismos.

1.1.1 Características de las Redes de Sensores

La tectónica de placas permite explicar de forma integradora la mayoría de los procesos geológicos de nuestro planeta a lo largo de millones de años y ha acabado por convertirse en el mejor modelo para dar respuesta a los problemas que surgen de la investigación continua en las ciencias de la tierra, así como es la manifestación superficial de su actividad interna (Enseñanza de las Ciencias de la Tierra, 2013, p.168).

Esta propone que la superficie de la Tierra está compuesta por siete grandes fragmentos y otros más pequeños llamados placas tectónicas que miden entre 70 y 150km y constituyen la litósfera. La astenósfera es la capa debajo de la litósfera donde se producen las corrientes de convección térmica y mediante éstas, el movimiento de las placas, haciéndolas desplazarse hasta velocidades de 8cm por año, ver en la Fig 1-1.



Figura 1-1 Placas Tectónicas continentales

Fuente: (IGUNAM, 2014)

Los bordes de las placas son de tres tipos:

- **Divergentes o constructivos:** Son aquellos en donde el movimiento de las placas es de separación, dando lugar a una abertura entre ellas donde las rocas magmáticas generan nueva corteza oceánica. También se denominan zonas de dorsal o límites constructivos.
- **Convergentes o destructivos:** En este caso una de las placas (la más densa) se introduce bajo la otra en un proceso llamado subducción. Estas zonas presentan alta sismicidad y actividad volcánica.
- **Límites transformantes:** Las placas se mueven de forma paralela en sentido contrario. También son conocidos como zonas de falla transformante o límites transcurrentes, en ellos se presenta una intensa actividad sísmica.

1.1.2 Sismología

Es la ciencia que estudia los terremotos. Implica la observación de las vibraciones naturales del terreno y de las señales sísmicas generadas de forma artificial, con muchas ramificaciones teóricas y prácticas. Como rama de la geofísica, la sismología ha aportado contribuciones esenciales a la comprensión de la tectónica de placas, la estructura del interior de la Tierra, la predicción de terremotos y es una técnica valiosa en la búsqueda de minerales. (Sivana Hinojosa, 2012).

Una amenaza sísmica es denominada la estimación o el potencial de que ocurran sismos dañinos en un área geográfica. Dentro de ella se toman en cuenta los sismos que puedan ocurrir en el epicentro o en zonas aledañas (Manual Sobre Sismo Resistencia. Ministro de Ambiente, Vivienda y Desarrollo Territorial, 2007, p 10.).

1.1.2.1 Sismos

Son el rompimiento repentino de las rocas en el interior de la Tierra. Esta liberación repentina de energía se propaga en forma de ondas que provocan el movimiento del terreno. (C. Gonzales, 2009).

Según su origen los sismos se pueden clasificar como:

- **Interplaca:** Estos se originan en las zonas de contacto de las placas tectónicas, tienen una alta magnitud, un foco profundo y liberan gran cantidad de energía. Alejados de los centros de población
- **Intraplaca:** Se generan de las placas tectónicas, en las fallas locales. Tienen magnitudes pequeñas o moderadas y se producen cerca a los centros de población.
- **Volcánicos:** Son generados por la actividad propia de los volcanes, generalmente son de baja magnitud y se limitan al aparato volcánico.
- **Provocados por el ser humano:** Estos se generan por explosiones o el colapso de galerías en explosiones mineras.

1.2 Ondas Sísmicas y Tipos

Denominadas también ondas elásticas, son la propagación de perturbaciones temporales del campo de esfuerzo que generan pequeños movimientos en un medio, estas se pueden generar por movimientos telúricos naturales o generarse artificialmente por explosiones; estas ondas son estudiadas por la sismica que es una rama de la sismología,

Estas ondas pueden ser ondas de cuerpo y ondas superficiales.

1.2.1 Ondas de cuerpo

Estas ondas viajan por el interior de la tierra, siguen diversos caminos debido a la derivada densidad y composición del interior de la Tierra. Las ondas de cuerpo transmiten los temblores preliminares de un terremoto pero poseen poco poder destructivo. Estas se pueden dividir en ondas primarias (P) y ondas secundarias (S).

1.2.1.1 Ondas P

Las ondas P producen el suelo un efecto de compresión y dilatación alternada en dirección de la propagación. Estas pueden viajar a través de cualquier tipo de material, alcanzan velocidades típicas 330m/s, 1450 m/s en el agua y 5000m/s en granito.

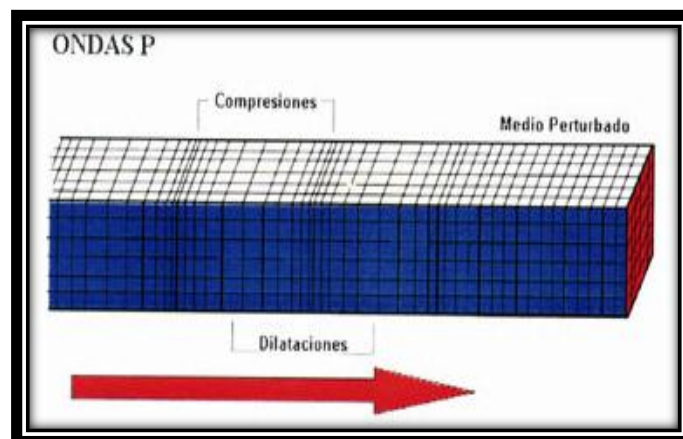


Figura 1-2 Propagación de la onda P

Fuente: (www.funvisis.gob.ve, 2014)

1.2.1.2 Ondas S

Estas ondas son transversales o de corte, lo que produce en el suelo un desplazamiento perpendicular a la dirección de propagación, alternadamente hacia un lado y hacia el otro. Estas ondas viajan únicamente a través de sólidos debido a que los líquidos no pueden soportar esfuerzos de corte, su velocidad es 58% la de una onda P y tienen una mayor amplitud.

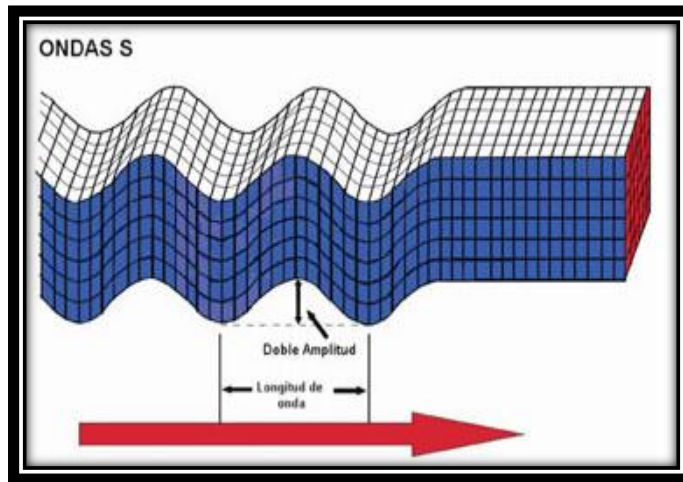


Figura 1-3 Propagación de la onda S

Fuente: (www.funvisis.gob.ve, 2014)

1.2.2 Ondas Superficiales

Estas ondas son análogas a las ondas de agua y viajan sobre la superficie de la Tierra, tienen un desplazamiento de menor velocidad a las ondas de cuerpo. Debido a su baja frecuencia provocan resonancia en edificios lo que las convierte en las ondas más destructivas. Estas pueden ser ondas Rayleigh y ondas Love.

1.2.2.1 Ondas Rayleigh

Son ondas superficiales que viajan como ondulaciones similares a las del agua. John William Strutt fue el que descubrió su existencia.

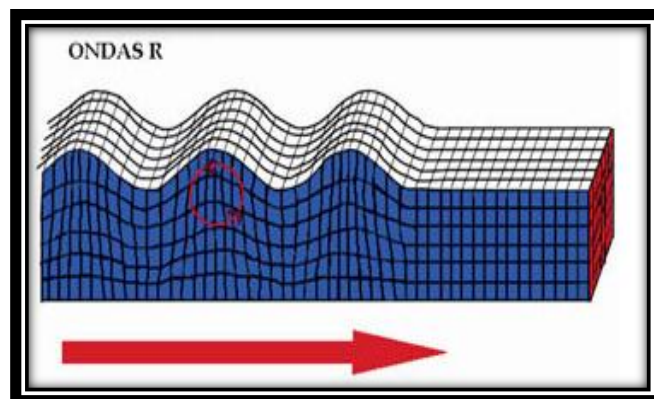


Figura 1-4 Propagación de la onda R

Fuente: (www.funvisis.gob.ve, 2014)

1.2.2.2 Ondas Love

Son ondas superficiales que realizan cortes horizontales en la Tierra. Estas ondas son levemente más lentas que las ondas Rayleigh

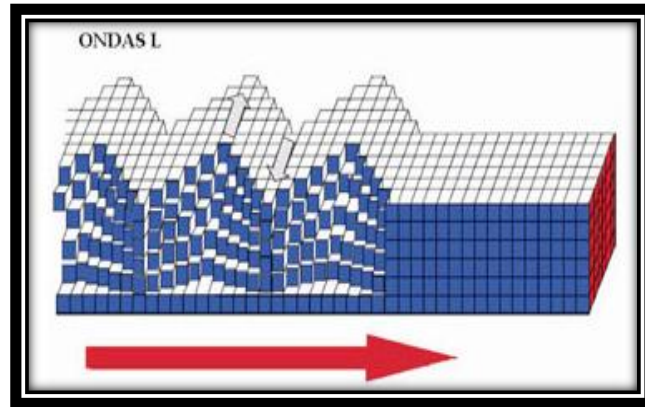


Figura 1-5 Propagación de la onda Love

Fuente: (www.funvisis.gob.ve, 2014)

1.3 Magnitud e Intensidad sísmica

1.3.1 Magnitud Sísmica

Se hace referencia a la escala propuesta por Charles Richter el mismo que observó que la amplitud sísmica tendía a disminuir en forma predecible con la distancia. El elaboró una ecuación para relacionar tanto la distancia como la amplitud de la onda S.

La escala de Richter es logarítmica, lo cual significa que si se produjeran dos sismos en un mismo sitio una distancia de 100 km con una amplitud máxima de 1 mm el primero y de 10 mm el segundo, la magnitud de los mismos sería 3 y 4 respectivamente. Si la amplitud de la traza fuera de 100 mm, entonces el sismo sería de magnitud 5. En otras palabras, un sismo de magnitud 4 es diez veces más grande que uno de magnitud 3 y un sismo de magnitud 5 sería cien veces más grande que uno de magnitud 3.

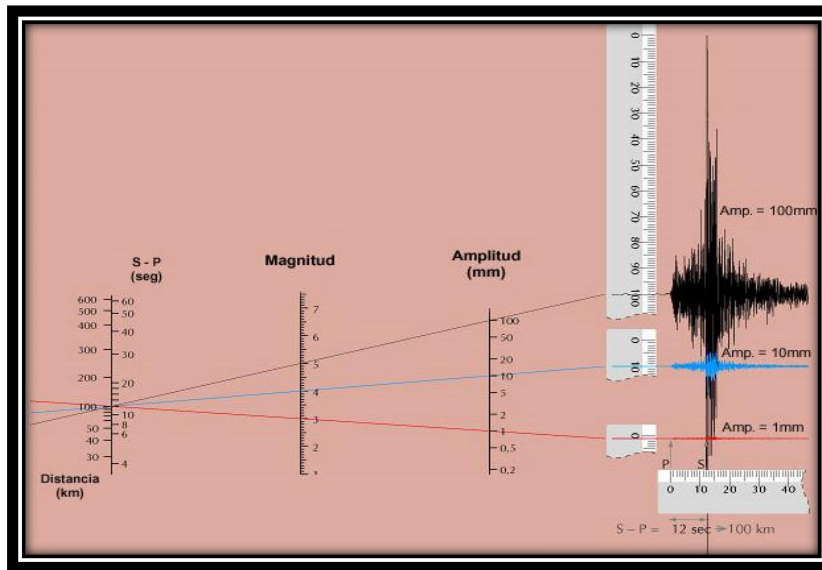


Figura 1-6 Escala de Richter

Fuente: (www.lis.ucr.ac.cr)

1.3.2 Intensidad Sísmica

Fue propuesta por Giuseppe Mercalli, a diferencia de la magnitud esta se relaciona con los efectos sobre las personas y el grado de daño sobre las estructuras. Esta escala consta de 12 grados.

Escala de Mercalli	Aceleración sísmica (g)	Percepción del temblor	Potencial de daño
I	< 0.0017	No apreciable	Ninguno
II-III	0.0017 – 0.014	Muy leve	Ninguno
IV	0.014 – 0.039	Leve	Ninguno
V	0.039 – 0.092	Moderado	Muy leve
VI	0.092 – 0.18	Fuerte	Leve
VII	0.18 – 0.34	Muy fuerte	Moderado
VIII	0.34 – 0.65	Severo	Moderado a fuerte
IX	0.65 – 1.24	Violento	Fuerte
X+	> 1.24	Extremo	Muy fuerte

Figura 1-7 Escala de Mercalli

Fuente: (www.lis.ucr.ac.cr)

1.4 Actividad Sísmica en el Ecuador

Según datos de sismos ocurridos en el Ecuador desde 1901 a 1981 que aparecen en el catálogo CERESIS. Ciento veinticinco sismos de magnitud $M_s > 4$, se han producido en la plataforma y en el continente, el 65% de todos los hipocentros fueron superficiales, es decir, que se produjeron a profundidades menores de 70 Km.

En el Ecuador se distinguen dos ambientes generadores de epicentros. El primero de ellos está formado por el borde de consumo de la Placa de Nazca con la sudamericana y por el posible contacto con la Placa Coco al norte de Ecuador. El segundo ambiente generador de sismos en el interior del país, se ubica en la región interandina y se debe a las fallas tectónicas activas en el sector.

Al interior del país han ocurrido terremotos fuertes como el sismo del 5 de marzo de 1987 de magnitud $M_s = 6.5$, que causó la muerte de unas 1000 personas y daños por un monto de más U.S.\$ 3.000'000.000; otro sismo fuerte es el ocurrido el 11 de marzo de 1955, con magnitud de 6.8 en el valle interandino de la región de Ibarra, Otavalo y Cayambe; o el terremoto del 16 de agosto de 1968, que causó 4.485 muertos en Ibarra, 4.000 en Otavalo y sobre las 3.000 en Cotacachi. Otra región de sismos fuertes es la de Ambato, Pelileo, y Riobamba.

Sin embargo, los terremotos con epicentro en el interior, son excedidos en magnitud por los ocurridos en la plataforma continental, como el sismo del 31 de enero de 1906 de magnitud 8.7 en la escala de Richter frente a Esmeraldas, considerado como uno de los más fuertes del mundo.

En Ecuador el monitoreo de sismos es llevado a cabo por el Instituto Geofísico de la Escuela Politécnica Nacional, este mantiene un activo programa de monitoreo instrumental en tiempo real, que asegura la vigilancia científica permanente sobre volcanes activos y fallas tectónicas en el territorio nacional. El IGEPN entrega informes diarios de la actividad sísmica en nuestro país, así nos podemos dar cuenta que en nuestro país están ocurriendo estos eventos de manera continua en todo el territorio.

Algunos datos obtenidos hasta el mes de abril del presente año nos dan una muestra clara de que estos eventos se manifiestan en gran parte de nuestro territorio, el más reciente y devastador ocurrió el 16 de abril del 2016, a las 18h58 con epicentro entre las parroquias de Pedernales y Cojmies en la provincia de Manabí, a una profundidad de 20 km. y una magnitud de 7.8 grados en la escala de Richter, y fue sentido en todo el Ecuador, hasta llegar a países vecinos como Colombia y Perú.

1.5 Redes de sensores inalámbricos

Las redes de sensores inalámbricos WSN (wireless sensor networks por sus siglas en inglés) se basan en la interacción de una serie de pequeños dispositivos electrónicos dispersos dentro de un área geográfica que pueden obtener información del medio y transmitirla a los nodos cercanos hasta llegar al nodo principal (Gateway). Estos nodos son comúnmente llamados “motes” debido a su traducción desde el inglés que significaría “pequeña partícula de polvo” por sus dimensiones reducidas y la idea de que pueden ser fácilmente ubicadas en cualquier sitio.

Los motes brindan mucha facilidad para su instalación y funcionamiento debido a dos razones. La primera de ellas es que al ser dispositivos autónomos poseen baterías. La segunda es que las comunicaciones son realizadas por protocolos de bajo consumo lo que permite pasar de estados inactividad a realizar comunicaciones de manera rápida minimizando el gasto de energía. (D. Gascon, 2010)

Las tecnologías de redes inalámbricas han tenido un rápido desarrollo en los últimos años. Hemos pasado de los veteranos infrarrojo (Irda) para comunicaciones punto a punto a las WPAN de corto alcance y multipuntos como “Bluetooth” o las redes de rango de alcance medio multisaltos como “ZigBee”. Otras tecnologías inalámbricas que podemos nombrar son, la tecnología WIFI para redes locales (WLAN), la tecnología “WIMAX” para redes WMAN. También la telefonía celular de largo alcance (GPRS) o el desarrollo de las comunicaciones M2M con tecnología inalámbrica.

El desarrollo más interesante de las WSN, debido a sus múltiples aplicaciones, en distintos sectores (seguridad, medio ambiente, industria, agricultura etc.). Los principales analistas tecnológicos, dentro de las tecnologías inalámbricas, valoran las redes inalámbricas de sensores (WSN) como una de las opciones de futuro más prometedora. Fabricantes como Microsoft, Intel, IBM, Motorola y Texas Instruments, por citar algunos, han lanzado líneas de investigación en esta tecnología.

1.5.1 Características de las redes de sensores

Una red de sensores inalámbrica debe cumplir tareas primordiales como la detección o recolección de datos, la comunicación de estos hacia los nodos principales, y finalmente el procesamiento de estos datos para obtener información útil. Existen diversas características que son utilizadas para evaluar una red de sensores inalámbrica. Entre los factores más importantes es posible citar:

- **Escalabilidad:** Debe poder soportar un gran número de nodos e incrementarlos sin problemas para aumentar la cobertura de la red, e incluso operar sin problemas a pesar de una posible alta densidad de nodos.
- **Energía:** En el caso de las WSN muchas veces deben funcionar en ambientes precarios con poca o nula supervisión humana. De manera que al funcionar con baterías, es necesario un óptimo control del consumo energético con protocolos de comunicación eficientes que permitan extender la vida útil de la red.
- **Auto-configuración:** Cuando existan nodos nuevos deben integrarse fácilmente y adaptarse a la red automáticamente
- **Tolerancia a fallos:** Muchas veces los nodos pueden sufrir averías llegando a funcionar defectuosamente o no funcionar por completo. El fallo de unos de los nodos sensores no debe comprometer el funcionamiento del resto de la red.
- **Costo de producción:** Ciertamente el número de nodos de una WSN puede llegar a ser realmente muy numeroso, por tanto es imprescindible que el costo individual de los nodos sea el menor posible para tener una red rentable.

1.5.2 Arquitectura de las redes de sensores

Hay varias arquitecturas que pueden ser usadas para implementar una aplicación de WSN como pueden ser: estrella, malla, árbol, que es una híbrida entre ellas. Cada topología presenta desafíos, ventajas y desventajas. La topología se refiere a la configuración de los componentes (hardware), y como los datos son transmitidos a través de esa configuración. Cada topología es apropiada bajo ciertas circunstancias y puede ser inapropiada en otras.

Para entender las diferentes topologías es necesario conocer los diferentes componentes de la WSN.

- **Nodos Sensores o Motes:** Su función es la de registrar datos del medio en el cual está siendo usado.

- Estación base o coordinador: Es el dispositivo más completo, debe existir uno por red pues su única función de formar una red. Es el responsable de establecer el canal de comunicaciones y del PAN ID (identificador de red) para toda la red.

Una vez establecidos estos parámetros, el coordinador puede formar una red, permitiendo unirse a él dispositivos finales. Una vez formada la red, el coordinador hace las funciones de ruteador, participa en el enrutamiento de paquetes y es origen y/o destinatario de información.

- Puertas de enlace: Recoge los datos de la red y sirve como punto de unión con una red LAN o con Internet.

En cuanto a la estructura de las topologías se refiere a la configuración de ubicación y comunicación de los elementos de hardware y como se transmiten los datos a través de esa configuración. Cada topología es conveniente bajo ciertas circunstancias, pero puede ser inadecuada en otras. Entre las principales destacan:

- Topología en estrella: Se caracteriza porque todos los nodos se conectan directamente con el Gateway, que es el centro de la red. Los nodos no se comunican unos con otros directamente, sino solo con el Gateway que transmite los datos fuera de la red y trabaja como intermediario en caso de que los nodos necesiten intercambiar información entre ellos. La comunicación es monosalto, es decir, la información solo necesita un salto para llegar a su destino, no existen intermediarios. Por lo general la distancia entre los nodos y el Gateway va desde 30m a 100m, por lo que su latencia es muy baja. Tiene poca tolerancia a fallas, porque si se cae el Gateway deja de funcionar toda la red.

Las ventajas de ésta topologías son:

- Topología muy robusta.
- Baja latencia.
- Sencillez y rapidez al desarrollarlas.
- Fácil de desplegar.
- Fiabilidad muy alta.
- Gasto de energía uniforme para todos los nodos.
- Bajo costo.

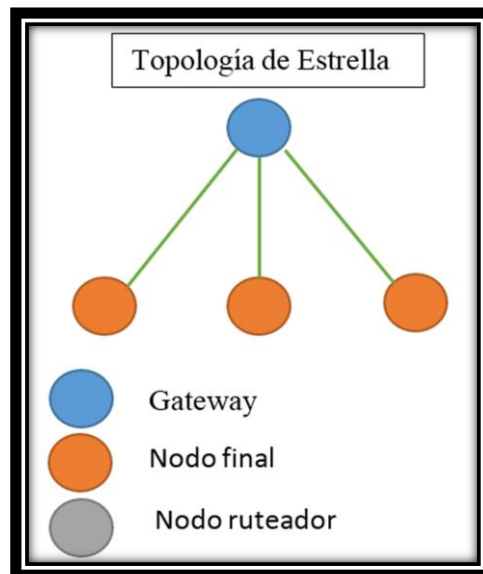


Figura 1-8 Topología WSN en estrella

Realizado por: (Garcés A., Demera S., 2016)

- Topología de árbol: Es una topología multisalto donde existen nodos enrutadores entre el Gateway y los nodos finales, los cuales hacen de intermediarios para aumentar la cobertura de la red. La función de los enrutadores es escuchar tanto al Gateway como a los nodos finales y encaminar los datos. Se le llama de árbol porque el Gateway haría la función de raíz, los nodos finales de hojas y los enrutadores de ramas para comunicar ambos extremos. Es una topología con alta tolerancia a fallos, pero si caen los enrutadores se pierde parte de la red.

Las ventajas de ésta topologías son:

- Topología muy robusta.
- Baja latencia.
- Poca dificultad para desplegar (aumenta en relación al tamaño la red).
- Fiabilidad muy alta.
- Alta escalabilidad.
- Porcentaje de colisiones bajo.

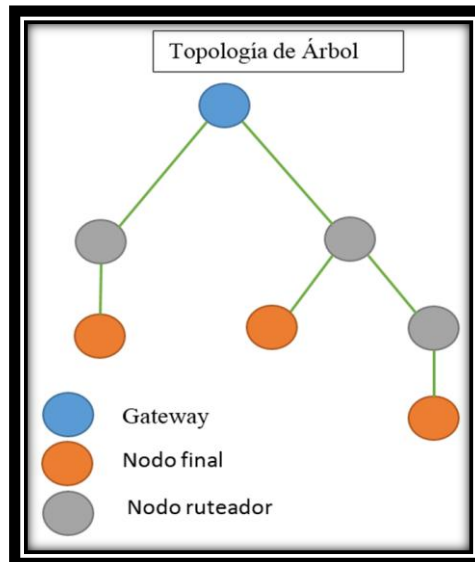


Figura 1-9 Topología WSN en árbol

Realizado por: (Garcés A., Demera S., 2016)

- Topología de malla (Mesh): Al igual que la topología de árbol, cuenta con enrutadores que comunican nodos finales con el Gateway, pero se diferencian porque estos enrutadores pueden interconectarse entre ellos lo cual aumenta la redundancia y la cobertura ya que se vuelve en teoría ilimitada. Tiene alta tolerancia a fallas porque al fallar cualquier enrutador los enlaces redundantes permiten mantener comunicados los nodos finales y el Gateway a, costo de una latencia más elevada.

