

PONTIFICIA UNIVERSIDAD CATÓLICA DE VALPARAÍSO – CHILE
ESCUELA DE INGENIERÍA ELÉCTRICA



**DETECCIÓN E IDENTIFICACIÓN DE PRESENCIA UTILIZANDO
MÓDULOS XBEE**

Rodolfo Antonio Flores Bravo

INFORME FINAL DEL PROYECTO
PRESENTADO EN CUMPLIMIENTO
DE LOS REQUISITOS PARA OPTAR
AL TÍTULO PROFESIONAL DE
INGENIERO ELECTRÓNICO

Diciembre de 2017

DETECCIÓN E IDENTIFICACIÓN DE PRESENCIA UTILIZANDO MÓDULOS XBEE

INFORME FINAL

Presentado en cumplimiento de los requisitos para optar al título profesional de Ingeniero Electrónico otorgado por la Escuela de Ingeniería Eléctrica de la Pontificia Universidad Católica de Valparaíso

Rodolfo Antonio Flores Bravo

Profesor guía:	Sr. Emilio Quezada Valencia
Profesor Correferente:	Sr. Guillermo Fernández Segovia
Profesor Correferente:	Sr. Raimundo Villarroel Valencia

Diciembre, 2017

ACTA DE APROBACIÓN

La comisión calificadora designada por la Escuela de Ingeniería Eléctrica ha aprobado el texto del Informe Final del Proyecto de Titulación, desarrollado entre el primer semestre de 2008 y segundo semestres del 2008, y denominado:

DETECCIÓN E IDENTIFICACIÓN DE PRESENCIA UTILIZANDO MÓDULOS XBEE

Presentado por el Señor
Rodolfo Antonio Flores Bravo

Sr. Emilio Quezada Valencia
Profesor Guía

Sr. Guillermo Fernández Segovia
Segundo Revisor

Sr. Sebastián Fingerhuth
Secretario Académico

Valparaíso, diciembre de 2017

A mi madre, mi mujer e hija. Sus manos amables, su abrazo acogedor, su infinita paciencia e incondicional apoyo. Que éste y todos mis futuros logros sean para ellas.

DETECCIÓN E IDENTIFICACIÓN DE PRESENCIA UTILIZANDO MÓDULOS XBEE

Rodolfo Antonio Flores Bravo

Profesor Guía Sr. Emilio Quezada Valencia

RESUMEN

Con el propósito de contribuir a las tecnologías domóticas, en el presente Proyecto, se diseña, construye y evalúa un sistema de Detección e Identificación de Presencia de personas dentro de un área determinada.

Las unidades desarrolladas, integran componentes electrónicos de última generación, y la programación apropiada, para que sean factibles de incorporar a los hogares. Se caracterizan por ser fáciles de usar y por sus capacidades de interconexión para conformar una red domótica domiciliaria.

La información adquirida se maneja a través de una red inalámbrica que, además de un servidor, utiliza los dispositivos denominados Zigbee incorporados en los módulos remotos.

Se hace la evaluación económica del Proyecto considerando que se dispone del apoyo de Capital Semilla de la Corporación de Fomento. El resultado que se obtiene, indica que, para una tasa interna de retorno aceptable, el precio del producto excede al que espera el mercado objetivo. En consecuencia se concluye que, hoy en día, es inviable la comercialización del sistema domótico desarrollado. Ello, a pesar de las innovadoras y ventajosas características experimentadas en el Proyecto.

ÍNDICE

RESUMEN.....	1
ÍNDICE.....	2
GLOSARIO DE TÉRMINOS.....	5
LISTADO DE FIGURAS	7
LISTADO DE TABLAS.....	9
INTRODUCCIÓN	10
CAPÍTULO 1: DEFINICIÓN DEL PROBLEMA Y PLANTEAMIENTO DE UNA SOLUCIÓN.....	13
1.1 DEFINICIÓN DEL PROBLEMA.....	13
1.2 SOLUCIÓN PROPUESTA.....	14
1.3 OBJETIVOS DEL PROYECTO.....	15
1.3.1. Objetivo General.....	15
1.3.2. Objetivos Específicos	15
1.3.3. Utilidades y fortalezas propuestas en el proyecto	15
1.4 DESCRIPCIÓN DEL SISTEMA A REALIZAR Y SU FUNCIONAMIENTO.....	16
CAPÍTULO 2: ELECCIÓN DE LA TECNOLOGÍA.....	19
2.1 POR QUÉ PIC.....	19
2.2 POR QUÉ XBEE.....	20
2.3 XBEE O XBEE PRO.....	22
2.3.1. Calidad de enlace.....	23
2.3.2 Antena.....	25