Las ventajas de ésta topologías son:

- Alta escalabilidad.
- Alta tolerancia a fallas.
- Menor costo en relación a la topología de árbol para redes extensas.
- Fiabilidad muy alta.

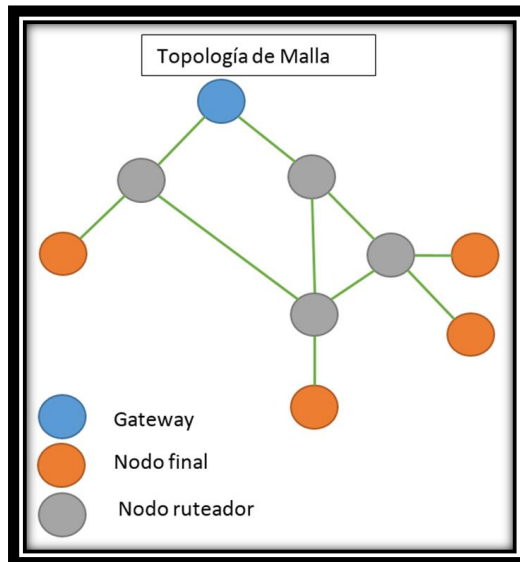


Figura 1-10 Topología WSN en malla

Realizado por: (Garcés A., Demera S., 2016)

1.5.3 Arquitectura de los nodos

Las motas dotan de procesamiento y de comunicación al sensor. Los procesadores de radio, toman los datos del sensor a través de sus puertas de datos, y envían la información a la estación base. Los componentes típicos son:

- Baterías
- Una CPU.
- Memoria externa
- Una placa de sensores: luz, humedad, presión, etc.
- Radio para comunicar con otras motas.

Son resistentes a la intemperie y terrenos inhóspitos y capaces de ejecutar una aplicación

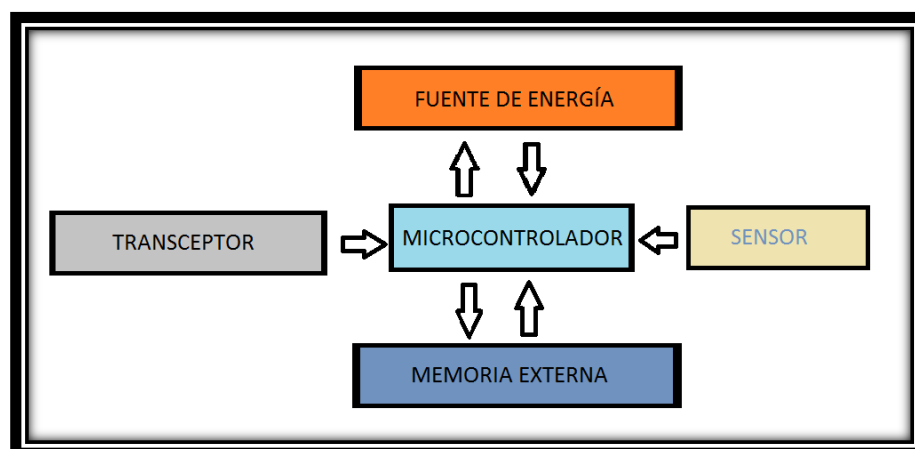


Figura 1-11 Arquitectura del nodo

Realizado por: (Garcés A., Demera S., 2016)

1.5.4 Comunicaciones M2M

El concepto M2M (máquina a máquina) la capacidad de intercambiar datos entre dos máquinas remotas, de forma que, mediante este intercambio, es posible controlar y supervisar de forma automática procesos en los que intervienen máquinas. El foco principal de aplicación de M2M se ubica en los entornos relacionados con la telemetría y el telecontrol.

1.5.5 Comunicaciones SWAP

SWAP (Protocolo Simple Abstracto Inalámbrico) es un protocolo compacto y ligero diseñado para ser utilizado con dispositivos de radio basados en el microcontrolador CC11XX de Texas Instruments. SWAP se centra principalmente en estos chips porque se basa en su estructura de paquetes, pero este protocolo se puede adaptar a otras estructuras de paquete (y por lo tanto a otros CI de) con poco trabajo.

SWAP se puede utilizar bajo dos diferentes esquemas de direccionamiento: direcciones de un solo byte y direcciones de 2 bytes. El byte función le dirá a la aplicación que el esquema se utiliza en cada trama.

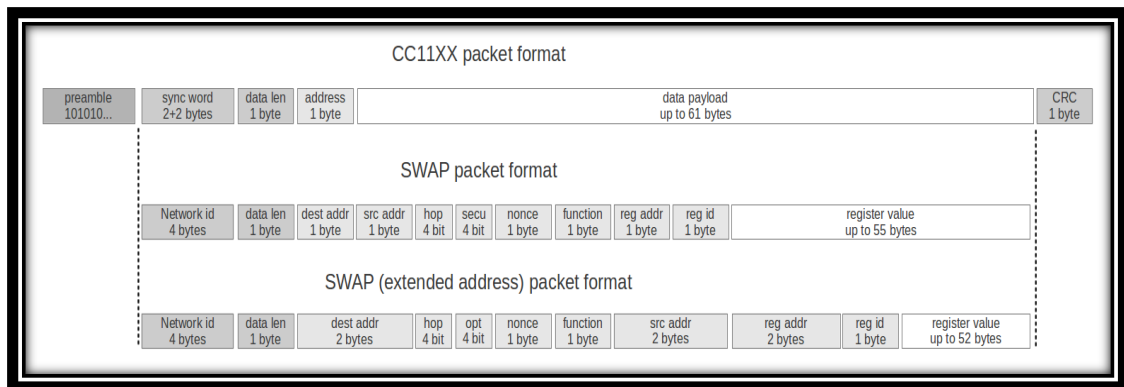


Figura 1-12 Estructura de paquetes SWAP

Fuente: (www.panstamp.com,2016)

1.6 Panstamp

Son módulos inalámbricos de baja potencia autónomos programables desde el IDE de Arduino dirigidos a proyectos de telemetría y control. Estos módulos se comunican a través de las bandas 868-900-915 MHz y están diseñados para durar meses e incluso años cuando un par de pilas alcalinas simples, dependiendo del ciclo de trabajo y el intervalo de la transmisión programada.

Además, pueden formar ecosistemas inalámbricos completos con conectividad directa a Internet. La compañía ofrece aplicaciones de *software* para la configuración de redes inalámbricas y convertir cualquier ordenador en un servidor de automatización.

1.6.1 *Panstamp NRG 2.0*

Es el módulo inalámbrico más potente del Panstamp. Basado en el popular SoC (system-on-chip) CC430F5137, proporciona puertos de entradas de 12 bits, digitales, analógicos, reloj en tiempo real y muchas otras herramientas útiles para proyectos de telemetría y control inalámbricos. La mayor parte de los aspectos críticos de NRG (radio, administración de energía, RTC, etc.)

Están cubiertos por las funciones básicas Panstamp y funciones únicas genéricas como UART, SPI, I2C y gestión de E / S se delegan en Arduino con el fin de garantizan la compatibilidad. (ver Anexo A)

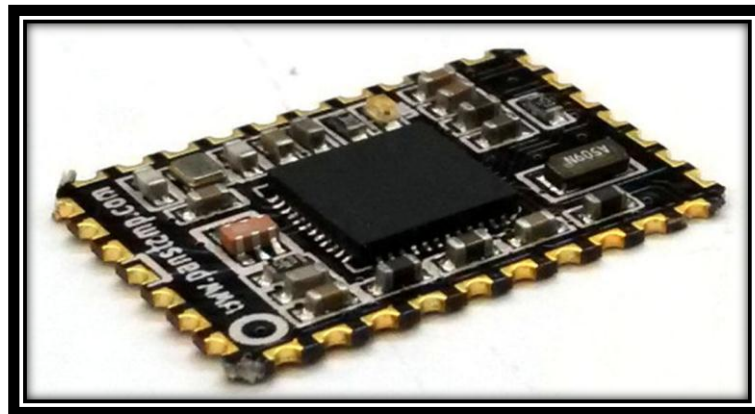


Figura 1-13 Modulo Panstamp NRG 2.0

Fuente: (www.panstamp.com,2016)

Especificaciones

- Dimensiones: 0,63 x 0,86 en (16 x 22 mm)
- MCU (Memory controller unit): CC430F5137 (MSP430 de radio CC11XX núcleo + SOC)
- Velocidad: la velocidad programable entre 8 MHz y 24 MHz
- Infilamación: 32 KB
- RAM: 4 KB
- Información de la memoria: 512 bytes
- Seis entradas ADC de 12 bits

- dirección MAC única
- Rango de tensión: desde 2VCC a 3.6VDC
- Rx de corriente: 18 mA máx
- Tx corriente: 36 mA máx
- corriente de reposo: 1-2 uA
- Máxima potencia de transmisión: +12 dBm
- bandas de RF: 433/868/905/915/918 MHz ISM
- Comunicación longitud: 200 m en espacios abiertos a 0 dBm con cable de antena
- 128 bits de cifrado AES de Seguridad
- A bordo del LED
- Programable a través de SBW, BSL serie y de forma inalámbrica (SWAP)

1.6.2 Panstick 4.0

Es un Gateway USB-UART especial diseñado para panStamp 2. Proporciona la huella necesaria para incluir un panStamp (NRG o AVR) y también se puede utilizar sin un módulo inalámbrico, apilando o enchufando bajo o sobre otras carrier boards. Este Gateway puede ser utilizado tanto como un programador serial, así como como un módem serial para redes inalámbricas.

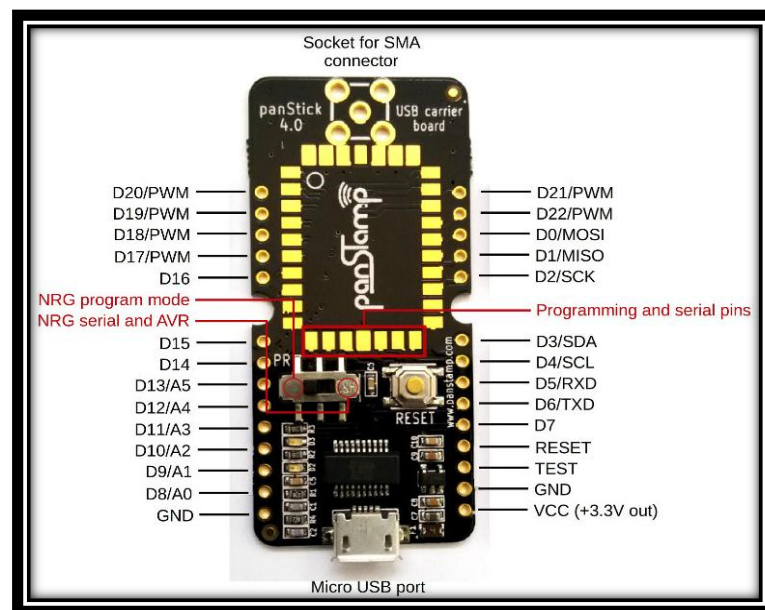


Figura 1-14 Modulo Panstick 4.0

Fuente: (www.panstamp.com,2016)

Basta con colocar el panstamp sobre el panstick y conectarlo a un puerto USB para poder cargar nuevos programas desde el entorno de programación de Arduino. (ver Anexo B)

Especificaciones

- Tamaño: 19 x 45 mm
- USB/UART CONVERTER: FTDI FT232 RL
- Supported OS: Linux, MacOS y Windows

1.6.3 Tablero de minibat ntc

Es un panel de soporte de baterías diseñado para albergar un panStamp y uno o varios sensores, y es compatible con los pines de otras placas mini-portadoras realizadas por la empresa.

Cuenta con 37 terminales, un socket para conector SMA, pista para la colocación del Panstamp, sensor acelerómetro/magnetómetro/giroscopio, sensor de presión/temperatura, sensor de humedad/temperatura y termistor, además de un botón de reset (ver Anexo C).

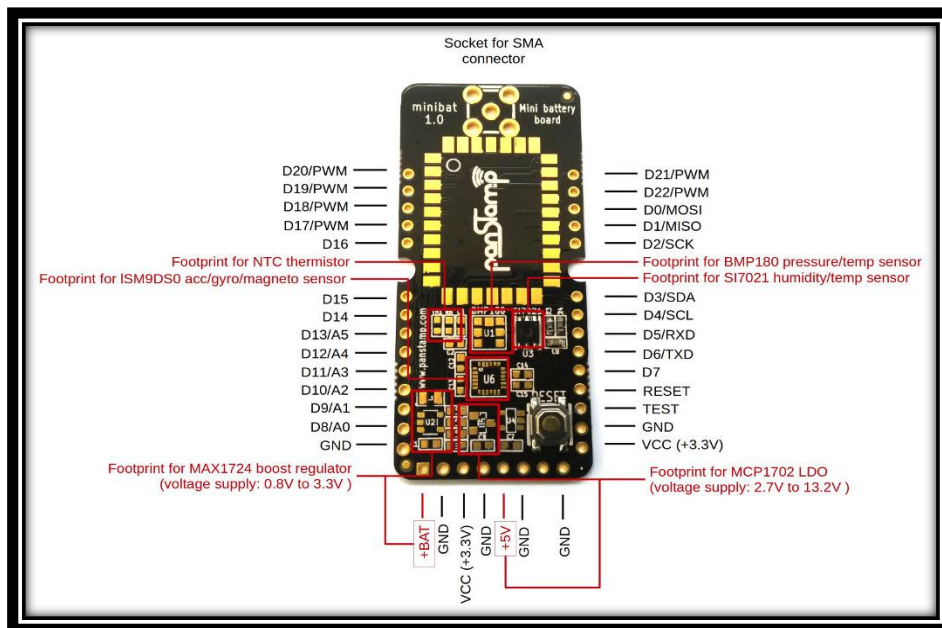


Figura 1-15 Minibat

Fuente: (www.panstamp.com,2016)

1.7 Sensor Acelerómetro MPU6050

El sensor InvenSense MPU-6050 contiene un acelerómetro MEMS y un giroscopio MEMS en un solo chip. Es muy preciso, ya que contiene 16 bits de conversión analógica a digital de hardware para cada canal. Para ello captura los canales x, y, z al mismo tiempo. El sensor utiliza el bus I2C para interconectarse con el Arduino. (ver Anexo D)

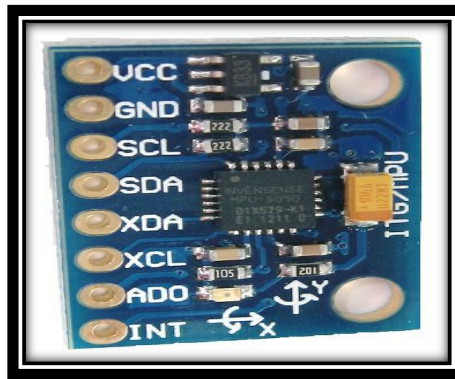


Figura 1-16 Minibat

Fuente: (www.cdiweb.com, 2016)

Especificaciones

- Regulador de voltaje de operación LDO a bordo: 3V a 5V
- Comunicación: I2C
- Convertidor de 16 bits AD Incorporado, la salida de datos de 16 bits
- Rango de los giroscopios: +/- 250 500 1000 2000 grados / segundo
- Variación de aceleración: +/- 2g, +/-4g, +/- 8 g, +/-16 g
- Distancia entre ejes de conectores: 2,54 mm (0,1 pulg)

1.8 GPRS Shield V2.0

Es una tarjeta GPRS de comunicación inalámbrica ultra compacta. La tarjeta es compatible con todos los modelos de Arduino, además puede ser controlada con otros microcontroladores. La tarjeta está basada en el módulo SIM900 4.

La tarjeta GPRS permite la configuración por vía UART usando comandos AT. Solo es necesario conectar la tarjeta al microcontrolador, Arduino, etc, y empieza la comunicación a través de comandos AT. Ideal para sistemas remotos, comunicación recursiva, puntos de control, envío de mensajes de texto a celulares, etc. (ver Anexo E)

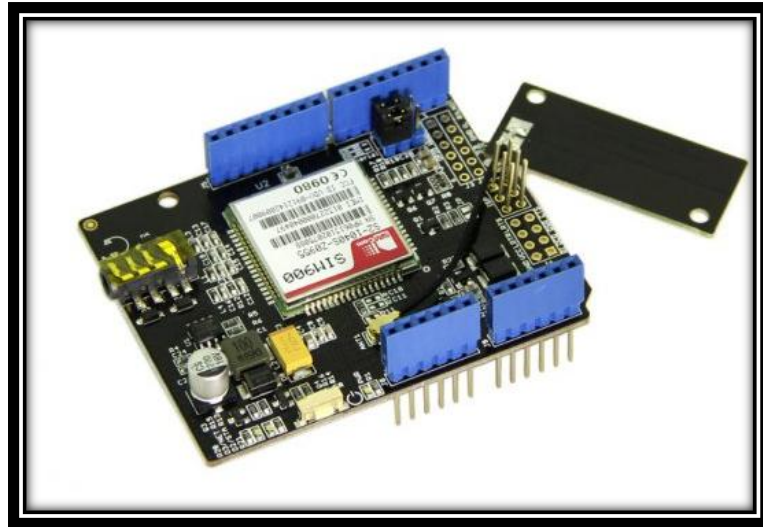


Figura 1-17 Shield GPRS

Fuente: (www.instructables.com, 2016)

1.9 Arduino Mega

Es una placa de desarrollo basada en el ATmega1280. Cuenta con 54 pines digitales de entrada / salida (de los cuales 14 se pueden ser utilizados como salidas PWM), 16 entradas analógicas, 4 UARTs (puertos serie de hardware), un oscilador de 16MHz, una conexión USB, un conector de alimentación, una cabecera ICSP, y un botón de reinicio.

Contiene lo necesario para apoyar el microcontrolador; basta con conectarlo a un ordenador con un cable USB, con un adaptador de AC a CC o a una batería. Arduino Mega es compatible con la mayoría de los shield para el Arduino. (ver Anexo F)

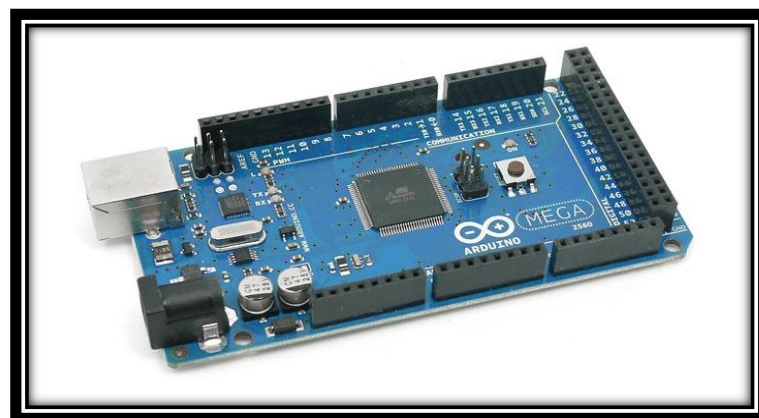


Figura 1-18 Modulo Arduino Mega 2560

Fuente: (www.arduino.cc, 2016)

CAPITULO II

2. MARCO METODOLÓGICO

En este capítulo se realizó el proceso de selección de los componentes necesarios para la implementación del sistema, tomando en cuenta sus características más relevantes y haciendo una valoración de cada una respecto a variables cuantitativas y cualitativas para determinar su idoneidad.

Para el desarrollo del proyecto se utilizó el método comparativo para poder determinar los dispositivos adecuados para el desarrollo del mismo, así evaluar diferentes alternativas que existen en el mercado y optar por los dispositivos más convenientes en este caso de estudio, la clasificación de sus características dejara una idea más clara de los dispositivos que existen y cuáles serán de nuestra elección.

Con el método experimental se realizaron mediciones de variables en la red SWAP tales como: la cobertura, aceleraciones y consumo energético, todo esto nos permitirán saber que los nodos estarán funcionando de forma eficiente.

Los dispositivos fueron ubicados en las instalaciones de la ESPOCH, para facilitar la medición de las variables necesarias para el desarrollo del proyecto. Los nodos inalámbricos estarán ubicados en espacios abiertos, el recurso energético es de suma importancia, cada nodo tendrá su fuente de alimentación propia.

2.1 Comparación de sensores

Para determinar los sensores a utilizar en la implementación del proyecto se tomó en cuenta algunos factores como: costos de los dispositivos, tamaño, consumo energético, alimentación, comunicación, memoria RAM y el tipo de CPU.

En base a estos factores se realizó un análisis entre cuatro posibles dispositivos que existen en el mercado que nos serian de utilidad a la hora de llevar a cabo la implementación.

Para seleccionar los dispositivos idóneos que cubran las necesidades que presenta la red de sensores inalámbrica se realizó una tabla de las características desde la de mayor a menor

importancia (ponderación) y así dar valores cuantitativos del 0 (nada) al 4 (máximo); el análisis se realizó para cada uno de los tipos de dispositivos propuestos con respecto a cada característica.

Tabla 2-1 Escala cuantitativa

VALORACIÓN				
0	1	2	3	4
0	25%	50%	75%	100%

Realizado por: Garcés A, Demera S, 2016

Tabla 2-2 Escala cualitativa

4	3	2	1	0
Muy adecuado	Adecuado	Poco adecuado	Inadecuado	Nada adecuado

Realizado por: Garcés A, Demera S, 2016

2.1.1 Sensores para medir aceleración

Tabla 2-3 Sensores de aceleración disponibles en el mercado

Características	ADXL210	MPU6050	MXR2312G/M
Rango de aceleración	±10 g	±2g/±6g	±2.0 g
Ancho de banda	2,5kHz	100Hz	17Hz
Sostenibilidad	40% /g	2g=Vdd/5 V/g	312 mV/g
Voltaje de alimentación	3 V - 6 V	2.5V – 3.6V	3,00 V - 5,25 V
Fabricante	ANALOG	INVENSENSE	MENSIC
Inversión	Alta	Baja	Alta
Aplicaciones	-Industriales -Biodinámica -Automoción	-Airbag -Uso automoción -Vibraciones.	-Industrial -Sistema alarma -Uso general
Package	E-8	No dispone	LCC8

Tabla 2-4 Valoración de las características para los sensores de aceleración

Nivel de Importancia	Característica	ADXL210	MPU6050	MXR2312G/M
1	Rango de aceleración	4	4	3
2	Ancho de banda	0	4	0
3	Sostenibilidad	3	4	4
4	Voltaje de alimentación	2	3	3
5	Fabricante	3	4	4
6	Inversión	2	4	2
7	Aplicaciones	4	4	3
8	Package	4	4	3
TOTAL DE PUNTOS:		22	31	22

Realizado por: Garcés A, Demera S, 2016

Interpretación: Se ha elegido el sensor de aceleración modelo MPU6050 que desarrolla y distribuye la empresa *Invensense* (Alternativa 2) porque de acuerdo a la tabla 2-5 esta opción adquiere mayor porcentaje de 96,88%, y en la escala cualitativa se encuentra entre el rango de adecuado - muy adecuado

Tabla 2-5 Análisis de las alternativas de sensores de aceleración

Puntos máximos	Alternativa 1	Alternativa 2	Alternativa 3
32	22	31	22
100%	68,75%	96,88%	68,75%

Realizado por: Garcés A, Demera S, 2016

El sensor de aceleración se conecta a la board de desarrollo *PANSTAMP* y se encarga de tomar valores de aceleraciones para enviarlos de forma inalámbrica al nodo principal *panstick*, para posteriormente ser monitorizados en un computador.

2.1.2 Hardware para desarrollar la comunicación

Tabla 2-6 Placas de desarrollo disponibles en el mercado

Características	Arduino	Raspberry Pi	Panstamp
CPU	ATmega1280	Quad-core ARM Cortex-A7	MSP430
RAM	8 kb	1Gb	4kb
Comunicación	Red Celular (GSM)	Ethernet 10/100	Inalámbrica
Consumo	5V	800 mA/4w/5V	1 – 2 uA
Tamaño	101.52 x 53.3 mm	85 x 56 mm	17.7 x 30.5mm
Alimentación	7V – 12V	MicroUSB / GPIO	2 – 3.6 DC
Costo	\$75	\$ 85	\$ 25,68

Realizado por: Garcés A, Demera S, 2016

Tabla 2-7 Valoración de las características de las placas de desarrollo

Nivel de Importancia	Característica	Arduino	Raspberry Pi	Panstamp
1	CPU	3	4	4
2	RAM	4	0	4
3	Comunicación	3	3	4
4	Consumo	3	2	4
5	Tamaño	4	2	4
6	Alimentación	4	2	3
7	Costo	4	4	4
TOTAL DE PUNTOS:		25	17	27

Realizado por: Garcés A, Demera S, 2016

Interpretación: Se ha elegido *Panstamp* que fabrica y distribuye la empresa *PANSTAMP*, (Alternativa 3) porque es el que mayor porcentaje obtiene de acuerdo a sus características como podemos observar en la tabla 2-8 con un 96,43%, y de acuerdo a la escala cualitativa se encuentra entre el rango de adecuado - muy adecuado; es importante señalar que el tamaño hace posible una mejor manipulación de los mismos.

Tabla 2-8 Porcentaje de las alternativas para la comunicación

Puntos máximos	Alternativa 1	Alternativa 2	Alternativa 3
28	25	17	27
100%	89,29%	60,71%	96,43%

Realizado por: Garcés A, Demera S, 2016

Esta placa de desarrollo es programable con el Ide de Arduino, lo que nos permite utilizar un entorno amigable y conocido, además de su costo inferior al de los equipos Arduino.

Dispositivos adicionales:

➤ Case

Cerramiento empotrable para PanStamp resistente al agua. En el cabe perfectamente un contenedor de baterías AAA. Dimensiones: 57x51x28 mm (contorno interno irregular)



Figura 2-1 Vista frontal del case

Fuente: (www.panstamp.com, 2016)

➤ Conector SMA

Es compatible con completamente todos los tipos de antenas SMA.



Figura 2-2 Conector SMA

Fuente: (www.panstamp.com, 2016)

➤ Antenas SMA articulada

Antena articulada tipo dipolo de $1/4$ de longitud de onda con conector tipo SMA. Están diseñadas para funcionar en las bandas ISM de 868 MHz y 915 MHz.



Figura 2-3 SMA Articulada

Fuente: (www.panstamp.com, 2016)

➤ Antena SMA

Antena de $1/4$ de longitud de onda con conector SMA y base magnética. Esta antena proporciona 2m de cable entre el conector y la antena. Funcionan en las bandas ISM de 868 MHz y 915 MHz.



Figura 2-4 Antena SMA de base magnetica

Fuente: (www.panstamp.com,2016)

➤ Contenedor de batería

Es una gran opción para proveer energía a los módulos PanStamp y permitirle al nodo inalámbrico tener autonomía.

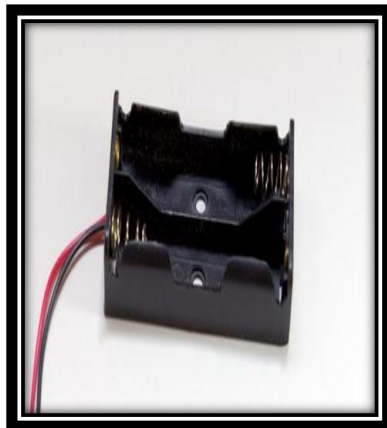


Figura 2-5 Battery holder

Fuente: (www.panstamp.com,2016)

CAPITULO III

3. MARCO DE RESULTADOS

Para determinar los resultados fue necesaria la implementación del prototipo funcional de la red de sensores inalámbrica. Se realizaron pruebas de funcionamiento para establecer las ventajas y dificultades del presente trabajo.

3.1 Diseño del Prototipo de la Red Inalámbrica

Una vez realizado el estudio comparativo entre los dispositivos a usarse, se procedió con el diseño del prototipo funcional que por la ubicación de los nodos panstamp se utilizó una topología mono-salto en estrella que se puede observar en la fig 3-1.

Cada nodo fue equipado con un sensor MPU6050 para realizar las mediciones de aceleraciones, estos se conectaron a un circuito embebido Panstamp que se encarga de enviarlos al a través de la red swap Gateway de ser necesario.

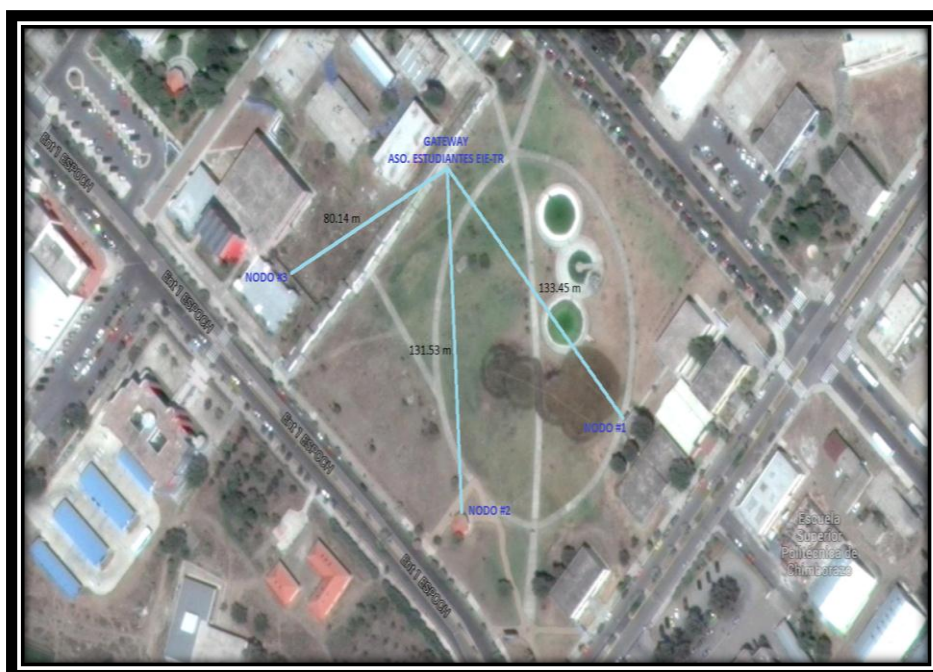


Figura 3-1 Topología Estrella en base a la ubicación de los tanques

Fuente: Google Earth

El Gateway panstick se encargará de recibir la información por medio de comunicación inalámbrica y enviarla al módulo arduino, que por medio de puerto serial se visualizará en un computador para su monitorización. Dependiendo la intensidad del movimiento sísmico el módulo arduino envía SMS's de alerta a través del shield GPRS.

Los nodos Panstamp fueron ubicados a diferentes distancias para evitar que sean afectados por un evento no sísmico y se ejecute la alarma por equivocación. El Gateway Panstick fue ubicado en la oficina de la Asociación de Escuela EIE-TR ESPOCH por su ubicación estratégica, que permite tener línea de vista con los nodos, el acceso a un ordenador como estación de monitoreo y una fuente de energía estable.

3.1.1 Diagrama de bloques de la red

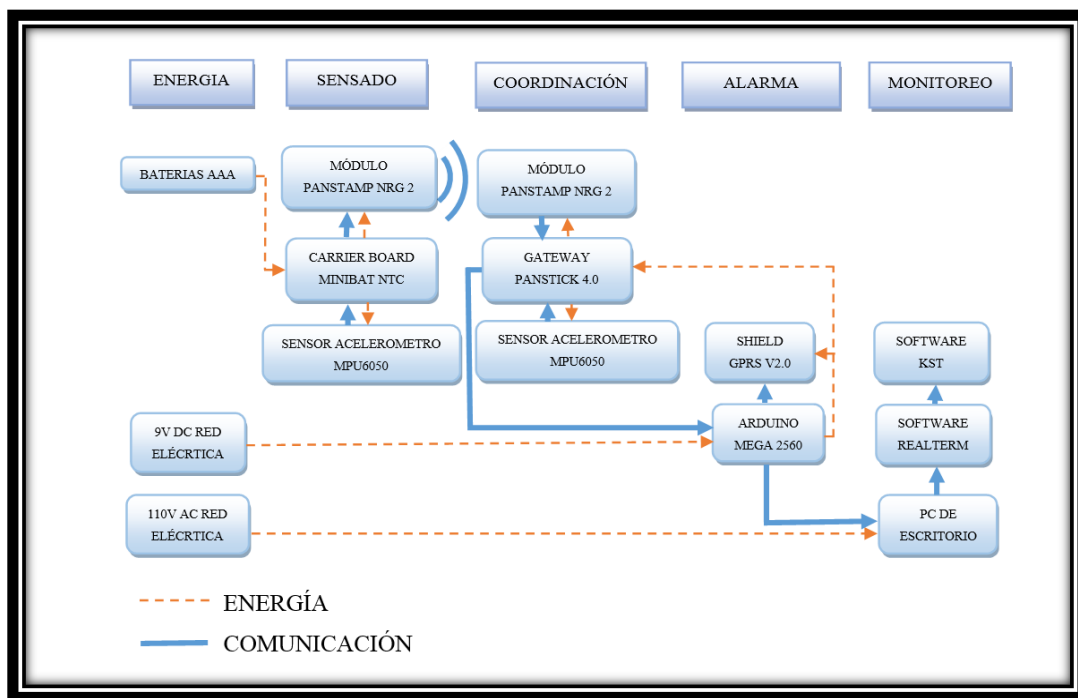


Figura 3-2 Diagrama de bloques del prototipo de la WSN

Realizado por: Garcés A, Demera S, 2016

3.1.2 Diagrama de flujo de la red

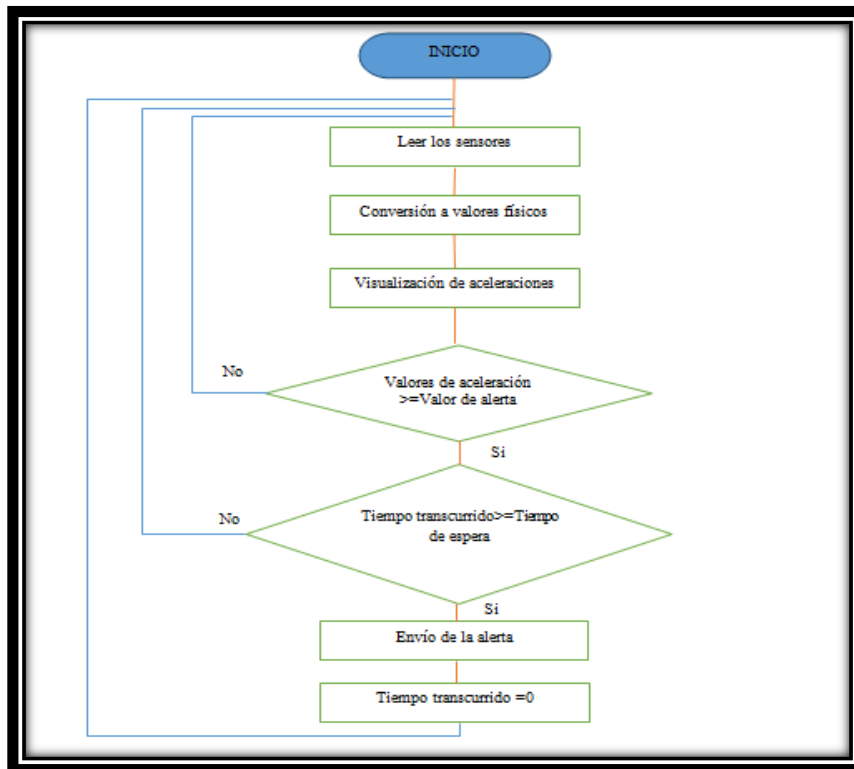


Figura 3-3 Diagrama de flujo del prototipo de la WSN

Realizado por: (Garcés A, Demera S, 2016)

3.2 Configuración de la Red Inalámbrica

3.2.1 Configuración de hardware

Incluye cada módulo, se explica detalladamente las funciones que realiza y los procedimientos efectuados para ponerlos en marcha.

3.2.1.1 Configuración del Gateway Panstick

La función básica del Panstick es estar a la espera de los datos que puedan ser enviados desde los nodos, leer el sensor que posee, etiquetar los datos y enviarlos al sistema de alerta. Estos datos no son convertidos a unidades físicas, sino que mantienen el formato que da el sensor para minimizar el uso del procesador.

Otra tarea importante asignada al Gateway es establecer la comunicación cuando los nodos inician y guardar las ID's de cada uno para identificar que datos corresponden a cada nodo. Esta operación solo se realiza la primera vez que los nodos entran en funcionamiento.

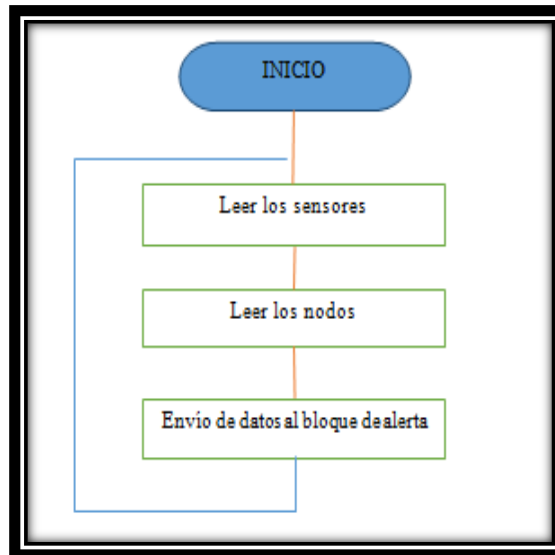


Figura 3-4 Diagrama de flujo del Panstick

Realizado por: (Garcés A, Demera S, 2016)

Lo primero antes de estar listo para ser configurado es soldar el panstamp sobre la huella que le corresponde, y los conectores para configurar a los nodos, ya que vienen separados. Luego es necesario realizar la programación para el funcionamiento deseado, lo cual se puede realizar en el entorno de programación de Arduino. Por último, se carga el programa en el dispositivo y se comprueba su funcionamiento para solucionar cualquier anomalía.

Otro aspecto importante es que con la ayuda del panstick podemos configurar los nodos, porque estos no cuentan con un puerto micro-usb, por lo que se los debe apilar sobre el panstick para pasarles los sketches diseñados para cada uno. La programación debe hacerse con el switch colocado en la posición izquierda (en modo de programación NRG), la posición derecha es para que funcione como modem.

El programa que fue cargado en este caso tiene la función de estar siempre a la espera de alguna transmisión desde los nodos, en caso de recibirlas las identifica y clasifica al comparar el identificador que acompaña a los datos y de la misma manera los transmite inmediatamente al módulo de alerta.

3.2.1.2 Configuración del módulo Panstamp

El primer paso es preparar el dispositivo para ser programado, es decir soldar a la carrier board minibat el panstamp, los conectores, el battery holder y el sensor, con lo que el nodo está listo para ser programado y entrar en funcionamiento.

Previo a realizar la programación de los nodos es necesario indicar la url que proporciona el fabricante para añadir las características de la tarjeta Panstamp NGR, que por defecto no viene instalada en el IDE de Arduino. De igual manera se requiere descargar e instalar las librerías necesarias para programar los sketches de cada dispositivo.

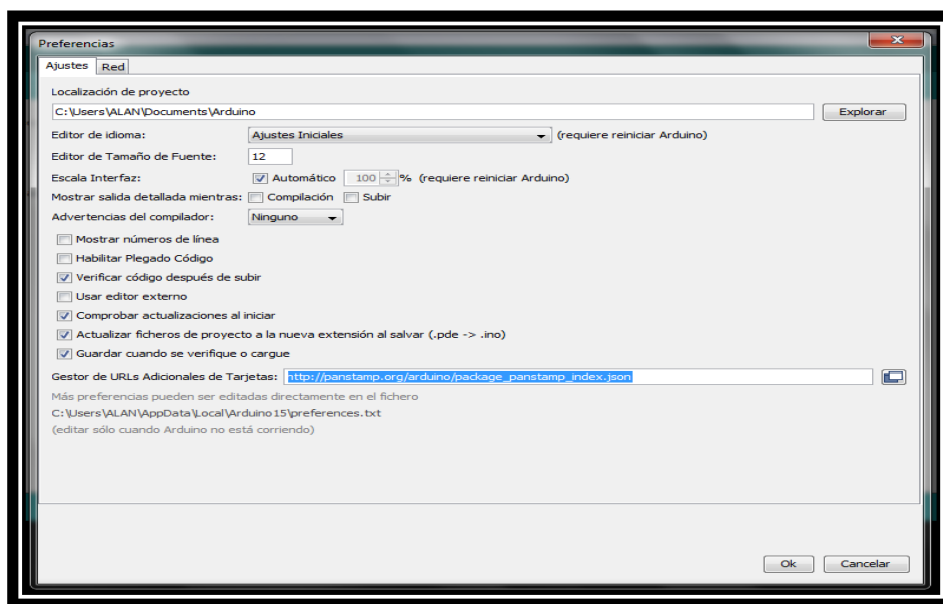


Figura 3-5 Instalación de librería panstamp en Arduino

Realizado por: Garcés A, Demera S, 2016

Para configurar cada nodo panstamp es necesario primero configurar el panstick en modo modem, seleccionar los parámetros adecuados como seleccionar la tarjeta panStamp NRG 2 w/cc430f5137 y seleccionar el puerto correspondiente al dispositivo panstick, colocar el switch en modo de programación NRG es decir en la posición izquierda, luego desde el IDE de arduino cargar el sketch “modem” proporcionado por el fabricante. Con esto solo queda conectar por apilamiento o por conexión cableada en los conectores indicados (UART RX, UART TX, GND, VCC, RST, TST) al panstamp y cargar el programa deseado.

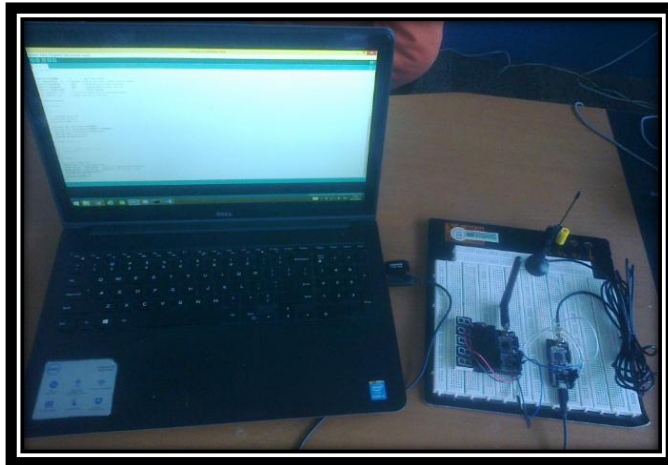


Figura 3-6 Configuración de panstamp

Realizado por: Garcés A, Demera S, 2016

El funcionamiento para los nodos especificado en la programación cumple básicamente cuatro etapas, en la primera se establece la comunicación SWAP con el Gateway donde intercambian información como la ID de red y la ID de los dispositivos, esta acción solo se realiza cuando inicia cada nodo y después de esto se deshabilita la transmisión. La segunda etapa comprende la lectura de los datos obtenidos por los sensores y es complementada con las siguientes ya que estos valores son comparados con un valor definido en 0.18g (2949 ADC) en la tercera etapa, de ser mayores, son enviados al Gateway, lo que comprende la cuarta etapa. Al igual que en el Gateway los datos no son convertidos a unidades físicas, sino que mantienen el formato que da el sensor para minimizar el uso del procesador.

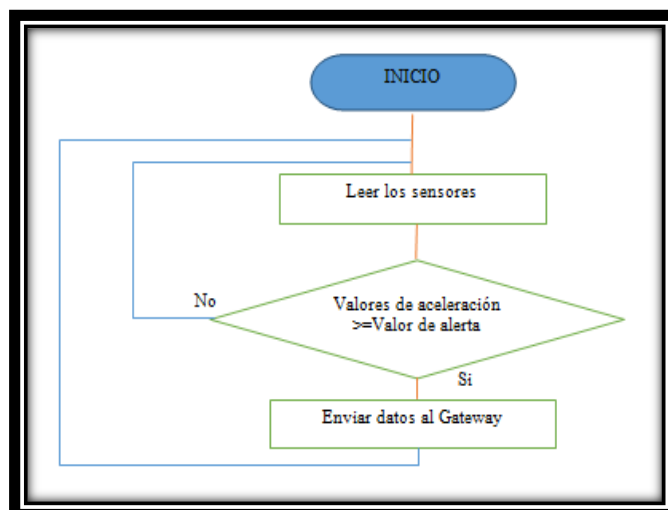


Figura 3-7 Diagrama de flujo de los nodos

Realizado por: Garcés A, Demera S, 2016

3.2.1.3 Configuración del módulo Arduino Mega

En este caso se trata del desarrollo de un sistema de monitoreo en tiempo real de aceleraciones producidas por la Tierra, la función que debe cumplir el módulo Arduino es la de convertir los datos obtenidos en unidades físicas, compararlos con el valor de referencia para el envío de la alerta y transmitir por puerto serial los datos recolectados por el Gateway a un PC.

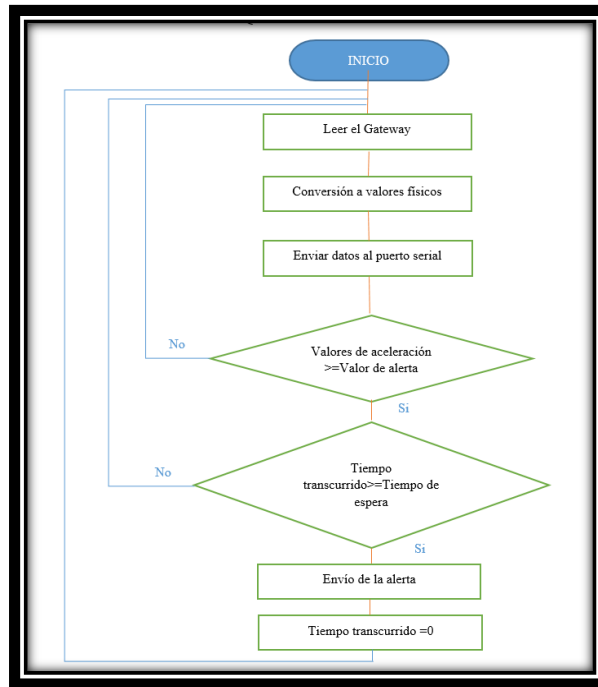


Figura 3-8 Diagrama de flujo del módulo Arduino

Realizado por: Garcés A, Demera S, 2016

El módulo Arduino se enfoca en leer los datos del Gateway, convertirlos en datos que puedan ser interpretados, enviarlos por puerto serial al pc, compararlos con el valor de referencia que fue establecido en 0.2g (equivalente a un sismo fuerte en la escala de Mercalli), en caso de ser superado este valor, pasa al siguiente filtro que es un temporizador establecido en 1 minuto, si el tiempo transcurrido desde la anterior alerta es superior al tiempo de espera, se procede con el envío de la alerta a los números que han sido establecidos. Este bucle se repite indefinidamente ya que el sistema funciona de manera permanente.

El shield GPRS trabaja en la banda de frecuencia de 850 Mhz de la red claro, debe contar con un chip que contenga saldo disponible en todo momento para que pueda enviar los SMS necesarios en el momento que ocurra un evento. El mensaje que se transmite es “Existe riesgo de movimiento sísmico, por favor tome las debidas precauciones. –SISTEMA DE ALERTA TEMPRANA DE SISMOS ESPOCH--”

Para lograr la correcta lectura de estos datos el módulo Arduino está programado para tomar los valores de cada nodo al mismo tiempo, es decir que va a listar todos los valores de aceleraciones de cada nodo transmisor estos valores son guardados en un archivo txt por el programa Realterm y visualizados por el programa Kst.