CAPÍTULO 3: DESARROLLO DE HARWARE Y SOFTWARE DEL SISTEMA....	29
3.1 INTERCONEXIÓN.....	29
3.1.1. Conexión entre XBee y PIC.....	29
3.1.2 Conexión típica del actuador.....	31
3.2 SOFTWARE	32
3.2.1 Lenguaje C.....	32
3.2.2 Lenguaje PHP (PHP INC.).....	37
3.2.3 Instrucciones y gestión MySQL.....	40
3.3 INTERCONEXIÓN GENERAL.....	41
CAPÍTULO 4. MONTAJE DEL SISTEMA.....	42
4.1 DETERMINACIÓN DE REQUISITOS DEL SISTEMA.....	42
4.1.1. Distancia entre módulos.....	42
4.1.2. Autonomía.....	43
4.1.3. Plataforma para el servidor.....	45
4.2 SOFTWARE DE SERVIDOR.....	46
4.2.1 Sistema operativo.....	46
4.2.2 Cómo se instala Linux.....	47
4.2.3 Linux como servidor.....	48
CAPÍTULO 5: MANEJO DEL SOFTWARE.....	51
5.1 ACCESO.....	51
5.1.1 Menú principal.....	51
5.1.2 Revisar actividad.....	52
5.1.3 Autorizar nuevos módulos.....	53

5.2	SEGURIDAD DEL SISTEMA.....	54
5.2.1	Seguridad de acceso con navegadores	54
5.2.2	Seguridad de acceso vía FTP	54
5.2.3	Seguridad contra uso de acceso no autorizado a administración de DB	55
CAPITULO 6: PRUEBAS EN FUNCIONAMIENTO		56
CAPÍTULO 7: EVALUACIÓN ECONÓMICA.....		58
7.1	PREMISAS.....	58
7.2	ELEMENTOS Y CONDICIONES PARA EL ANÁLISIS.....	59
7.3	ANÁLISIS DEL PERFIL ECONOMÉTRICO.....	62
7.4	FINANCIAMIENTO.....	64
7.5	ANÁLISIS FINANCIERO.....	64
CONCLUSIONES		67
BIBLIOGRAFÍA		69
APÉNDICE A: SOLICITUD DE FINANCIAMIENTO PARA CAPITAL SEMILLA CORFO.....		70
APÉNDICE B: CÁLCULO DE RENTABILIDAD A 6 CICLOS		76
APÉNDICE C: FUNCIONES DE CÓDIGO PHP PARA GESTIÓN DE INFORMACIÓN API XBEE		78
APÉNDICE D: FUNCIONES EN CÓDIGO C PARA GESTIÓN DE INFORMACIÓN API XBEE		83
APÉNDICE E: FUNCIONES EN CÓDIGO PHP PARA GESTIÓN DE INFORMACIÓN API XBEE		88
APÉNDICE F: CD DE MÁQUINA VIRTUAL CON DAMN SMALL LINUX, SERVIDOR LAMP Y SISTEMA MINICAT		93

GLOSARIO DE TÉRMINOS

ACK

Abreviatura de la palabra en inglés “Acknowledgment”, en español significa “acuse de recibo” (reconocimiento). 16

API

Siglas de “Aplication Programming Interface”. Utilizado en programación de software y se refiere al conjunto de funcione o procedimientos que ofrece cierta librería para ser utilizado por otro software. 28

Autohealing

Una manera eficiente de restablecer comunicaciones luego de una falla de un nodo o de una caída. Permite a una subred que se ha desconectad volver a la red con un mínimo de intercambio de paquetes. Esta eficiencia es conseguida mediante permitir que el nodo con problemas se reconecte estableciendo tantos enlaces como sea posible. 13

Banda libre

La FCC ha proporcionado varias bandas de licencia libre (Bandas ISM & U-NII) para que puedan ser utilizadas por la comunidad inalámbrica 15

Clusters

Computadores interconectados y coordinados para procesar determinada información actuando en conjunto 43

EEPROM

Siglas de Electrically Erasable Programmable Read-Only Memory (ROM programable y borrrable eléctricamente) 10

GPL.