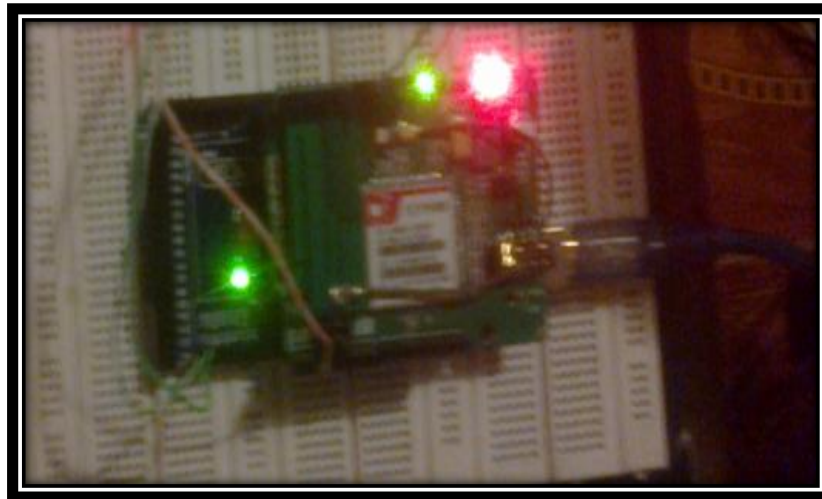


Figura 3-9 Módulo Arduino conectado al shield GPRS

Realizado por: Garcés A, Demera S, 2016

3.2.1.4 *Calibración de los sensores MPU6050*

Para que los sensores de aceleración funcionen de manera óptima se procedió a realizar la calibración de los mismos. El sensor MPU6050 tiene un Mens (MicroElectroMechanicalSystems) que de forma similar a un sistema de masa resorte permite medir la aceleración. Esto le permite estar sensando siempre la aceleración de la gravedad que es $9,8\text{m/s}^2$ equivalente a $1g$. Este valor servirá de referencia a la hora de calibrar el sensor.

El modulo se comunica por I2C, esto le permite trabajar con la mayoría de microcontroladores. Es necesario calibrar el sensor antes de hacerlo funcionar en el proyecto, porque es probable que no se encuentre en una posición horizontal al cien por ciento, esto sucede porque el sensor al ser soldado en el módulo puede estar desnivelado agregando un error en cada componente. La configuración se la realiza en OFFSETS del módulo, es decir que el programa modifica constantemente los OFFSETS intentando eliminar el error con la medida real que deseamos: aceleración $x=+16384$, aceleración $y = 0$, aceleración $z = 0$, esto se realiza con un filtro complementario que elimina en lo posible el ruido y las vibraciones inherentes al acelerómetro. Cada 100 lecturas probamos si los valores se acercan a los deseados, dependiendo de esto se

disminuye o aumentan los OFFSETS hasta que los valores converjan. Una vez exista convergencia se finaliza el programa y los acelerómetros están listos para funcionar. Ver programación en ANEXO G.

Supongamos que la IMU esté perfectamente alineada con el suelo. Entonces, como puedes ver en la imagen, el eje Z marcará 9.8, y los otros dos ejes marcarán 0. Ahora supongamos que giramos la IMU 90 grados. Ahora es el eje X el que está perpendicular al suelo, por lo tanto, marcará la aceleración de la gravedad.

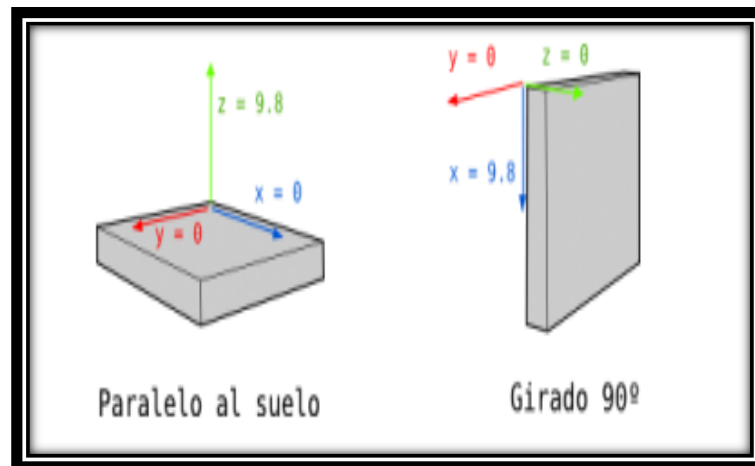


Figura 3-10 Calibración del sensor MPU6050

Fuente: (<http://robologs.net/2014/10/15/tutorial-de-arduino-y-mpu-6050/>, 2016)

Dispositivo final

Una vez realizadas la configuración en cada módulo panstamp, se los acopló en un case con todos sus componentes debidamente conectados en un solo dispositivo listo para ser implementado en el lugar que corresponda.

Cada dispositivo cuenta con un acelerómetro MPU6050, una antena SMA dipolo de $\frac{1}{4}$ de onda que permite la comunicación con el Gateway, también posee su propia fuente de alimentación que consta de 2 pilas AAA DURACEL MN2400BKD de 1.5 V cada una; las mismas que le da independencia al nodo hasta 6 meses desde su implementación.

La placa del módulo Arduino está conectada al shield GPRS SIM900 el cual envía un mensaje de alerta sísmica cada vez que ocurra un evento de magnitud fuerte en la escala de Mercalli, correspondiente a 0.2g de aceleración en los ejes horizontales, de igual manera está conectado a un ordenador de escritorio donde se visualizan las respectivas señales de los nodos. El módulo

Arduino está alimentado por un adaptador de corriente que provee 9V DC a partir de la red eléctrica de la Asociación de escuela de 110V AC.

Para el monitoreo de los nodos se debe poner en funcionamiento los programas realterm para que guarde los valores obtenidos en un archivo txt y Kst para ver los valores en forma gráfica, se pueden observar las diferentes variaciones de aceleraciones en los 3 ejes en tiempo real.



Figura 3-11 Sistema prototipo final

Realizado por: Garcés A, Demera S, 2016

3.2.1.5 Estructura de conexiones

El módulo Panstamp esta soldado sobre la huella del minibat ntc, al cual también están conectados la antena y el sensor mpu 6050 por I2C, es decir respectivamente los conectores SDA(azul), SCL(verde), INT(amarillo), GND(negro) y VCC(rojo), estos dos últimos provenientes del Battery holder.

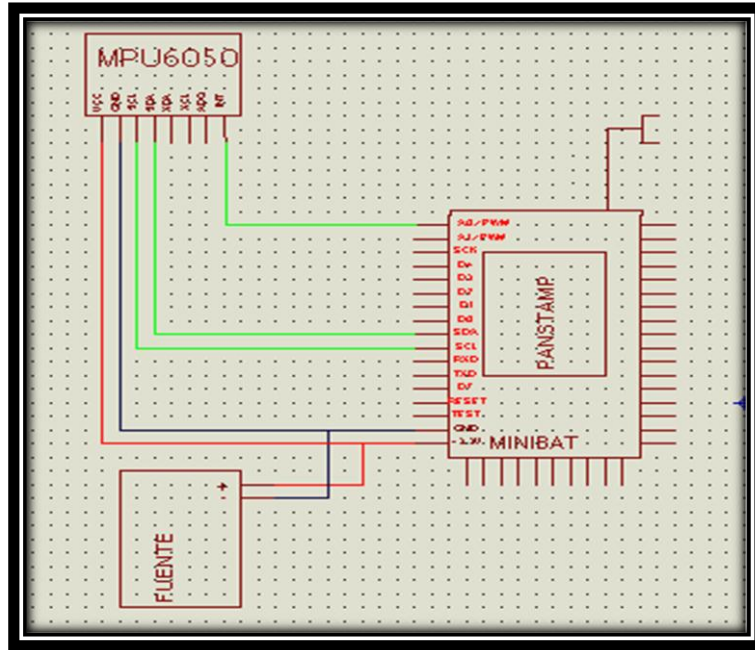


Figura 3-12 Diagrama de conexiones del nodo Panstamp

Realizado por: Garcés A, Demera S, 2016

Al Gateway están conectados el sensor, de forma similar al minibat ntc, la antena y el Arduino mega por serial múltiple. Al módulo Arduino se conectan además del Gateway, el shield GPRS por apilamiento, al pc por USB, y la fuente de 9V.

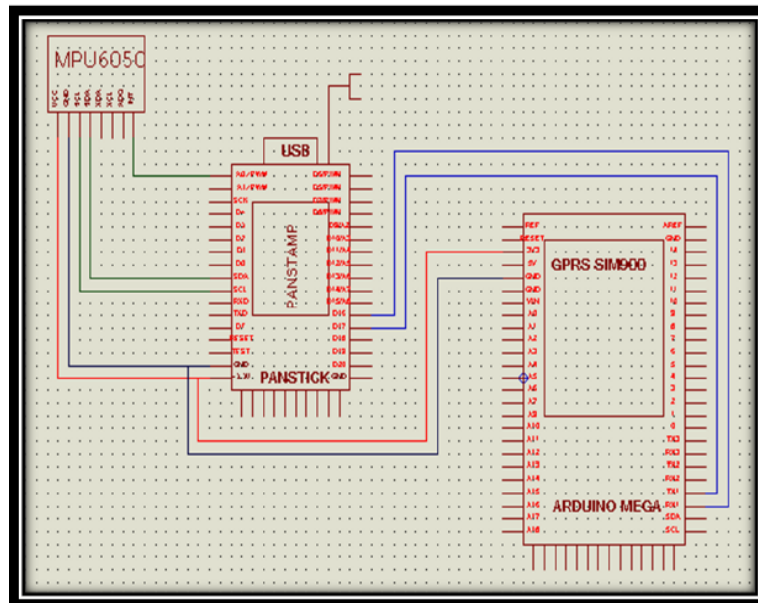


Figura 3-13 Diagrama de conexiones del Gateway

Realizado por: Garcés A, Demera S, 2016



Figura 3-14 Diagrama de conexiones del Gateway

Realizado por: Garcés A, Demera S, 2016

3.2.2 Configuración del software

La supervisión se realiza con el uso de dos softwares que se encargan de registrar los datos y mostrarlos gráficamente.



Figura 3-15 Diagrama de flujo de supervisión

Realizado por: Garcés A, Demera S, 2016

La información obtenida en los nodos es enviada hacia el Gateway, todos los datos son recibidos de manera conjunta y son procesados por el módulo Arduino, razón por la cual fue necesario disponer de un sistema de monitorización de los datos en tiempo real que sea útil y manejable para los usuarios en donde se pueda visualizar la información para analizarla e interpretarla.

La comunicación entre el módulo Arduino y el software es de manera serial, se debe seleccionar el puerto COM del Arduino para que el software empiece a recibir los datos de los nodos y muestre la información de una manera fácil.



Figura 3-16 Supervisión en el PC

Realizado por: Garcés A, Demera S, 2016

El programa Kst se encarga de mostrar la información que recibe del Gateway en tiempo real y visualiza los valores de las aceleraciones en sus tres ejes, lo que permite determinar en qué intervalos las aceleraciones alcanzaron valores picos y así brindar a los usuarios una mejor información de los movimientos sísmicos de la Tierra y que se produzcan principalmente en la ESPOCH.

Para el monitoreo de movimientos sísmicos se definieron valores máximos y mínimos permisibles que serán obtenidos por los nodos panstamp de la WSN, es decir, que mientras no se registre un evento, solo se visualizarán los datos obtenidos del Gateway este programa será puesto en marcha cada vez que el usuario desee realizar la monitorización.

El programa de monitoreo nos ofrece ventajas como la licencia gratuita, interface amigable con el usuario, fácil comprensión, datos graficados en tiempo real, capturas de eventos determinados, etc. Principalmente permite visualizar la información que es emitida por los nodos en tiempo real.

3.2.2.1 Configuración del software Realterm

Es un programa de terminal especialmente diseñado para el control, captura y debug de binarios y otros streams de datos. El uso de este programa permite capturar los datos enviados al puerto serial en el cual está conectado el Arduino mega, para su almacenamiento se crea un archivo de texto que es utilizado para la monitorización en tiempo real.

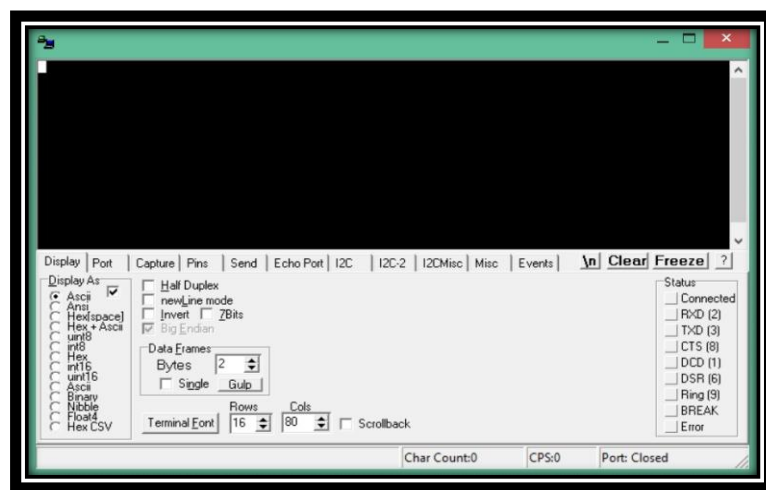


Figura 3-17 Página de inicio de Realterm

Realizado por: Garcés A, Demera S, 2016

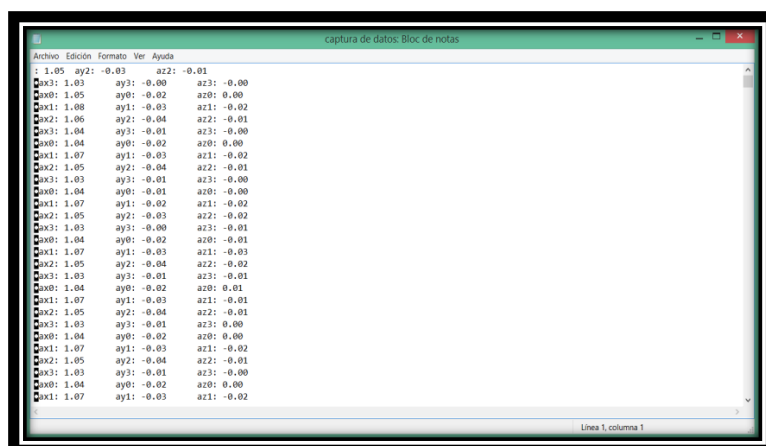


Figura 3-18 Archivo de texto creado

Realizado por: Garcés A, Demera S, 2016

3.2.2.2 Configuración del software Kst

Es una herramienta de visualización y graficación de datos en tiempo real con funcionalidades básicas de análisis de datos, que soporta una gran variedad de archivos con diferentes extensiones. La instalación del programa es sencilla como aceptar las condiciones y dar siguiente hasta finalizar. Luego de instalar el programa Kst, se tiene una pantalla en blanco y se debe empezar con la herramienta “Data Wizard”, que es un asistente que facilita mucho la selección de los elementos y configuraciones básicas necesarias.

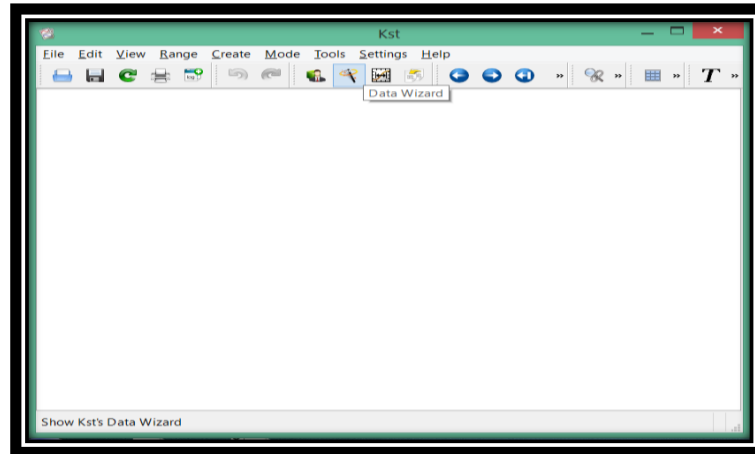


Figura 3-19 Página de inicio de Kst

Realizado por: Garcés A, Demera S, 2016

El programa mostrara las gráficas con escalas automáticas que serán configuradas posteriormente, esto se observa en la figura 3-20

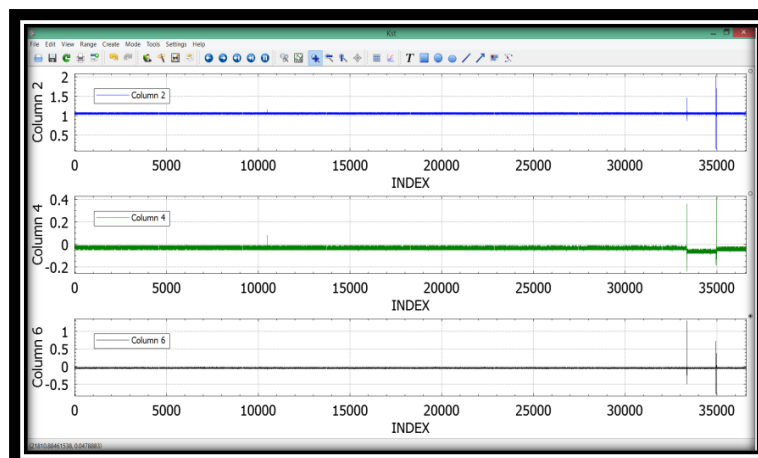


Figura 3-20 Visualización gráfica de los valores

Realizado por: Garcés A, Demera S, 2016

3.3 Pruebas de Funcionamiento del prototipo de la WSN

Las pruebas de funcionamiento se realizaron en las instalaciones de la ESPOCH parque del estudiante, con todos los nodos en funcionamiento, la información llegará hasta la oficina de la Asociación de Escuela EIETR donde se podrá monitorear los nodos.



Figura 3-21 Monitorización en el PC

Realizado por: Garcés A, Demera S, 2016

Para la comunicación entre los nodos y el Gateway, lo óptimo es que existan la menor cantidad de obstáculos posibles, para esto se ubicó la antena del Gateway en la parte superior de la Asociación de Escuela EIETR, y en las instalaciones se encontrará el modulo principal de comunicación para poder ingresar fácilmente a la información que se requiera. La antena del Gateway estará a una altura de 2,83m.



Figura 3-22 Monitorización en el PC

Realizado por: Garcés A, Demera S, 2016

Los nodos se ubicaron a diferentes distancias, a menos de 200 m del Gateway y a una altura superior a 2m del suelo. Las figuras siguientes muestran la ubicación de los nodos y la antena del Gateway para que exista una línea de vista lo más directa posible:

❖ **Punto 1**



Figura 3-23 Ubicación del nodo1 cabaña del parque del estudiante

Realizado por: Garcés A, Demera S, 2016

❖ **Punto 2**



Figura 3-24 Ubicación del nodo 2 estructura de los baños parque del estudiante

Fuente: Garcés A, Demera S, 2016

❖ Punto 3



Figura 3-25 Ubicación del nodo 3 parte posterior de la biblioteca

Fuente: Garcés A, Demera S, 2016

3.3.1 Pruebas de alcance de la red

Se realizaron mediciones en la práctica para determinar el alcance real de la comunicación con los nodos, de lo que se obtuvo lo siguiente:

Tabla 3-1 Alcance de la comunicación para cada nodo.

Distancia desde el Gateway	Nodo #1	Nodo #2	Nodo #3
Con línea de vista	167.36 m	183.43 m	191.58 m
Sin línea de vista	73.96 m	87.17 m	94.58 m

Realizado por: Garcés A, Demera S, 2016

Se monitoreó cada nodo mientras aumentaba la distancia hasta cortarse la comunicación, prueba que se realizó con y sin línea de vista.

Las mediciones fueron realizadas con google maps luego de conocer cada punto donde se cortó la comunicación para cada nodo.

Nodo 1

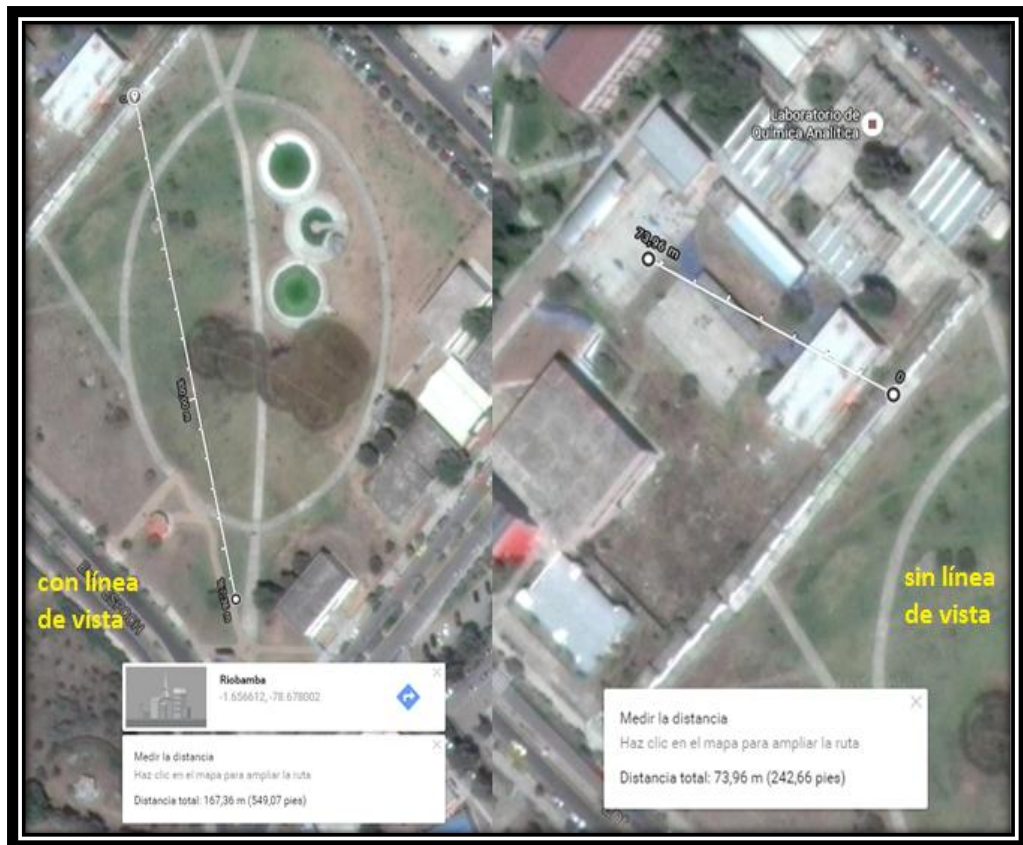


Figura 3-26 Ubicación del nodo 3 parte posterior de la biblioteca

Fuente: Garcés A, Demera S, 2016

Nodo 2



Figura 3-27 Ubicación del nodo 3 parte posterior de la biblioteca

Fuente: Garcés A, Demera S, 2016

Nodo 3



Figura 3-28 Ubicación del nodo 3 parte posterior de la biblioteca

Fuente: Garcés A, Demera S, 2016

3.3.2 Datos obtenidos con el prototipo de WSN para la alerta temprana de sismos

Se monitorearon todos los nodos ocho horas diariamente de lunes a viernes durante dos semanas sucesivas desde el lunes 2 de mayo del 216 hasta el viernes 20 de mayo del 2016 obteniendo los siguientes valores máximos de aceleración:

Tabla 3-2 Datos obtenidos con el prototipo funcional

Fecha	Nodo 1			Nodo 2			Nodo 3			Gateway		
	Ac. Eje X	Ac. Eje Y	Ac. Eje Z	Ac. Eje X	Ac. Eje Y	Ac. Eje Z	Ac. Eje X	Ac. Eje Y	Ac. Eje Z	Ac. Eje X	Ac. Eje Y	Ac. Eje Z
Lunes 02/05/2016	1,02	0,04	0,03	0,97	0,03	0,04	1,01	0,04	0,02	1,02	0,03	0,02
Martes 03/05/2016	1,02	0,04	0,03	0,99	0,04	0,03	1,01	0,03	0,03	1,02	0,03	0,03
Miércoles 04/05/2016	1,01	0,05	0,02	0,98	0,04	0,03	1	0,03	0,03	1,03	0,04	0,03
Jueves 05/05/2016	1,02	0,03	0,03	0,97	0,04	0,04	1,01	0,04	0,02	1,02	0,03	0,03
Viernes 06/05/2016	1,01	0,04	0,02	1,01	0,03	0,03	1,02	0,03	0,02	1,03	0,04	0,02
Lunes 09/05/2016	1,02	0,04	0,02	0,98	0,03	0,03	1,01	0,03	0,02	1,02	0,03	0,03
Martes 10/05/2016	1,02	0,04	0,02	0,98	0,03	0,03	1,02	0,03	0,02	1,01	0,03	0,03
Miércoles 11/05/2016	1,02	0,04	0,03	0,99	0,04	0,04	1,01	0,04	0,03	1,01	0,03	0,02
Jueves 12/05/2016	1,02	0,03	0,02	0,97	0,03	0,04	1,01	0,03	0,03	1,02	0,04	0,02
Viernes 13/05/2016	1,03	0,04	0,03	0,97	0,04	0,03	1,01	0,03	0,02	1,02	0,03	0,04
Lunes 16/05/2016	1,02	0,05	0,03	0,98	0,03	0,03	1,01	0,03	0,03	1,02	0,03	0,03
Martes 17/05/2016	1,02	0,04	0,03	0,98	0,03	0,04	1,01	0,04	0,04	1,01	0,03	0,03
Miércoles 18/05/2016	1,03	0,05	0,04	1,01	0,04	0,04	1,02	0,04	0,03	1,03	0,04	0,04
Jueves 19/05/2016	1,03	0,03	0,03	1,01	0,04	0,03	1,02	0,04	0,03	1,02	0,04	0,03
Viernes 20/05/2016	1,01	0,03	0,02	0,97	0,03	0,04	1,01	0,03	0,02	1,02	0,02	0,03

Realizado por: Garcés A, Demera S, 2016

3.3.3 *Análisis estadístico*

Al analizar el total de datos obtenidos durante el muestreo en las fechas mencionadas se obtuvo una muestra de 291202 datos almacenados para cada eje de aceleración de cada nodo, para un total de 3494424 datos, de ellos analizamos lo siguiente:

❖ **NODO 1**

Rango: $acx= 0.97(g)$ a $1.03(g)$

$acy= -0.03(g)$ a $0.05(g)$

$acz= -0.05(g)$ a $0.04(g)$

Media aritmética

$acx= 1.002357(g)$

$acy= 0.002377(g)$

$acz= 0.002653(g)$

Varianza

$acx= 2.12 \cdot 10^{-5}$

$acy= 1.98 \cdot 10^{-5}$

$acz= 4.83 \cdot 10^{-5}$

Desviación estándar

$acx= 0.004606$

$acy= 0.004454$

$acz= 0.006952$

De lo anterior se interpreta que, por los valores muy pequeños obtenidos en la varianza, la dispersión es extremadamente baja y los datos del sensor tienden al valor de la media, por lo tanto, el comportamiento del sensor es muy estable.

Correlación

$acy \leftrightarrow acz= 0.02548$

Se infiere que la relación entre las aceleraciones es muy baja, lo que implica un alto grado de independencia entre las aceleraciones horizontales.

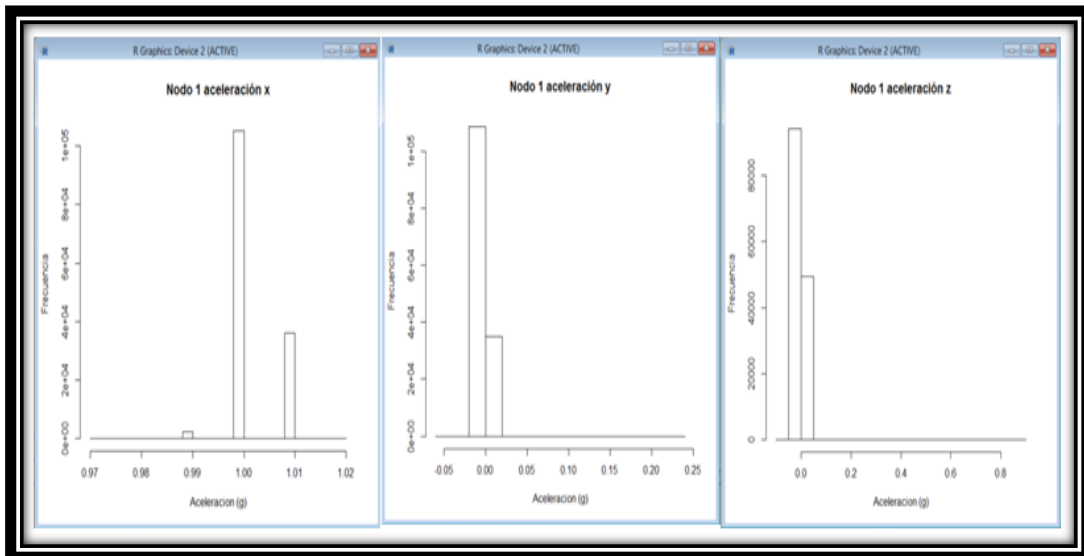


Figura 3-29 Histograma del nodo 1

Realizado por: Garcés A, Demera S, 2016

❖ **Nodo 2**

Rango: $acx = 0.96(g)$ a $1.01(g)$

$acy = -0.03(g)$ a $0.04(g)$

$acz = -0.04(g)$ a $0.04(g)$

Media aritmética

$acx = 0.99247(g)$

$acy = 0.00823(g)$

$acz = 0.008952(g)$

Varianza

$acx = 2.35 \cdot 10^{-5}$

$acy = 3.09 \cdot 10^{-5}$

$acz = 2.97 \cdot 10^{-5}$

Desviación estándar

$acx = 0.004848$

$acy = 0.00556$

$acz = 0.005457$

De lo anterior se interpreta que, por los valores muy pequeños obtenidos en la varianza, la dispersión es extremadamente baja y los datos del sensor tienden al valor de la media, por lo tanto, el comportamiento del sensor es muy estable.

Correlación

$$acy \leftrightarrow acz = 0.02802$$

Se infiere que la relación entre las aceleraciones es muy baja, lo que implica un alto grado de independencia entre las aceleraciones horizontales.

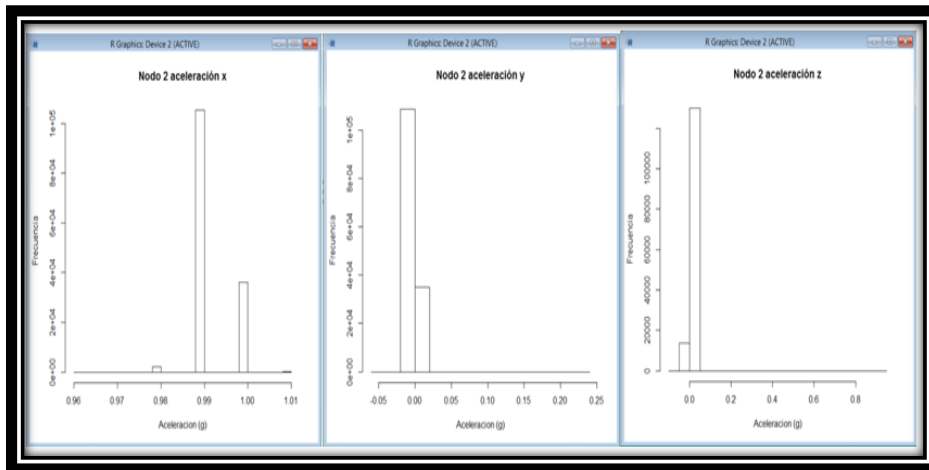


Figura 3-30 Histograma del nodo 2

Realizado por: Garcés A, Demera S, 2016

❖ Nodo 3

Rango: $acx = 0.97(g)$ a $1.02(g)$

$acy = -0.04(g)$ a $0.04(g)$

$acz = -0.03(g)$ a $0.04(g)$

Media aritmética

$acx = 1.02408(g)$

$acy = 0.003913(g)$

$acz = 0.000405(g)$

Varianza

$acx = 2.13 \cdot 10^{-5}$

$acy = 1.99 \cdot 10^{-5}$

$acz = 3.76 \cdot 10^{-5}$

Desviación estándar

$acx = 0.004624$

$acy = 0.004468$

$acz = 0.006131$

De lo anterior se interpreta que, por los valores muy pequeños obtenidos en la varianza, la dispersión es extremadamente baja y los datos del sensor tienden al valor de la media, por lo tanto, el comportamiento del sensor es muy estable.

Correlación

$$acy \leftrightarrow acz = 0.01957$$

Se infiere que la relación entre las aceleraciones es muy baja, lo que implica un alto grado de independencia éntrelas aceleraciones horizontales.

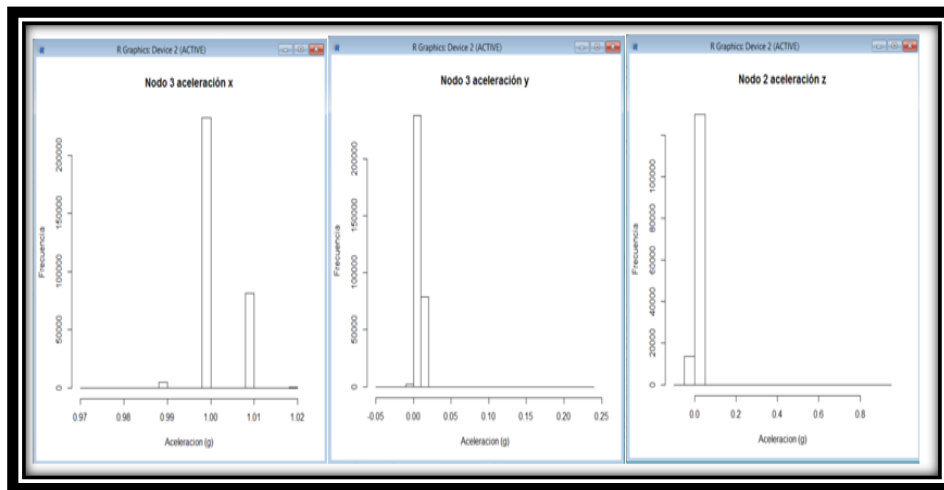


Figura 3-31 Histograma del nodo 3

Realizado por: Garcés A, Demera S, 2016

❖ **Nodo 4 Gateway**

Rango: $acx = 0.98(g)$ a $1.03(g)$

$acy = -0.02(g)$ a $0.04(g)$

$acz = -0.03(g)$ a $0.04(g)$

Media aritmética

$acx = 1.01280(g)$

$acy = 0.004039(g)$

$acz = 0.005885(g)$

Varianza

$acx = 2.24 \cdot 10^{-5}$

$acy = 2.016 \cdot 10^{-5}$

$$acz=4.104*10^{-5}$$

Desviación estándar

$$acx=0.004738$$

$$acy= 0.004490$$

$$acz=0.006406$$

De lo anterior se interpreta que, por los valores muy pequeños obtenidos en la varianza, la dispersión es extremadamente baja y los datos del sensor tienden al valor de la media, por lo tanto, el comportamiento del sensor es muy estable.

Correlación

$$acy \leftrightarrow acz = -0.01345$$

Se infiere que la relación entre las aceleraciones es muy baja, lo que implica un alto grado de independencia éntrelas aceleraciones horizontales.

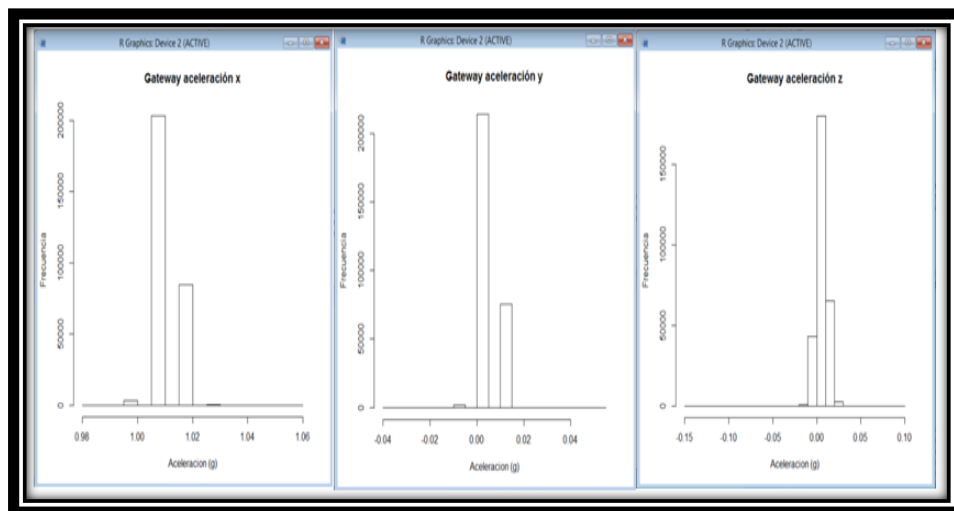


Figura 3-32 Histograma del nodo 4

Realizado por: Garcés A, Demera S, 2016

3.3.4 Cálculo de Pérdidas (db) en el espacio libre

Las pruebas con los nodos se realizarán a campo abierto en la ESPOCH (parque del estudiante), una zona urbana, desde los transmisores hasta el receptor existe línea de vista, lo que permite facilitar los cálculos.

En el proyecto se usará el modelo de propagación de Walfish Ikegami directamente con la fórmula para lugares en donde existe línea de vista y es zona urbana entre los nodos:

➤ NODO 1

$$L_p(\mathbf{db}) = 42,6 + 26\log d + 20\log f$$

$$L_p(\mathbf{db}) = 42,6 + 26\log (0,135\mathbf{km}) + 20\log(915\mathbf{MHz})$$

$$L_p(\mathbf{db}) = 42,6 - 22,61 + 59,22$$

$$L_p(\mathbf{db}) = 79,22 \mathbf{db}$$

➤ NODO2

$$L_p(\mathbf{db}) = 42,6 + 26\log d + 20\log f$$

$$L_p(\mathbf{db}) = 42,6 + 26\log (0,13\mathbf{km}) + 20\log(915\mathbf{MHz})$$

$$L_p(\mathbf{db}) = 42,6 - 23,04 + 59,22$$

$$L_p(\mathbf{db}) = 78,79 \mathbf{db}$$

➤ NODO3

$$L_p(\mathbf{db}) = 42,6 + 26\log d + 20\log f$$

$$L_p(\mathbf{db}) = 42,6 + 26\log (0,087\mathbf{km}) + 20\log(915\mathbf{MHz})$$

$$L_p(\mathbf{db}) = 42,6 - 27,57 + 59,22$$

$$L_p(\mathbf{db}) = 74,25 \mathbf{db}$$

3.3.5 *Cálculo de la potencia recibida en el enlace inalámbrico*

Se realizará un balance de potencias para determinar la potencia recibida en el Gateway usando valores como la sensibilidad que tienen los nodos

$$\mathbf{Potencia}_{TX} = 2\mathbf{Mw} = 3\mathbf{db}$$

$$\mathbf{Ganancia}_{TX} = 3 \mathbf{dbi}$$

$$\mathbf{Ganancia}_{RX} = 3 \mathbf{dbi}$$

$$\mathbf{Sensibilidad} = -111 \mathbf{dbi}$$

Balance de Potencias:

NODO 1

$$P_{RX1} = P_{TX} + G_{TX} - L_{TOTAL} + G_{RX} - S$$

$$P_{RX1} = 12dbm + 3dbi + 2.1db - 79.22db + 3dbi - (-111dbm)$$

$$P_{RX1} = 19.78 \text{ dBm}$$

NODO 2

$$P_{RX2} = P_{TX} + G_{TX} - L_{TOTAL} + G_{RX} - S$$

$$P_{RX2} = 12dbm + 3dbi + 2.1db - 78.79db + 3dbi - (-111dbm)$$

$$P_{RX2} = 20.21 \text{ dBm}$$

NODO 3

$$P_{RX3} = P_{TX} + G_{TX} - L_{TOTAL} + G_{RX} - S$$

$$P_{RX3} = 12dbm + 3dbi + 2.1db - 79.22db + 3dbi - (-111dbm)$$

$$P_{RX3} = 24.75 \text{ dBm}$$

3.3.6 Cálculo de la distancia límite

Para conocer a partir de que distancias debe producirse un sismo para darnos tiempo para actuar, ya que el tiempo aproximado que tarda el sistema hasta que el SMS de alerta llega a los números establecidos es 5, se debe conocer la velocidad típica con que se propagan las ondas sísmicas que son: 6km/s para la onda principal, y la onda secundaria 3.6km/s (60% de la velocidad de la onda principal), y a partir de esto descubrir esta distancia.

Los valores de velocidades tomados son los más altos que puede tomar la onda, ya que su velocidad varia en dependencia del material del que esté conformado el suelo, siendo mayores las velocidades, mientras más alta sea la densidad del suelo.

$$d_1 = V_1 * t_1 ; d_2 = V_2 * t_2$$

$$t_2 - t_1 = 5s$$

$$d_1 = d_2$$

$$V_1 * t_1 = V_2 * t_2$$

$$V_1 * (t_2 - 5s) = V_2 * t_2$$

$$V_1 * t_2 - V_1 * (5s) = V_2 * t_2$$

$$V_1 * t_2 - V_2 * t_2 = V_1 * (5s)$$

$$\begin{aligned}
t_2(V_1 - V_2) &= V_1 * (5s) \\
t_2 &= \frac{V_1 * (5s)}{(V_1 - V_2)} \\
t_2 &= \frac{6km/s * (5s)}{(6km/s - 3.6km/s)} \\
t_2 &= 12.5s \\
t_1 &= t_2 - 5s \\
t_1 &= 7.5s \\
d_1 &= V_1 * t_1 ; d_2 = V_2 * t_2 \\
d_1 &= 45km ; d_2 = 45km
\end{aligned}$$

La distancia mínima a partir de la cual el mensaje de texto llegará antes que la onda secundaria es de 45Km.

3.3.7 *Análisis económico*

En este apartado se realizará un estudio económico para determinar el beneficio financiero que se tendrá con la implementación de la WSN con panstamp. En la investigación realizada se ha encontrado que el IGEPN es el único organismo encargado del monitoreo y aviso de los eventos que ocurren en la naturaleza, específicamente los sismos y la actividad volcánica en nuestro país.

Los equipos que realizan esta labor son de un elevado costo lo que requiere de una gran inversión, en nuestro estudio observaremos los valores de los equipos profesionales que utiliza el IGEPN y los valores de nuestros equipos que servirán para dar una alerta de sismo en la ESPOCH.

Tabla 3-3 Inversión en equipos para movimiento sísmico

MARCA	MODELO	DESCRIPCION	PRECIO
Guralp	CMG-5TDE	El CMG-5TD es un acelerómetro triaxial completamente digital.	\$ 12235.71
Reftek	130-01	Acelerógrafo que combina el Digitalizador Sísmico de Banda-ancha y un Acelerómetro de bajo nivel de ruido.	\$ 9299.00
Guralp	CMG-CD24	El CD24 es un compacto, eficiente y multi-propósito digitalizador. Posee tres entradas diferenciales de 24 bits y 8 entradas de baja velocidad.	\$ 2633.00
Trimble	NetRS	Receptor GPS versátil con control de comunicación avanzado	\$ 2195.00
Trimble	Zephyr Geodetic P/N 41249-00	Antena satelital	\$ 349.00
	bateria 12 voltios 45 Ah	Batería usada para equipos de bajo consumo en el campo.	\$ 24.00
Zytech Solar	ZT 100S	Fotocelda monocristalina para alimentar una batería de 12V y 45Ah	\$587.50
TOTAL			\$27323.21

Realizado por: Garcés A, Demera S, 2016

Inversión del Prototipo

Tabla 3-4 Valor de equipos utilizados para el prototipo

DISPOSITIVO	CANT.	MARCA	MODELO	VALOR	TOTAL
PANSTAMP NRG 2	4	PANSTAMP	NGR V2	\$ 19.71	\$ 78.84
PANSTICK	1	PANSTAMP	V4.0	\$ 19.35	\$ 19.35
MINIBAT	3	PANSTAMP	NTC 1.0	\$ 16.69	\$ 50.07
ARDUINO MEGA	1	ARDUINO	2560	\$ 59.99	\$ 59.99
SENSORES	4	INVENSENSE	MPU6050	\$18.75	\$ 75.00
BATTERY HOLDER	3	-	-	\$ 2.12	\$6.36
CAJA HERMETICA	3	-	-	\$ 7.16	\$ 21.48
ANTENA SMA ARTICULADA	3	-	-	\$ 6.56	\$ 19.68
ANTENA SMA BASE MAGNETICA	1	-	-	\$ 9.52	\$ 9.52
CAJA PARA ARDUINO	1	-	-	\$ 11.90	\$ 11.90
SHIELD GPRS V2.0b	1	SEED STUDIO	V2.0b	\$ 67.50	\$ 67.50
ADAPTADOR DE VOLTAJE 1.5- 12VDC, 1A	1	COBY	MV7100	\$ 12.50	\$ 12.50
ORDENADOR DE ESCRITORIO	1	-	-	\$350.00	\$ 350.00
TASAS E IMPUESTOS	-	-	-	\$ 60.12	\$ 60.12
COSTO DE ENVIO	-	-	-	\$ 52.32	\$ 52.32
TOTAL					\$ 894.63

Realizado por: Garcés A, Demera S, 2016

Interpretación:

Con la red de sensores implementada se puede realizar un monitoreo y análisis el día y hora que se requiera, es importante señalar que se utilizan programas de interface gráfica para una mayor facilidad al momento de monitorizar los eventos promoviendo el desarrollo de nuevas investigaciones a futuro. En la tabla 3-4 se puede observar que la implementación del prototipo brinda un beneficio financiero de \$26428.58 con respecto a un sistema profesional, para el beneficio que brinda representa ser un sistema conveniente para el lugar donde se implementó.

Tabla 3-5 Análisis comparativo de inversiones

Supervisión sísmica	Total (\$)
Inversión en un sistema profesional	27323.21
Inversión del prototipo	894.63
Ahorro	26428.58

Realizado por: Garcés A, Demera S, 2016

3.3.8 Evaluación de la fuente de alimentación

Para el análisis del consumo energético de las fuentes de alimentación de los nodos se ha considerado el uso de pilas AAA Y AA, debido a su tamaño pequeño lo que permitirá una fácil instalación y portabilidad de los mismos.

Se necesita conocer algunos parámetros para realizar este análisis; el consumo energético de los sensores y de los panstamp, estos últimos tienen un consumo variable debido a que están alternando entre un modo de bajo consumo y modo de transmisión, se debe conocer el porcentaje de tiempo correspondiente a cada modo.

Para este análisis tomamos en consideración el INFORME SÍSMICO PARA EL ECUADOR AÑO 2012 realizado por el IGEPN según el cual desde el 2000 al 2012 el promedio de sismos en el Ecuador superiores a 4 grados es de 277.25 sismos al año, por lo tanto, si tomamos como promedio una duración de 1 minuto se puede afirmar que el tiempo total de actividad de los nodos va hacer el promedio de los sismos durante el año multiplicado por su duración y el tiempo en que los nodos estarán en modo de bajo consumo, será el tiempo restante.

Se deduce:

$$\text{Consumo Total} = \text{Consumo}_{act} + \text{Consumo}_{inact} + \text{Consumo}_{sensor}$$

$$\text{Consumo}_{act} = 36mA * \% \text{Tiempo}_{act}$$

$$\text{Consumo}_{inact} = 2\mu A * \% \text{Tiempo}_{inact}$$

$$\% \text{Tiempo Total} = \% \text{Tiempo}_{act} + \% \text{Tiempo}_{inact}$$

$$\text{Tiempo}_{act}(\text{anual}) = \text{Numero de eventos en el año} * \text{duracion de los eventos}$$

$$\text{Tiempo}_{act}(\text{anual}) = (277.25 * 1\text{min})$$

$$\text{Tiempo}_{act}(\text{anual}) = 277.25 \text{ min}$$

$$\text{Tiempo}_{act}(\text{anual}) = 4.62 \text{ h}$$

$$\frac{\% \text{Tiempo Total}}{\% \text{Tiempo}_{act}} = \frac{1 \text{ año}}{\text{Tiempo}_{act}}$$

$$\% \text{Tiempo}_{act} = \frac{\% \text{Tiempo Total} * \text{Tiempo}_{act}}{1 \text{ año}}$$

$$\% \text{Tiempo}_{act} = \frac{100\% * 4.62h}{8760h}$$

$$\% \text{Tiempo}_{act} = 0.053\%$$

$$\% \text{Tiempo Total} = \% \text{Tiempo}_{act} + \% \text{Tiempo}_{inact}$$

$$\% \text{Tiempo}_{inact} = \% \text{Tiempo Total} - \% \text{Tiempo}_{act}$$

$$\% \text{Tiempo}_{inact} = 100\% - 0.053\%$$

$$\% \text{Tiempo}_{inact} = 99.947\%$$

$$\text{Consumo Total} = \text{Consumo}_{act} + \text{Consumo}_{inact} + \text{Consumo}_{sensor}$$

$$\text{Consumo Total} = (36mA * \% \text{Tiempo}_{act}) + (0.002mA * \% \text{Tiempo}_{inact}) + (0.5mA)$$

$$\text{Consumo Total} = (36mA * 0.00053) + (0.002mA * 0.99947) + (0.5mA)$$

$$\text{Consumo Total} = 0.521mA$$

$$\text{Duracion} = \frac{\text{Energia de las baterias}}{\text{Consumo total}}$$

$$\text{Duracion} = \frac{1150mAh * 2}{0.521mA}$$

$$\text{Duracion} = 4414.59h$$

$$\text{Duracion} = 183.94 \text{ dias}$$

$$\text{Duracion} = 6.13 \text{ meses}$$

Cálculo realizado con pilas alcalinas AAA DURACELL MN2400BKD.

$$Duracion = \frac{Energia\ de\ las\ baterias}{Consumo\ total}$$

$$Duracion = \frac{800mAh * 2}{0.521mA}$$

$$Duracion = 3071.02h$$

$$Duracion = 127.96\ dias$$

$$Duracion = 4.27\ meses$$

Cálculo realizado con pilas recargables AAA DURACELL DX2400B2.

Para la implementación se decidió utilizar pilas DURACELL AAA MN2400BKD porque tienen un tiempo de vida mayor a las pilas recargables y considerando que 6 meses es un tiempo aceptable para los equipos inalámbricos.

El Gateway y el módulo Arduino serán conectados a una toma AC disponible en el lugar de monitorización de los eventos.

CONCLUSIONES

- ❖ Se desarrolló una red de sensores inalámbricos tipo estrella que consiste de tres nodos sensores, un Gateway y una estación de supervisión, capaz de medir las aceleraciones en tiempo real y realizar la alerta de eventos sísmicos superiores a 0.2g, equivalente a magnitud 5 en la escala de Mercalli.
- ❖ Se desplegaron nodos capaces de poseer autonomía durante 6.13 meses, con un consumo de 0.521 mA, al ser alimentados por pilas AAA con capacidad superior a 1100mAh.
- ❖ Se realizaron pruebas del alcance real de la comunicación entre los nodos y el Gateway. El valor más alto obtenido fue de 191.58m con línea de vista, que es inferior al indicado por el fabricante, pero suficientemente amplio para el funcionamiento adecuado del sistema.
- ❖ Se tomaron registros de las aceleraciones sensadas por los nodos y se obtuvo un total de 3499424 muestras. Se concluye que el sistema se comporta de manera estable, con una varianza de $1.98 \cdot 10^{-5}$, y que sus ejes horizontales son independientes al tener una correlación entre ellos de -0.01345.
- ❖ Se pudo concluir que los equipos Panstamp se acoplan mejor al presente proyecto debido que al ser comparados con Arduino y Raspberry Pi, demostró ser la placa de desarrollo más adecuada para la implementación del sistema.
- ❖ La distancia mínima a partir de la cual se debe producir un sismo, para que el mensaje de texto llegue antes que la onda secundaria es de 45Km.

RECOMENDACIONES

- ❖ Es necesario realizar pruebas del sistema en lugares donde exista este tipo de eventos con una mayor frecuencia para poder determinar si cumple con los requerimientos para su comercialización.
- ❖ Incorporar un módulo de almacenamiento al sistema para poder obtener datos históricos del mismo en el momento que se requiera.
- ❖ Desarrollar un software que permita incorporar nuevos dispositivos a la red y de esta manera ampliar el alcance del sistema para obtener más beneficios.
- ❖ Para obtener los valores de aceleraciones con bajo índice de error es necesario realizar la calibración de cada uno de los sensores sobre la superficie que será colocado en caso de ser reubicados.
- ❖ Utilizar como fuente de alimentación pilas AAA superiores a 1100mAh y realizar su reposición preventiva cada cinco meses para garantizar el funcionamiento de los nodos.
- ❖ Investigar una forma de alimentación adecuada en función del lugar donde sea implementado.
- ❖ Tener un conocimiento claro del funcionamiento del software Realterm y Kst para realizar la supervisión del sistema.
- ❖ Para que el sistema pueda enviar los mensajes de alerta es imprescindible que el chip tenga siempre saldo disponible.

GLOSARIO

APL	Application (Aplicación)
BLUETOOTH	Especificación industrial de redes inalámbricas para área personal.
CPU	Central Processing Unit (Unidad Central de Proceso)
DB	Unidad que se utiliza para medir la intensidad del sonido y otras magnitudes físicas.
DSSS	Direct sequence spread spectrum (Espectro ensanchado de secuencia directa)
FFD	Full Function Device (Dispositivo completo de funciones)
FHSS	Frequency hopping spread spectrum (Saltos de frecuencia de espectro ensanchado)
GATEWAY	puerta de enlace, equipo para conectar redes
GPRS	General packet radio services
HID	Human Interface Device (Dispositivos para la Interfaz Humana)
IDE	Integrated Development Environment (Entorno de desarrollo integrado)
IEEE	Institute of Electrical and Electronics Engineers (Instituto de Ingeniería Eléctrica y Electrónica)
IGEPN	Instituto Geofísico de la Escuela Politécnica Nacional
INEC	Instituto Nacional de Estadísticas y Censos
ISO	International Standard Organization (Organización Internacional de Normalización)
KST	es un programa de trazado y visualización de datos.
LR-WPAN	Low-rate wireless personal area network (Red de área personal inalámbrica de baja velocidad)
M2M	machine to machine (máquina a máquina)
MAC	Media Access Control (Control de acceso al medio)
NWK	Network (Red)
OFFSET	Es el valor de salida del sensor cuando la magnitud medida es cero.
OSI	Open System Interconnection (Interconexión de Sistema Abierto)
PHY	Physical Layer (Capa física)
REALTERM	es un programa de terminal especialmente diseñado para capturar, controlar y depurar secuencias de datos binarios y otros difíciles.
RFD	Reduced Function Device
RTP	Real-time Transport Protocol (Protocolo de transporte en tiempo real)
Rx	Receiving Data (Recepción de Datos)
SMS	Servicio de mensajes cortos o servicio de mensajes simples.

SRM	Supplier Relationship Management (Gestión de las relaciones con los proveedores)
SWAP	Especificación de transmisión de voz y datos en una red doméstica.
TCL	Tool Command Language
Tx	Data transmission (Transmisión de datos)
UART	Universal Asynchronous Receiver- Transmitter (Transmisor – receptor Universal Asíncrono)
UDP	User Datagram Protocol (Protocolo de Datagrama de Usuario)
USB	Universal Serial Bus
VCC	Voltaje de Corriente Continua
WIFI	Wireless Fidelity (Fidelidad Inalámbrica)
WLAN	Wireless Local Area Network (Red Inalámbrica de área local)
WPAN	Wireless Personal Area Network (Red Inalámbrica de Área Personal)
WSN	Wireless sensor network (Red de Sensor Inalámbrica)
ZIGBEE	Protocolos de comunicación inalámbrica para su utilización con radiodifusión digital

BIBLIOGRAFÍA

- ❖ **AGUDELO, J.** ¿Qué relación existe entre la aceleración de cálculo del sismo y la escala sismológica de Richter y la de Mercalli?,
<http://estructurando.net/2015/09/14/que-relacion-existen-entre-la-aceleracion-de-calculo-del-sismo-y-la-escala-sismologica-de-richter-y-la-de-mercalli/>
2015-09-14

- ❖ **AKI, K.** “Scaling law of seismic spectrum.”
Journal of Geophysical Research.
1967.p.217-231

- ❖ **AMPERE.,** Estación Sísmica, Mexico
<http://www.ampere.mx/estacioacuten-siacutesmica.html>
2015-11-05

- ❖ **ARDUINO.,** Página en español Estados Unidos: 2013.
<https://www.arduino.cc/es/pm/wiki.php?n=>
2015-07-12

- ❖ **BERNAL, G.** “Sistema de estimación inmediata del daño pos-terremoto para la ciudad de Bogotá.” (TESIS). Universidad de los Andes. Facultad de Ingeniería, Bogotá. 2009. P.26-28.

- ❖ **COLOMBELLI, S; ZOLLO, A; FESTA, G.** “Early magnitude and potential damage zone estimates for the great earthquake.” Geophysical Research Letters.
2012. p.22

- ❖ **DIYHACKING,** arduino mpu 6050 imu sensor tutorial, Ontario Canadá. 2015
<http://diyhacking.com/arduino-mpu-6050-imu-sensor-tutorial/>

2016-01-17

- ❖ **ESPINDOLA, J; JIMENEZ, Z.** “Terremotos y ondas sísmicas”. Instituto de Geofísica de la UNAM. 2002. p.94-97.
- ❖ **GARCIA, M.,** “*Guía para la Implementación de Sensores Inalámbricos*”. (TESIS). Universidad Pontificia Bolivariana, Facultad de Ingeniería Electrónica, Bucaramanga. 2010. p. 19.
- ❖ **HAVSKOV, J.; ALGUACIL, G.** “Instrumentation in earthquake seismology” Spiringer. Holanda. 2004. p.326-331.
- ❖ **INSTITUTO GEOFÍSICO EPN.** Red Nacional de acelerógrafos , Ecuador.
<http://www.igepn.edu.ec/red-nacional-de-acelerografos>
2016-02-09
- ❖ **LAWRENCE , J.; COCHRAN , E.; CHUNG , A.** “Rapid Earthquake Characterization Using MEMS Accelerometers”. Nueva Seismic Society. P.184-192.
- ❖ **NATIONAL INSTRUMENTS.** *Redes Inalámbricas de Sensores*, Estados Unidos.
<http://www.ni.com/wsn/esa/>
2015-06-01
- ❖ **PANSTAMP.** *Placas de desarrollo en miniatura para aplicaciones wsn, España.*
<http://www.panstamp.com/es/>
2016-02-01.
- ❖ **PAREDES, I.** *Análisis comparativo de las placas Arduino (oficiales y compatibles).* Italia: 2014.
<http://comohacer.eu/analisis-comparativo-placas-arduino-oficiales-compatibles/>
2015-09-15

- ❖ **SECRETARÍA DE GESTIÓN DE RIESGOS.** *Informes de Situación: Terremoto 7.8 Pedernales (16/04/2016)*, 2016.
<http://www.gestionderiesgos.gob.ec/informes-de-situacion-actual-terremoto-magnitud-7-8/>.
2016-05-06

- ❖ **SERNA, J.** *Redes de Sensores Inalámbricas*, España: Jesus Serna 2007
<http://www.uv.es/~montanan/ampliacion/trabajos/Redes%20de%20Sensores.pdf>.
2015-04-26

- ❖ **TEXAS INSTRUMENTS.** *Low Power RF & Wireless Connectivity*, Estados Unidos.
http://e2e.ti.com/support/wireless_connectivity/
2016-03-01

- ❖ **USGS.GOV.** *U.S. Geological Survey*, Estados Unidos. 2016.
<http://earthquake.usgs.gov/earthquakes/map/>
2015-09-20

Anexo B

Datasheet Gateway panstick

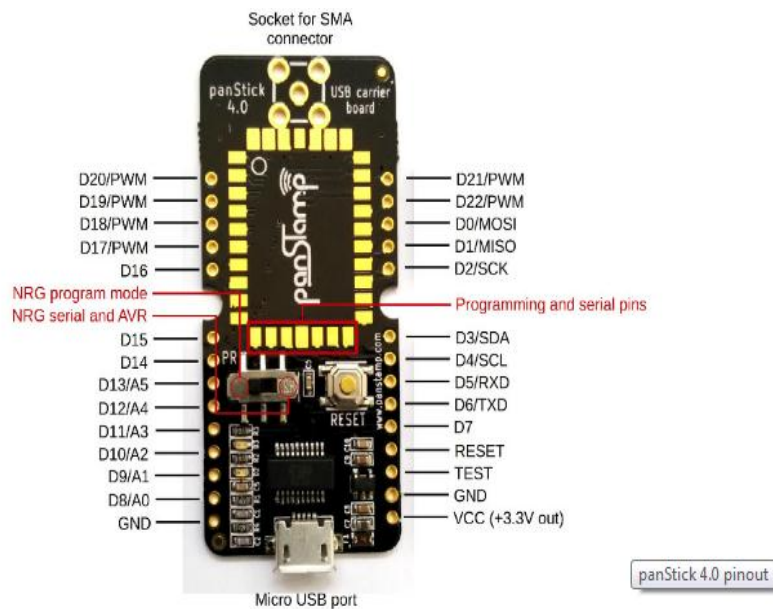
panStick is a special USB-UART gateway designed for panStamp 2. It provides the necessary footprint to include a panStamp (NRG or AVR) and can also be used without a wireless module, by stacking or plugging under or onto other carrier boards. This gateway can be used as either a serial programmer and as a serial modem for wireless networks.

This version can program either panStamps AVR 2 and NRG 2. By setting the on-board selector to the suitable position, you will be able to put the panStick into one of these modes:

- Left position (PR label): program and run applications on panStamp NRG 2. This position won't work when using the module serially from a serial terminal or as a modem, due to the special use of the RESET line (without a series capacitor).
- Right position: program and run applications on AVR modules. This position can also be used to run NRG applications including serial applications (USB).

Pinout and ports

panStick 4 has the same pinout as the rest of mini-carrier boards. This means that panStick can be stacked under (or onto) other carrier boards, as far as only one of the carrier boards has a panStamp module assembled.



Hardware specifications

- Size: 1.0 x 2.2 in (25.4 x 55.8 mm)
- Pin spacing: 0.1 in (2.54 mm)
- USB/UART converter: FTDI FT231XS
- Supported OS: Linux, MacOS and Windows

panStick as a serial modem

Apart from using panStick to program panStamps, it can be used as a serial interface to communicate with wireless networks from a serial terminal SWAPdmt, lagarto-swap, them or your custom software. In order to do this, you have to program your on-board panStamp with our "modem" sample application and place the manual switch on the right position (opposite to "PR"). You can read about our serial protocol and AT commands [here](#).

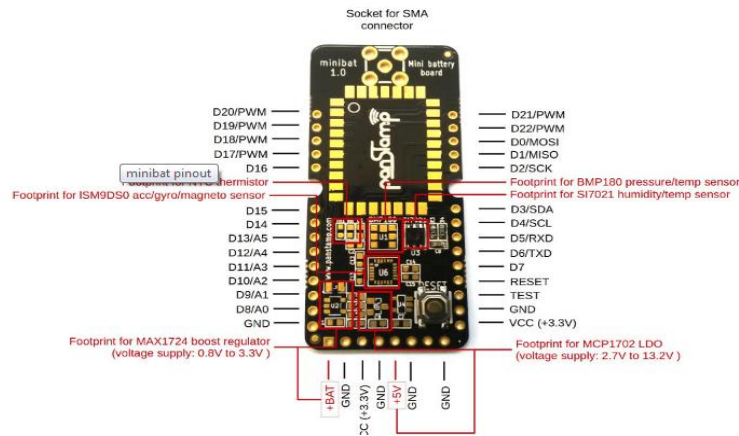
Anexo C

Datasheet Minibat

Mini-battery-board with on-board NTC. This minibat carrier board is the perfect hardware to make your panStamp module measure ambient temperatures without hassle. This board includes a reset push button. The board can be powered from any of our battery holders.

On the firmware side, we provide the Arduino sketch that works with this board. This sketch is RF-compatible and interoperable with the rest of SWAP sketches and applications.

Pinout and ports



Hardware specifications

- Size: 1.0 x 2.2 in (25.4 x 55.8 mm)
- Pin spacing: 0.1 in (2.54 mm)
- On-board 10KOhm NTC temperature sensor
- On-board reset button
- Input voltage (Vcc): 2.0 to 3.6 VDC

Battery-holders

A variety of battery-holders can be used with this carrier board. You can choose your preferred one from Accessories.

Links

- Schematics in PDF format
- Link to the wiki page

Anexo D

Datasheet Acelerometro MPU6050

	MPU-6000/MPU-6050 Product Specification	Document Number: PS-MPU-6000A-00 Revision: 3.4 Release Date: 08/19/2013
---	--	---

6.2 Accelerometer Specifications

VDD = 2.375V-3.46V, VLOGIC (MPU-6050 only) = 1.8V±5% or VDD, T_A = 25°C

PARAMETER	CONDITIONS	MIN	TYP	MAX	UNITS	NOTES
ACCELEROMETER SENSITIVITY						
Full-Scale Range	AFS_SEL=0 AFS_SEL=1 AFS_SEL=2 AFS_SEL=3		±2 ±4 ±8 ±16		g g g g	
ADC Word Length	Output in two's complement format		16		bits	
Sensitivity Scale Factor	AFS_SEL=0 AFS_SEL=1 AFS_SEL=2 AFS_SEL=3		16,384 8,192 4,096 2,048		LSB/g LSB/g LSB/g LSB/g	
Initial Calibration Tolerance			±3		%	
Sensitivity Change vs. Temperature	AFS_SEL=0, -40°C to +85°C		±0.02		%/°C	
Nonlinearity	Best Fit Straight Line		0.5		%	
Cross-Axis Sensitivity			±2		%	
ZERO-G OUTPUT						
Initial Calibration Tolerance	X and Y axes Z axis		±50 ±80		mg mg	1
Zero-G Level Change vs. Temperature	X and Y axes, 0°C to +70°C Z axis, 0°C to +70°C		±35 ±60		mg	
SELF TEST RESPONSE						
Relative	Change from factory trim	-14		14	%	2
NOISE PERFORMANCE						
Power Spectral Density	@10Hz, AFS_SEL=0 & ODR=1kHz		400		µg/√Hz	
LOW PASS FILTER RESPONSE						
	Programmable Range	5		260	Hz	
OUTPUT DATA RATE						
	Programmable Range	4		1,000	Hz	
INTELLIGENCE FUNCTION INCREMENT			32		mg/LSB	

1. Typical zero-g initial calibration tolerance value after MSL3 preconditioning
2. Please refer to the following document for further information on Self-Test: *MPU-6000/MPU-6050 Register Map and Descriptions*

Anexo E

Datasheet módulo GSM/GPRS SIM900

SIM900

The GSM/GPRS Module for M2M applications

General features

- Quad-Band 850/900/1800/1900 MHz
- GPRS multi-slot class 10/8
- GPRS mobile station class B
- Compliant to GSM phase 2/2+
 - Class 4 (2 W@850/900 MHz)
 - Class 1 (1 W@1800/1900MHz)
- Dimensions: 24*24*3 mm
- Weight: 3.4g
- Control via AT commands (GSM 07.07, 07.05 and SIMCOM enhanced AT Commands)
- SIM application toolkit
- Supply voltage range 3.4 ... 4.5 V
- Low power consumption
- Operation temperature: -30 °C to +60 °C

Specifications for fax

- Group 3, class 1

Specifications for data

- GPRS class 10: max. 85.6 kbps (downlink)
- PBCCH support
- Coding schemes CS 1, 2, 3, 4
- CSD up to 14.4 kbps
- USSD
- Non transparent mode
- PPP-stack

Interfaces

- Hands-free operation
- Echo suppression**
- AMR
 - Half Rate(HR)
 - Full Rate(FR)
- Interface to external SIM 3V/1.8V
- analog audio interface
- RTC backup
- SPI interface
- Serial interface
- Antenna pad
- I2C
- GPIO
- PWM
- ADC

Compatibility

- AT cellular command interface

Approvals (in planning)

- CE
- FCC
- ROHS
- PTCRB
- GCF
- AT&T
- IC
- TA

Specifications for SMS via GSM Pin Assignment

/ GPRS

- Point-to-point MO and MT
- SMS cell broadcast
- Text and PDU mode

Drivers

- MUX Driver

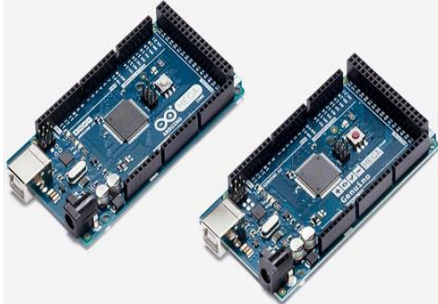
Specifications for voice

- Tricodex
 - Half rate (HR)
 - Full rate (FR)
 - Enhanced Full rate (EFR)

The diagram shows a square module with pins on all four sides. The top edge has 26 pins labeled with numbers 1 through 26. The right edge has 12 pins labeled with numbers 1 through 12. The bottom edge has 26 pins labeled with numbers 1 through 26. The left edge has 12 pins labeled with numbers 1 through 12. Specific pins are labeled with names: GND, VCC, SIM_V, SIM_I, SIM_R, SIM_P, SIM_C, SIM_D, SIM_E, SIM_F, SIM_G, SIM_H, SIM_I, SIM_J, SIM_K, SIM_L, SIM_M, SIM_N, SIM_O, SIM_P, SIM_Q, SIM_R, SIM_S, SIM_T, SIM_U, SIM_V, SIM_W, SIM_X, SIM_Y, SIM_Z, SIM_AA, SIM_AB, SIM_AC, SIM_AD, SIM_AE, SIM_AF, SIM_AG, SIM_AH, SIM_AI, SIM_AJ, SIM_AK, SIM_AL, SIM_AM, SIM_AN, SIM_AO, SIM_AP, SIM_AQ, SIM_AR, SIM_AS, SIM_AT, SIM_AU, SIM_AV, SIM_AW, SIM_AX, SIM_AY, SIM_AZ, SIM_BA, SIM_BB, SIM_BC, SIM_BD, SIM_BE, SIM_BF, SIM_BG, SIM_BH, SIM_BI, SIM_BJ, SIM_BK, SIM_BL, SIM_BM, SIM_BN, SIM_BO, SIM_BP, SIM_BQ, SIM_BR, SIM_BS, SIM_BT, SIM_BU, SIM_BV, SIM_BW, SIM_BX, SIM_BY, SIM_BZ, SIM_CA, SIM_CB, SIM_CC, SIM_CD, SIM_CE, SIM_CF, SIM_CG, SIM_CH, SIM_CI, SIM_CJ, SIM_CK, SIM_CL, SIM_CM, SIM_CN, SIM_CO, SIM_CP, SIM_CQ, SIM_CR, SIM_CS, SIM_CT, SIM_CU, SIM_CV, SIM_CW, SIM_CX, SIM_CY, SIM_CZ, SIM_DA, SIM_DB, SIM_DC, SIM_DD, SIM_DE, SIM_DF, SIM_DG, SIM_DH, SIM_DI, SIM_DJ, SIM_DK, SIM_DL, SIM_DM, SIM_DN, SIM_DO, SIM_DP, SIM_DQ, SIM_DR, SIM_DS, SIM_DT, SIM_DU, SIM_DV, SIM_DW, SIM_DX, SIM_DY, SIM_DZ, SIM_EA, SIM_EB, SIM_EC, SIM_ED, SIM_EE, SIM_EF, SIM_EG, SIM_EH, SIM_EI, SIM_EJ, SIM_EK, SIM_EL, SIM_EM, SIM_EN, SIM_EO, SIM_EP, SIM_EQ, SIM_ER, SIM_ES, SIM_ET, SIM_EU, SIM_EV, SIM_EW, SIM_EX, SIM_EY, SIM_EZ, SIM_FA, SIM_FB, SIM_FC, SIM_FD, SIM_FE, SIM_FF, SIM_FG, SIM_FH, SIM_FI, SIM_FJ, SIM_FK, SIM_FL, SIM_FM, SIM_FN, SIM_FO, SIM_FP, SIM_FQ, SIM_FR, SIM_FS, SIM_FT, SIM_FU, SIM_FV, SIM_FW, SIM_FX, SIM_FY, SIM_FZ, SIM_GA, SIM_GB, SIM_GC, SIM_GD, SIM_GE, SIM_GF, SIM_GG, SIM_GH, SIM_GI, SIM_GJ, SIM_GK, SIM_GL, SIM_GM, SIM_GN, SIM_GO, SIM_GP, SIM_GQ, SIM_GR, SIM_GS, SIM_GT, SIM_GU, SIM_GV, SIM_GW, SIM_GX, SIM_GY, SIM_GZ, SIM_HA, SIM_HB, SIM_HC, SIM_HD, SIM_HE, SIM_HF, SIM_HG, SIM_HH, SIM_HI, SIM_HJ, SIM_HK, SIM_HL, SIM_HM, SIM_HN, SIM_HO, SIM_HP, SIM_HQ, SIM_HR, SIM_HS, SIM_HT, SIM_HU, SIM_HV, SIM_HW, SIM_HX, SIM_HY, SIM_HZ, SIM_IA, SIM_IB, SIM_IC, SIM_ID, SIM_IE, SIM_IF, SIM_IG, SIM_IH, SIM_II, SIM_IJ, SIM_IK, SIM_IL, SIM_IM, SIM_IN, SIM_IO, SIM_IP, SIM_IQ, SIM_IR, SIM_IS, SIM_IT, SIM_IU, SIM_IV, SIM_IW, SIM_IX, SIM_IY, SIM_IZ, SIM_JA, SIM_JB, SIM_JC, SIM_JD, SIM_JE, SIM_JF, SIM_JG, SIM_JH, SIM_JI, SIM_JJ, SIM_JK, SIM_JL, SIM_JM, SIM_JN, SIM_JO, SIM_JP, SIM_JQ, SIM_JR, SIM_JS, SIM_JT, SIM_JU, SIM_JV, SIM_JW, SIM_JX, SIM_JY, SIM_JZ, SIM_KA, SIM_KB, SIM_KC, SIM_KD, SIM_KE, SIM_KF, SIM_KG, SIM_KH, SIM_KI, SIM_KJ, SIM_KK, SIM_KL, SIM_KM, SIM_KN, SIM_KO, SIM_KP, SIM_KQ, SIM_KR, SIM_KS, SIM_KT, SIM_KU, SIM_KV, SIM_KW, SIM_KX, SIM_KY, SIM_KZ, SIM_LA, SIM_LB, SIM_LC, SIM_LD, SIM_LE, SIM_LF, SIM_LG, SIM_LH, SIM_LI, SIM_LJ, SIM_LK, SIM_LL, SIM_LM, SIM_LN, SIM_LO, SIM_LP, SIM_LQ, SIM_LR, SIM_LS, SIM_LT, SIM_LU, SIM_LV, SIM_LW, SIM_LX, SIM_LY, SIM_LZ, SIM_MA, SIM_MB, SIM_MC, SIM_MD, SIM_ME, SIM_MF, SIM_MG, SIM_MH, SIM_MI, SIM_MJ, SIM_MK, SIM_ML, SIM_MM, SIM_MN, SIM_MO, SIM_MP, SIM_MQ, SIM_MR, SIM_MS, SIM_MT, SIM_MU, SIM_MV, SIM_MW, SIM_MX, SIM_MY, SIM_MZ, SIM_NA, SIM_NB, SIM_NC, SIM_ND, SIM_NE, SIM_NF, SIM_NG, SIM_NH, SIM_NI, SIM_NJ, SIM_NK, SIM_NL, SIM_NM, SIM>NN, SIM_NO, SIM_NP, SIM_NQ, SIM_NR, SIM_NS, SIM_NT, SIM_NU, SIM_NV, SIM_NW, SIM_NX, SIM_NY, SIM_NZ, SIM_OA, SIM_OB, SIM_OC, SIM_OD, SIM_OE, SIM_OF, SIM_OG, SIM_OH, SIM_OI, SIM_OJ, SIM_OK, SIM_OL, SIM_OM, SIM_ON, SIM_OO, SIM_OP, SIM_OQ, SIM_OR, SIM_OS, SIM_OT, SIM_OU, SIM_OV, SIM_OW, SIM_OX, SIM_OY, SIM_OZ, SIM_PA, SIM_PB, SIM_PC, SIM_PD, SIM_PE, SIM_PF, SIM_PG, SIM_PH, SIM_PI, SIM_PJ, SIM_PK, SIM_PL, SIM_PM, SIM_PN, SIM_PO, SIM_PP, SIM_PQ, SIM_PR, SIM_PS, SIM_PT, SIM_PU, SIM_PV, SIM_PW, SIM_PX, SIM_PY, SIM_PZ, SIM_QA, SIM_QB, SIM_QC, SIM_QD, SIM_QE, SIM_QF, SIM_QG, SIM_QH, SIM_QI, SIM_QJ, SIM_QK, SIM_QL, SIM_QM, SIM_QN, SIM_QO, SIM_QP, SIM_QQ, SIM_QR, SIM_QS, SIM_QT, SIM_QU, SIM_QV, SIM_QW, SIM_QX, SIM_QY, SIM_QZ, SIM_RA, SIM_RB, SIM_RC, SIM_RD, SIM_RE, SIM_RF, SIM_RG, SIM_RH, SIM_RI, SIM_RJ, SIM_RK, SIM_RL, SIM_RM, SIM_RN, SIM_RO, SIM_RP, SIM_RQ, SIM_RR, SIM_RS, SIM_RT, SIM_RU, SIM_RV, SIM_RW, SIM_RX, SIM_RY, SIM_RZ, SIM_SA, SIM_SB, SIM_SC, SIM_SD, SIM_SE, SIM_SF, SIM_SG, SIM_SH, SIM_SI, SIM_SJ, SIM_SK, SIM_SL, SIM_SM, SIM_SN, SIM_SO, SIM_SP, SIM_SQ, SIM_SR, SIM_SS, SIM_ST, SIM_SU, SIM_SV, SIM_SW, SIM_SX, SIM_SY, SIM_SZ, SIM_TA, SIM_TB, SIM_TC, SIM_TD, SIM_TE, SIM_TF, SIM_TG, SIM_TH, SIM_TI, SIM_TJ, SIM_TK, SIM_TL, SIM_TM, SIM_TN, SIM_TO, SIM_TP, SIM_TQ, SIM_TR, SIM_TS, SIM_TT, SIM_TU, SIM_TV, SIM_TW, SIM_TX, SIM_TY, SIM_TZ, SIM_UA, SIM_UB, SIM_UC, SIM_UD, SIM_UE, SIM_UF, SIM_UG, SIM_UH, SIM_UI, SIM_UJ, SIM_UK, SIM_UL, SIM_UM, SIM_UN, SIM_UO, SIM_UP, SIM_UQ, SIM_UR, SIM_US, SIM_UT, SIM_UU, SIM_UV, SIM_UW, SIM_UX, SIM_UY, SIM_UZ, SIM_VA, SIM_VB, SIM_VC, SIM_VD, SIM_VE, SIM_VF, SIM_VG, SIM_VH, SIM_VI, SIM_VJ, SIM_VK, SIM_VL, SIM_VM, SIM_VN, SIM_VO, SIM_VP, SIM_VQ, SIM_VR, SIM_VS, SIM_VT, SIM_VU, SIM_VV, SIM_VW, SIM_VX, SIM_VY, SIM_VZ, SIM_WA, SIM_WB, SIM_WC, SIM_WD, SIM_WE, SIM_WF, SIM_WG, SIM_WH, SIM_WI, SIM_WJ, SIM_WK, SIM_WL, SIM_WM, SIM_WN, SIM_WO, SIM_WP, SIM_WQ, SIM_WR, SIM_WS, SIM_WT, SIM_WU, SIM_WV, SIM_WW, SIM_WX, SIM_WY, SIM_WZ, SIM_XA, SIM_XB, SIM_XC, SIM_XD, SIM_XE, SIM_XF, SIM_XG, SIM_XH, SIM_XI, SIM_XJ, SIM_XK, SIM_XL, SIM_XM, SIM_XN, SIM_XO, SIM_XP, SIM_XQ, SIM_XR, SIM_XS, SIM_XT, SIM_XU, SIM_XV, SIM_XW, SIM_XX, SIM_XY, SIM_XZ, SIM_YA, SIM_YB, SIM_YC, SIM_YD, SIM_YE, SIM_YF, SIM_YG, SIM_YH, SIM_YI, SIM_YJ, SIM_YK, SIM_YL, SIM_YM, SIM_YN, SIM_YO, SIM_YP, SIM_YQ, SIM_YR, SIM_YS, SIM_YT, SIM_YU, SIM_YV, SIM_YW, SIM_YX, SIM_YZ, SIM_ZA, SIM_ZB, SIM_ZC, SIM_ZD, SIM_ZE, SIM_ZF, SIM_ZG, SIM_ZH, SIM_ZI, SIM_ZJ, SIM_ZK, SIM_ZL, SIM_ZM, SIM_ZN, SIM_ZO, SIM_ZP, SIM_ZQ, SIM_ZR, SIM_ZS, SIM_ZT, SIM_ZU, SIM_ZV, SIM_ZW, SIM_ZX, SIM_ZY, SIM_ZZ.

Anexo F

Datasheet módulo Arduino Mega



Arduino MEGA 2560 (USA ONLY)
& Genuino MEGA 2560 (OUTSIDE USA)

The MEGA 2560 is designed for more complex projects. With 54 digital I/O pins, 16 analog inputs and a larger space for your sketch it is the recommended board for 3D printers and robotics projects. This gives your projects plenty of room and opportunities.

Overview

5V 8-bit 16 MHz AVR

The Mega 2560 is a microcontroller board based on the [ATmega2560](#). It has 54 digital input/output pins (of which 15 can be used as PWM outputs), 16 analog inputs, 4 UARTs (hardware serial ports), a 16 MHz crystal oscillator, a USB connection, a power jack, an ICSP header, and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started. The Mega 2560 board is compatible with most shields designed for the Uno and the former boards Duemilanove or Diecimila.

The Mega 2560 is an update to the [Arduino Mega](#), which it replaces.

You can find [here](#) your board warranty informations.

Getting Started

You can find in the [Getting Started section](#) all the information you need to configure your board, use the Arduino Software (IDE), and start tinker with coding and electronics.

Technical specs

Microcontroller	ATmega2560
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limit)	6-20V
Digital I/O Pins	54 (of which 15 provide PWM output)
Analog Input Pins	16
DC Current per I/O Pin	20 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
Length	101.52 mm
Width	53.3 mm
Weight	37 g

Programming

The Mega 2560 board can be programmed with the [Arduino Software \(IDE\)](#). For details, see the [reference](#) and [tutorials](#).

The ATmega2560 on the Mega 2560 comes preprogrammed with a [bootloader](#) that allows you to upload new code to it without the use of an external hardware programmer. It communicates using the original STK500 protocol ([reference](#), [C header files](#)).

You can also bypass the bootloader and program the microcontroller through the ICSP (In-Circuit Serial Programming) header using [Arduino ISP](#) or similar; see these [instructions for details](#).

The ATmega16U2 (or 8U2 in the rev1 and rev2 boards) firmware source code is available in the [Arduino repository](#). The ATmega16U2/8U2 is loaded with a DFU bootloader, which can be activated by:

- On Rev1 boards: connecting the solder jumper on the back of the board (near the map of Italy) and then resetting the 8U2.
- On Rev2 or later boards: there is a resistor that pulling the 8U2/16U2 HWB line to ground, making it easier to put into DFU mode. You can then use [Atmel's FLIP software](#) (Windows) or the [DFU programmer](#) (Mac OS X and Linux) to load a new firmware. Or you can use the ISP header with an external programmer (overwriting the DFU bootloader). See [this user-contributed tutorial](#) for more information.

Memory

The ATmega2560 has 256 KB of flash memory for storing code (of which 8 KB is used for the bootloader), 8 KB of SRAM and 4 KB of EEPROM (which can be read and written with the [EEPROM library](#)).

Input and Output

See the mapping between Arduino pins and Atmega2560 ports:



Each of the 54 digital pins on the Mega can be used as an input or output, using [pinMode\(\)](#), [digitalWrite\(\)](#), and [digitalRead\(\)](#) functions. They operate at 5 volts. Each pin can provide or receive 20 mA as recommended operating condition and has an internal pull-up resistor (disconnected by default) of 20-50 k ohm. A maximum of 40mA is the value that must not be exceeded to avoid permanent damage to the microcontroller.

In addition, some pins have specialized functions:

- Serial: 0 (RX) and 1 (TX); Serial 1: 19 (RX) and 18 (TX); Serial 2: 17 (RX) and 16 (TX); Serial 3: 15 (RX) and 14 (TX). Used to receive (RX) and transmit (TX) TTL serial data. Pins 0 and 1 are also connected to the corresponding pins of the ATmega16U2 USB-to-TTL Serial chip.
- External Interrupts: 2 (interrupt 0), 3 (interrupt 1), 18 (interrupt 5), 19 (interrupt 4), 20 (interrupt 3), and 21 (interrupt 2). These pins can be configured to trigger an interrupt on a low level, a rising or falling edge, or a change in level. See the [attachInterrupt\(\)](#) function for details.
- PWM: 2 to 13 and 44 to 46. Provide 8-bit PWM output with the [analogWrite\(\)](#) function.
- SPI: 50 (MISO), 51 (MOSI), 52 (SCK), 53 (SS). These pins support SPI communication using the [SPI library](#). The SPI pins are also broken out on the ICSP header, which is physically compatible with the Arduino /Genuino Uno and the old Duemilanove and Diecimila Arduino boards.
- LED: 13. There is a built-in LED connected to digital pin 13. When the pin is HIGH value, the LED is on, when the pin is LOW, it's off.
- TWI: 20 (SDA) and 21 (SCL). Support TWI communication using the [Wire](#)

Anexo G

Código de los nodos panstamp

➤ Código NODO1

```
emisor_5 Arduino 1.6
Archivo Editar Programa Herramientas Ayuda
emisor_5
#define LED_PIN 13
bool blinkState = false;

#define RFCHANNEL 0 // Usaremos el canal 0
int sensorValue = 0; // variable to store the value coming from the sensor
#define SYNCWORD1 0xB5 // Synchronization word, high byte
#define SYNCWORD0 0x47 // Synchronization word, low byte
#define SOURCE_ADDR 5 // Device address
int sensorPin = A0; // seleccionamos la entrada
int ledPin = 13; // select the pin for the LED

CCPACKET packet;

void setup()
{
    // Setup LED output pin
    pinMode(LED, OUTPUT);
    digitalWrite(LED, LOW);

    panstamp.radio.setChannel(RFCHANNEL);
    panstamp.radio.setSyncWord(SYNCWORD1, SYNCWORD0);
    panstamp.radio.setDevAddress(SOURCE_ADDR);
    panstamp.radio.setCCregs();
    panstamp.setHighTxPower();

    packet.length = 10;
}

void loop()
{
    packet.data[0] = SOURCE_ADDR; //ASIGNO LA DIRECCION DEL DISPOSITIVO
    packet.data[1] = accelgyro.getAcceleration(&ax, &ay, &az); //leemos valores del sensor
    while (ax<2949||sz<2949){
        digitalWrite(LED, HIGH);
        panstamp.radio.sendData(packet); //envio el valor leído si es mayor que 0.18g que para valores ADC equivale a 2949 ( 2499-(0.18*32768)/2 )
    }
    panstamp.sleepMs(); //si no ay valores superiores continua en estado de bajo consumo sin transmitir
    digitalWrite(LED, LOW);
}
```

➤ Código NODO 2

```
emisor_6 Arduino 1.6.8
Archivo Editar Programa Herramientas Ayuda
emisor_6
bool blinkState = false;

#define RFCHANNEL 0 // Usaremos el canal 0
int sensorValue = 0; // variable to store the value coming from the sensor
#define SYNCWORD1 0xB5 // Synchronization word, high byte
#define SYNCWORD0 0x47 // Synchronization word, low byte
#define SOURCE_ADDR 6 // Device address
int sensorPin = A0; // seleccionamos la entrada
int ledPin = 13; // select the pin for the LED

CCPACKET packet;

void setup()
{
    byte i;

    // Setup LED output pin
    pinMode(LED, OUTPUT);
    digitalWrite(LED, LOW);

    panstamp.radio.setChannel(RFCHANNEL);
    panstamp.radio.setSyncWord(SYNCWORD1, SYNCWORD0);
    panstamp.radio.setDevAddress(SOURCE_ADDR);
    panstamp.radio.setCCregs();
    panstamp.setHighTxPower();

    packet.length = 10;

    /* for(i=0 ; i<packet.length ; i++)
    + packet.data[i] = i;*/
}

void loop()
{
    packet.data[0] = SOURCE_ADDR; //ASIGNO LA DIRECCION DEL DISPOSITIVO
    packet.data[1] = accelgyro.getAcceleration(&ax, &ay, &az); //leemos valores del sensor
    if(ax<2949||sz<2949){
        digitalWrite(LED, HIGH);
        panstamp.radio.sendData(packet); //envio el valor leído si es mayor que 0.18g que para valores ADC equivale a 2949 ( 2499-(0.18*32768)/2 )
    }
    panstamp.sleepMs(); //si no ay valores superiores continua en estado de bajo consumo sin transmitir
    digitalWrite(LED, LOW);
}
```

➤ Código NODO 3

```
emisor_7 Arduino 1.6
Archivo Editar Programa Herramientas Ayuda
emisor_7
#define LED_PIN 13
bool blinkState = false;

#define RFCHANNEL 0 // Usaremos el canal 0
int sensorValue = 0; // variable to store the value coming from the sensor
#define SYNCWORD1 0x85 // Synchronization word, high byte
#define SYNCWORD0 0x47 // Synchronization word, low byte
#define SOURCE_ADDR 7 // Device address
int sensorPin = A0; // seleccionamos la entrada
int ledPin = 13; // select the pin for the LED

CCFACRET packet;

void setup()
{
  byte i;

  // Setup LED output pin
  pinMode(LED, OUTPUT);
  digitalWrite(LED, LOW);

  panstamp.radio.setChannel(RFCHANNEL);
  panstamp.radio.setSyncWord(SYNCWORD1, SYNCWORD0);
  panstamp.radio.setDevAddress(SOURCE_ADDR);
  panstamp.radio.setCCregs();
  panstamp.setHighTxPower();

  packet.length = 10;

  /* for(i=0 ; i<packet.length ; i++)
  * packet.data[i] = i;*/
}

void loop()
{
  packet.data[0] = SOURCE_ADDR; //ASIGNO LA DIRECCION DEL DISPOSITIVO
  packet.data[1] = accelgyro.getAcceleration(&ax, &ay, &az); //leemos valores del sensor
  if(ay<2949 || az<2949){
    digitalWrite(LED, HIGH);
    panstamp.radio.sendData(packet); //envio el valor leído si es mayor que 0.10g que para valores ADC equivale a 2949 ( 0.10*(32768)/2 )
  }
  panstamp.sleepMs(); //si no ay valores superiores continua en estado de bajo consumo sin transmitir
}
```

➤ Código GATEWAY

```
gateway_panstick
Archivo Editar Programa Herramientas Ayuda
gateway_panstick

void clasificaciondepaquetes ()
{
  if( packet.data[0]==5)
  {
    ax1=ax;
    ay1=ay;
    az1=az;
  }
  if( packet.data[0]==6)
  {
    ax2=ax;
    ay2=ay;
    az2=az;
  }
  if( packet.data[0]==5)
  {
    ax3=ax;
    ay3=ay;
    az3=az;
  }
  else
  {
    accelgyro.getAcceleration(&ax, &ay, &az); //leemos valores del sensor
  }
}

void loop() {
  Wire.beginTransmission(2); //comienza la transmision i2c al arduino que es el elemento 2
  panstamp.radio.getData(packet);
  clasificaciondepaquetes();
  Wire.write(ax);
  Wire.write(ay);
  Wire.write(az);
  Wire.write(ax1);
  Wire.write(ay1);
  Wire.write(az1);
  Wire.write(ax2);
  Wire.write(ay2);
  Wire.write(az2);
  Wire.write(ax3);
  Wire.write(ay3);
  Wire.write(az3);
}
```


Anexo H

Código en Arduino del Sield GPRS SIM900

```
bloque_de_alerta Arduino
Archivo Editar Programa Herramientas Ayuda
bloque_de_alerta
#include<Wire.h>
#include <SoftwareSerial.h>

SoftwareSerial SIM900(7, 8);

int inPin = 7; //señal digital de alerta enviada desde el gateway panstick
int comp = LOW; //variable para almacenar la señal de alerta
#define LED_FIN 13
bool blinkState = false;
int long sacusal;
int long sanderior;
int long intervalo = 60000; //se toma 10 minutos para esperar a que termine el movimiento para dar la siguiente alerta en caso de producirse.
int16_t ax, ay, az;
int16_t ax1, ay1, az1;
int16_t ax2, ay2, az2;
int16_t ax3, ay3, az3;
float acx, acy, acz;
float acx1, acy1, acz1;
float acx2, acy2, acz2;
float acoc, acoc, acoc;
float acoc1, acoc1, acoc1;
float acoc2, acoc2, acoc2;
float acoc3, acoc3, acoc3;
float pacoc, pacoc;

void setup() {
  SIM900.begin(19200); //Iniciando el shield gprs
  SIM900power();
  delay(200);
  Wire.begin(4);
  Serial.begin(9600);
  Serial1.begin(9600);
}

// **FUNCION PARA EL ENCENDIDO DEL MODULO GPRS**
void SIM900power()
//es equivalente a presionar el boton de encendido
{
  digitalWrite(9, HIGH);
  delay(100);
  digitalWrite(9, LOW);
  delay(200);
}
```

```
bloque_de_alerta Arduino 1.6.
Archivo Editar Programa Herramientas Ayuda
bloque_de_alerta

// **FUNCION PARA EL ENVIO DE LA ALERTA ATRAVES DE SMS**
void sendSMS()
{
  SIM900.print("AT+CMGF=1\r"); // comando AT para enviar un SMS
  delay(200);
  SIM900.print("AT + CMSS = \"+593980229278\r"); // numero del movil 1 receptor, en formato internacional
  delay(200);
  SIM900.println("Existe riesgo de movimiento sismico, por favor tome las debidas precauciones.\n --SISTEMA DE ALERTA TEMPRANA DE SISMOS ESPOCH--");
  delay(200);
  SIM900.print((char)26); // termina los comandos AT con un ^Z, código ASCII 26
  delay(200);
  SIM900.println();
  delay(500); // da tiempo al modulo para enviar el sms
  SIM900.print("AT+CMGF=1\r"); // comando AT para enviar un SMS
  delay(200);
  SIM900.print("AT + CMSS = \"+593994421690\r"); // numero del movil 2 receptor, en formato internacional
  delay(200);
  SIM900.println("Existe riesgo de movimiento sismico, por favor tome las debidas precauciones.\n --SISTEMA DE ALERTA TEMPRANA DE SISMOS ESPOCH--");
  delay(200);
  SIM900.print((char)26); // termina los comandos AT con un ^Z, código ASCII 26
  delay(200);
  SIM900.println();
  delay(500); // da tiempo al modulo para enviar el sms
  SIM900power(); // apaga el módulo
}

// **FUNCION PARA CONVERTIR LOS VALORES ADC RECIBIDOS EN MAGNITUDES FÍSICAS
void lecturaYconversion()
{
  Wire.read();
  acx=ax; //pasamos los valores int de 16 bits a float
  acy=ay;
  acz=az;
  acx1=ax1;
  acy1=ay1;
  acz1=az1;
  acx2=ax2;
  acy2=ay2;
  acz2=az2;
  acx3=ax3;
  acy3=ay3;
  acz3=az3;
  acoc=(acx/32768)*(2); //transformamos los valores a magnitudes físicas (g)
  acoc=(acy/32768)*(2);
}
```



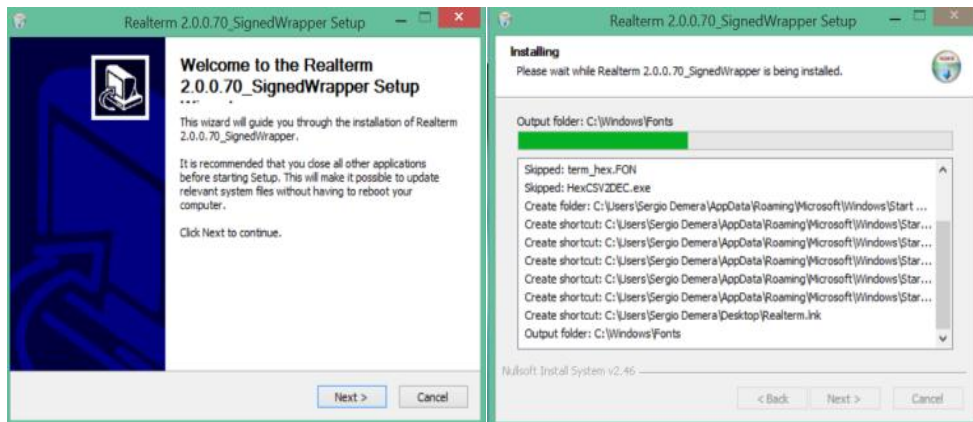
```

bloque_de_alerta
acc2=((acc2/32768)*(2));
acc3=((acc3/32768)*(2));
accy3=((accy3/32768)*(2));
accz3=((accz3/32768)*(2));
paccy=( accy+accy1+accy2+accy3)/4;
paccz=( accz+accz1+accz2+accz3)/4;
}
void comparacion()
{
if (paccy>0.2||paccz>0.2){
comp=HIGH;
}
}
// **FUNCION PARA MOSTRAR DATOS POR FUERIO SERIAL
void impresionserial()
{
Serial.print("ax0: ");Serial.print(accx); Serial.print("\n");
Serial.print("ay0: ");Serial.print(accy); Serial.print("\n");
Serial.print("az0: ");Serial.println(accz);
Serial.print("ax1: ");Serial.print(accx1); Serial.print("\n");
Serial.print("ay1: ");Serial.print(accy1); Serial.print("\n");
Serial.print("az1: ");Serial.println(accz1);
Serial.print("ax2: ");Serial.print(accx2); Serial.print("\n");
Serial.print("ay2: ");Serial.print(accy2); Serial.print("\n");
Serial.print("az2: ");Serial.println(accz2);
Serial.print("ax3: ");Serial.print(accx3); Serial.print("\n");
Serial.print("ay3: ");Serial.print(accy3); Serial.print("\n");
Serial.print("az3: ");Serial.println(accz3);
}
// **BUCLE PRINCIPAL**
void loop() {
lecturayconversion();
impresionserial();
comparacion();
if(comp==HIGH){
tactual=millis();
if(tactual-tanterior>intervalo){
sendSMS();
tanterior=tactual;
}
}
delay(5); // se hacen lecturas cada 5ms para tener una frecuencia de muestreo de 200Hz ya que la frecuencia de una onda sismica va de 1 a 100 Hz aprox
}
    
```

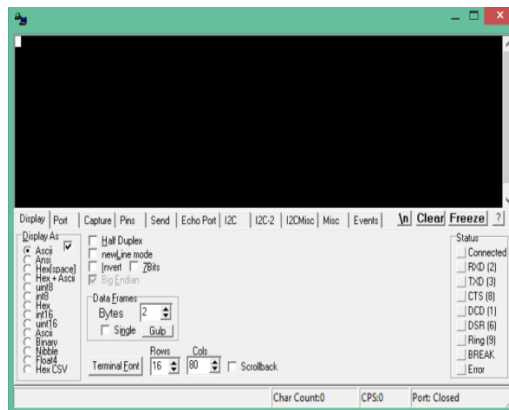
Anexo I

Configuración de Realterm

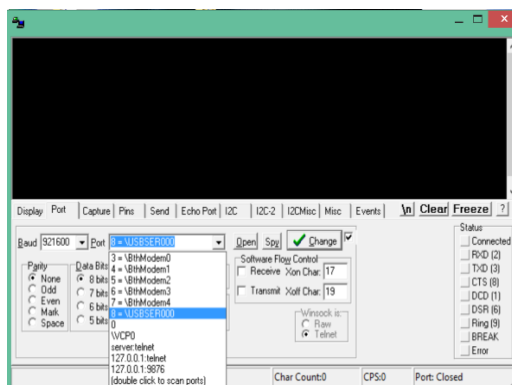
Para poder realizar la captura de los datos es necesario primero instalar el programa



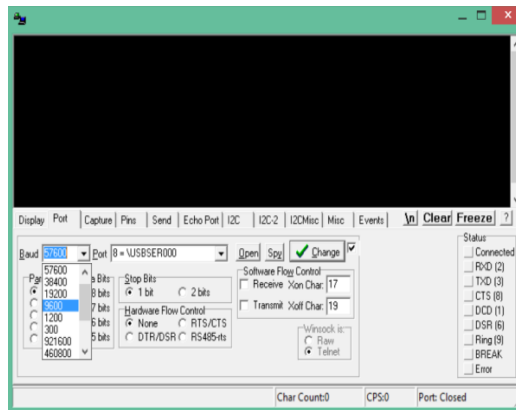
Al iniciarlo se procede a configurarlo de la siguiente manera:



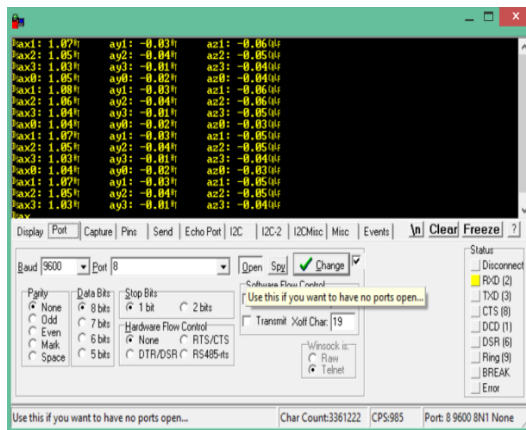
En la pestaña Port se selecciona el puerto serial que corresponde al Arduino.



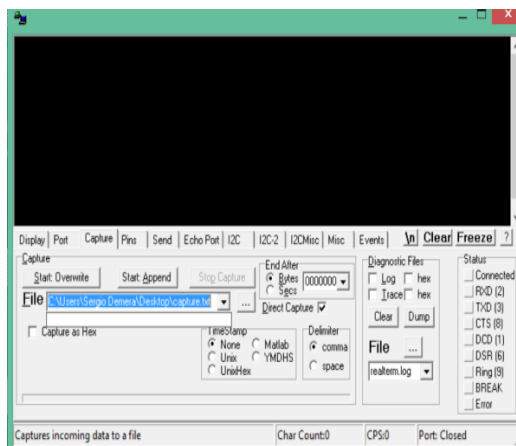
Se define la tasa de símbolos por segundo en 9600 Baudios.



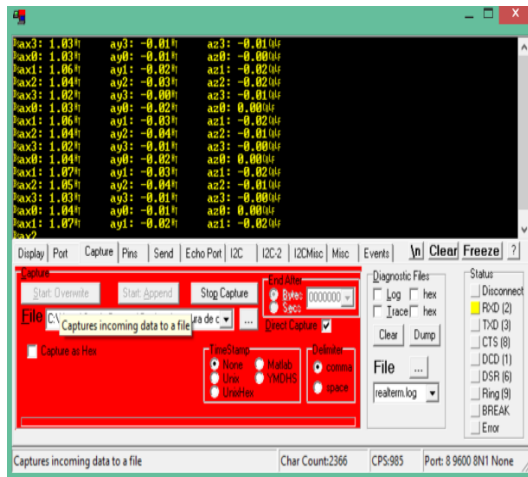
Se selecciona “open” para abrir el puerto y obtener los valores de los sensores, se puede notar como se reciben los datos y se visualizan, al igual que el estado de la conexión en la barra inferior izquierda.



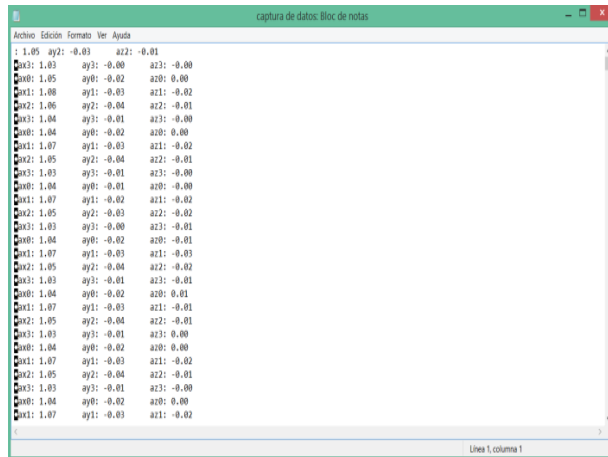
Lo siguiente es pasar a la pestaña “capture”, donde vamos a seleccionar la ruta del archivo .txt que será generado automáticamente y en el cual se escribirán los datos obtenidos del puerto serial.



Por último solo queda iniciar la captura haciendo click sobre el botón “Start Overwrite”



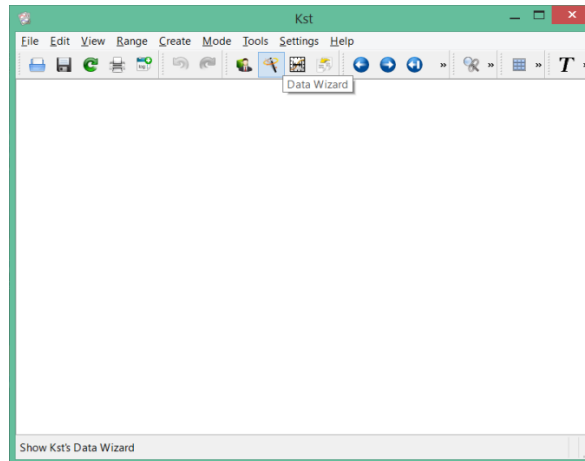
Los valores recibidos en el puerto serial pueden ser visualizados inmediatamente en el archivo de texto que se ha creado.



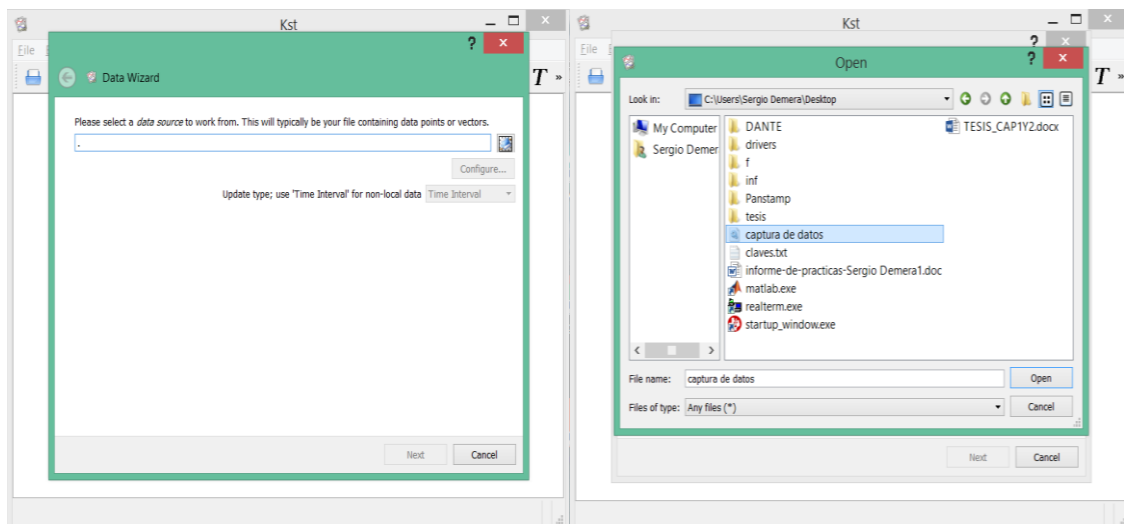
Anexo J

Configuración de Kst

Iniciamos el programa Kst



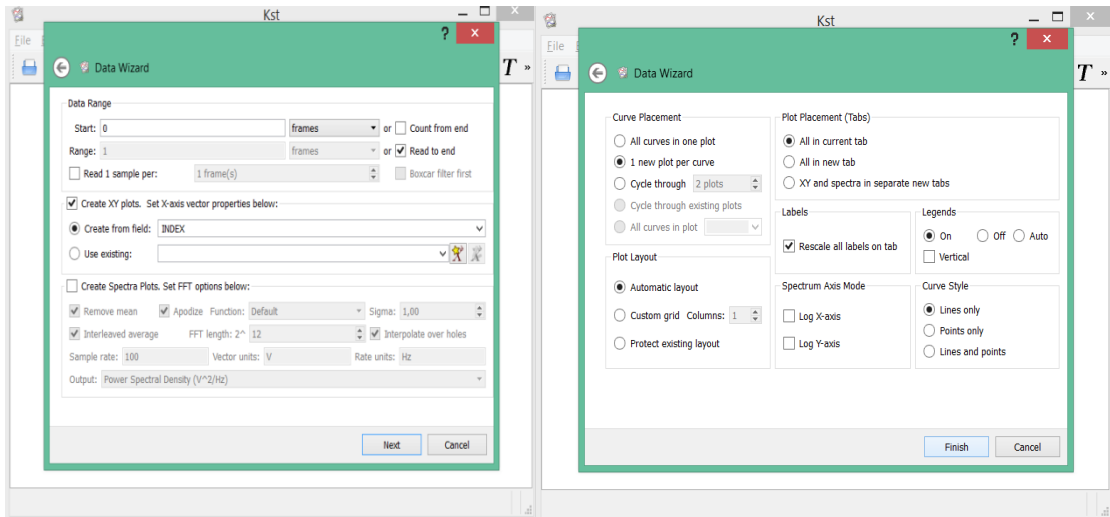
Primero seleccionamos el archivo .txt que contiene los datos que son capturados del puerto serial con el programa Realterm en tiempo real.



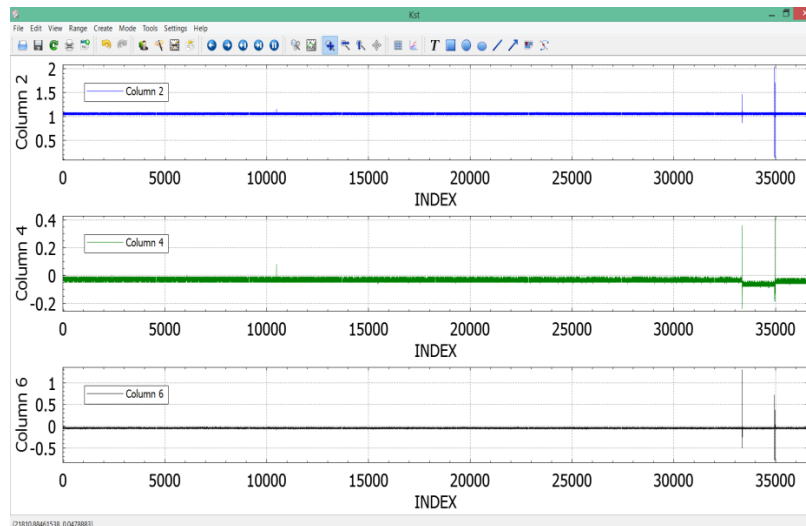
Lo que sigue es seleccionar las columnas que contienen los valores numéricos que van a ser graficados, para este caso son la segunda, cuarta y sexta columna.



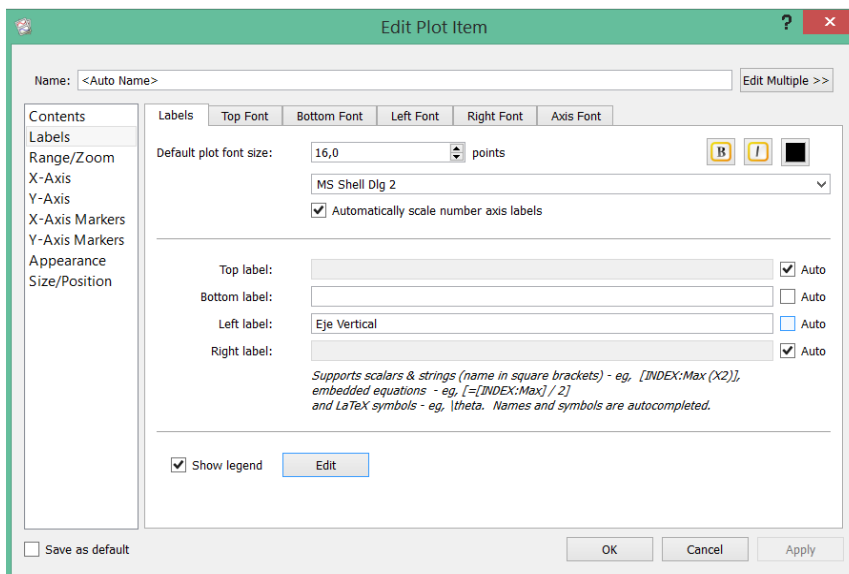
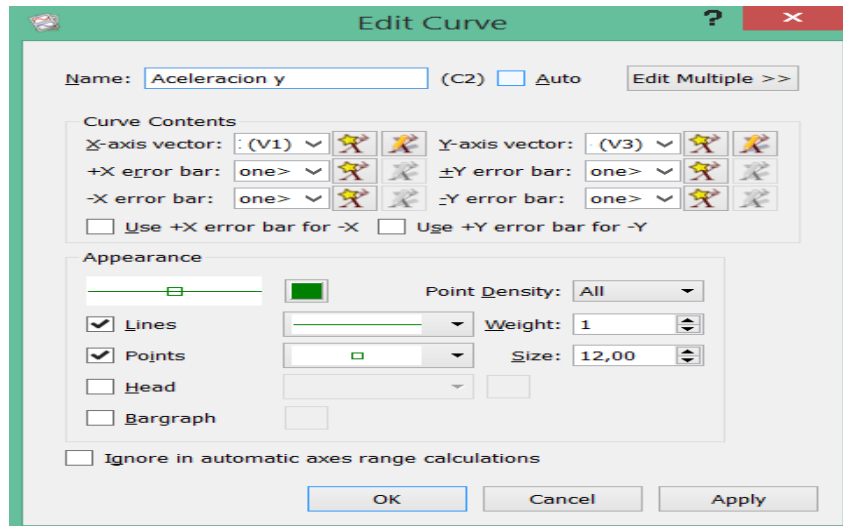
A partir de eso dejamos las configuraciones por defecto proporcionadas por el programa.



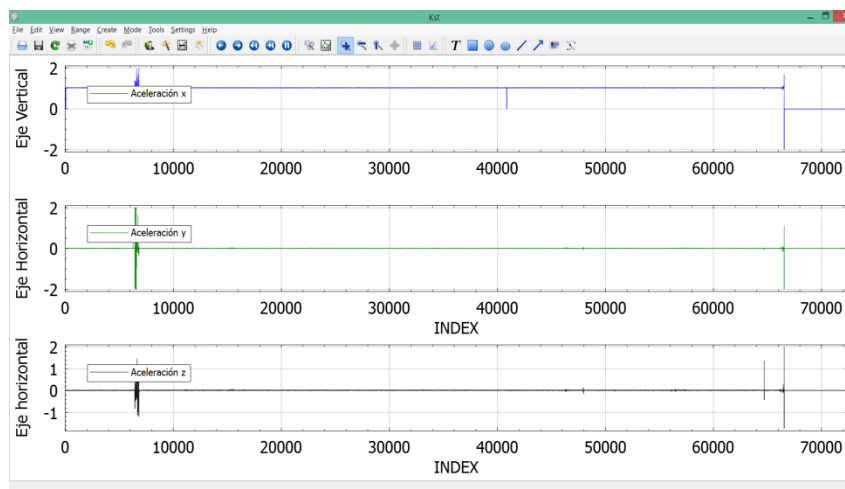
El programa empezara con la graficación de lo que se ha registrado hasta ese momento con escala automática, la cual varía de acuerdo a los valores que se obtienen.



Las configuraciones adicionales que se deben hacer para mostrar apropiadamente los datos son cambiar las etiquetas en los ejes y el tiempo. Esto los hacemos dando doble click sobre las etiquetas y desmarcando los valores por defecto y colocando una mejor descripción que facilite la interpretación.



Y queda finalmente un gráfico fácil de interpretar:



Muchos otros parámetros pueden ser configurados como el intervalo y el rango que por defecto tienen escala automática.

Anexo K

Programa para calibrar acelerómetro MPU6050

```
calibracion_acelerometro

// Librerías I2C para controlar el mpu6050
// la librería MPU6050.h necesita I2Cdev.h, I2Cdev.h necesita Wire.h
#include "I2Cdev.h"
#include "MPU6050.h"
#include "Wire.h"

// La dirección del MPU6050 puede ser 0x68 o 0x69, dependiendo
// del estado de ADO. Si no se especifica, 0x68 estará implícito
MPU6050 sensor;

// Valores RAW (sin procesar) del acelerómetro y giroscopio en los ejes x,y,z
int ax, ay, az;
int gx, gy, gz;

// Variables usadas por el filtro pasa bajos
long f_ax, f_ay, f_az;
int p_ax, p_ay, p_az;
long f_gx, f_gy, f_gz;
int p_gx, p_gy, p_gz;
int counter=0;

// Valor de los offsets
int ax_o, ay_o, az_o;
int gx_o, gy_o, gz_o;

void setup() {
  Serial.begin(57600); //Iniciando puerto serial
  Wire.begin(); //Iniciando I2C
  sensor.initialize(); //Iniciando el sensor

  if (sensor.testConnection()) Serial.println("Sensor iniciado correctamente");

  // Leer los offset los offsets anteriores
  ax_o=sensor.getXAccelOffset();
  ay_o=sensor.getYAccelOffset();
  az_o=sensor.getZAccelOffset();
  gx_o=sensor.getXGyroOffset();
  gy_o=sensor.getYGyroOffset();
  gz_o=sensor.getZGyroOffset();

  Serial.println("Offsets:");
  Serial.print(ax_o); Serial.print("\t");
  Serial.print(ay_o); Serial.print("\t");
  Serial.print(az_o); Serial.print("\t");
  Serial.print(gx_o); Serial.print("\t");
  Serial.print(gy_o); Serial.print("\t");
  Serial.print(gz_o); Serial.print("\t");
}
```

```
calibracion_acelerometro
Serial.println("Offsets:");
Serial.print(ax_0); Serial.print("\t");
Serial.print(ay_0); Serial.print("\t");
Serial.print(az_0); Serial.print("\t");
Serial.print(gx_0); Serial.print("\t");
Serial.print(gy_0); Serial.print("\t");
Serial.print(gz_0); Serial.print("\t");
Serial.println("\nEnvie cualquier caracter para empezar la calibracionnn");
// Espera un caracter para empezar a calibrar
while (true) if (Serial.available()) break;
Serial.println("Calibrando, no mover IMU");
}

void loop() {
// Leer las aceleraciones y velocidades angulares
sensor.getAcceleration(&ax, &ay, &az);
sensor.getRotation(&gx, &gy, &gz);

// Filtrar las lecturas
f_ax = f_ax - (f_ax >> 5) + ax;
p_ax = f_ax >> 5;

f_ay = f_ay - (f_ay >> 5) + ay;
p_ay = f_ay >> 5;

f_az = f_az - (f_az >> 5) + az;
p_az = f_az >> 5;

f_gx = f_gx - (f_gx >> 3) + gx;
p_gx = f_gx >> 3;

f_gy = f_gy - (f_gy >> 3) + gy;
p_gy = f_gy >> 3;

f_gz = f_gz - (f_gz >> 3) + gz;
p_gz = f_gz >> 3;

//Cada 100 lecturas corregir el offset
if (counter==100) {
//Mostrar las lecturas separadas por un [tab]
Serial.print("promedio:"); Serial.print("\t");
Serial.print(p_ax); Serial.print("\t");
Serial.print(p_ay); Serial.print("\t");
Serial.print(p_az); Serial.print("\t");
}
}
```

```
calibracion
Archivo Editar Programa Herramientas Ayuda
calibracion_acelerometro
p_gy = f_gy >> 3;

f_gz = f_gz - (f_gz >> 3) + gz;
p_gz = f_gz >> 3;

//Cada 100 lecturas corregir el offset
if (counter==100) {
//Mostrar las lecturas separadas por un [tab]
Serial.print("promedio:"); Serial.print("\t");
Serial.print(p_ax); Serial.print("\t");
Serial.print(p_ay); Serial.print("\t");
Serial.print(p_az); Serial.print("\t");
Serial.print(p_gx); Serial.print("\t");
Serial.print(p_gy); Serial.print("\t");
Serial.println(p_gz);

//Calibrar el acelerometro a 1g en el eje z (ajustar el offset)
if (p_ax > 0) ax_0--;
else {ax_0++;}
if (p_ay > 0) ay_0--;
else {ay_0++;}
if (p_az - 16384 > 0) az_0--;
else {az_0++;}

sensor.setXAccelOffset(ax_0);
sensor.setYAccelOffset(ay_0);
sensor.setZAccelOffset(az_0);

//Calibrar el giroscopio a 0°/s en todos los ejes (ajustar el offset)
if (p_gx > 0) gx_0--;
else {gx_0++;}
if (p_gy > 0) gy_0--;
else {gy_0++;}
if (p_gz > 0) gz_0--;
else {gz_0++;}

sensor.setXGyroOffset(gx_0);
sensor.setYGyroOffset(gy_0);
sensor.setZGyroOffset(gz_0);

counter=0;
}
counter++;
}
```