Licencia Pública General de GNU o más conocida por su nombre en inglés GNU General Public License (o simplemente sus siglas del inglés GNU GPL) es la licencia más ampliamente usada en el mundo del software y garantiza a los usuarios finales (personas, organizaciones, compañías) la libertad de usar, estudiar, compartir (copiar) y modificar el software. 42

Idle	
Anglicismo para denotar algo que está en reposo y listo para operar.	12
LAMP	
Siglas en inglés de Linux, Apache, MySQL y PHP	11
PCB	
Del inglés Placa de Circuito Impreso	15
PXE	
servidor de sistemas operativos. Permite instalar un sistema operativo en cualquier computador que tenga una BIOS habilitada con capacidades PXE por medio de una conexión ethernet	43
Raspberry Pi	
Odenador de placa única de bajo costo desarrollado en Reino Unido por la fundación Rasperry Pi.	37
RF	
Siglas par Radio Frecuencias.	13
RSSI	
Siglas para “Recieved Signal Strength Indicator”, en español significa Indicador de fuerza de señal de recepción.	49
switching	
Técnica que permitiría separar para cierto tipo de flujo de datos el servidor en cuestión, aumentando considerablemente su fiabilidad.....	51
W3C	
Siglas en inglés de World Wide Web Consortium, institución internacional que produce recomendaciones para la World Wide Web que determina estándares de codificación orientada a internet.	11
x86	
abarca desde computadoras 386 y procesadores Intel/AMD de 32bits	43

LISTADO DE FIGURAS

Figura 1 - 1 Ejemplo típico de implementación	14
Figura 1 - 2. Diagrama general del sistema	16
Figura 1 - 3. Representación del módulo flotante	17
Figura 1 - 4. Representación del módulo estructural	18
Figura 1 - 5. Representación del módulo central, servidor y comunicaciones	18
Figura 2 - 1. Implementación típica de XBee y microcontrolador.....	22
Figura 2 - 2. Antena de 1/4 de largo de onda	24
Figura 2 - 3. Imagen de modelos	25
Figura 2 - 4. Pattern horizontal de XBeePro usando antena integrada, normalizada a la mayor ganancia registrada del dipolo (0dB).....	26
Figura 2 - 5. Pattern horizontal de XBeePro usando antena de alambre, normalizada a la mayor ganancia registrada por el dipolo (0dB)	26
Figura 2 - 6. Pattern horizontal de XBeePro usando antena dipolo normalizada a la mayor ganancia registrada por el dipolo (0dB)	27
Figura 2 - 7. Pattern vertical de XBee usando antena integrada (chip antenna) normalizada al mejor rendimiento obtenido	28
Figura 3 - 1. Conexión PIC y Xbee	30
Figura 3 - 2. Diagrama de comunicaciones	30
Figura 3 - 3. Conexión de un relé activado por una señal digital de salida del PIC	32
Figura 3 - 4.Función XBeeRF_str	34
Figura 3 - 5. Función XBee_str	36
Figura 3 - 6. apiOut(\$API).....	38
Figura 3 - 7. txApi(\$datos, \$addr, \$ack).....	39
Figura 3 - 8. INTERCONEXIÓN GENERAL.....	41

Figura 4 - 1. Esquema de prueba de potencia.....	42
Figura 4 - 2. Diagrama de consumo promedio para un ciclo de reposo-transmisión y recepción	43
Figura 4 - 3. Gráfico de Autonomía del sistema.....	45
Figura 4 - 4. Relación Servidor/Cliente	49
Figura 4 - 5. Herramienta XAMPP	50
Figura 5 - 1. Menú principal de gestión de sistema.....	51
Figura 5 - 2. Despliegue de actividad registrada.....	52
Figura 5 - 3. Autorizar nuevo módulo	53
Figura 6 - 1. Flujo general del sistema.....	56
Figura 7 - 1. Gráfico de Sensibilidad de TIR v/s producción inicial	63
Figura 7 - 2. Gráfico TIR en función de margen de comercialización y de unidades producidas al inicio	65

LISTADO DE TABLAS

Tabla 2 - 1. Consumo por uC en base a frecuencias de hasta 5Mhz y VDC de 5V.	19
Tabla 2 - 2. Comparativa Bluetooth y ZigBee	20
Tabla 2 - 3. Pruebas de enlace XBee v/s XBeePro	24
Tabla 3 - 1. Módulos y sus componentes	29
Tabla 7 - 1. Costos por unidad para construcción de módulo de comunicaciones MiniCat.....	60
Tabla 7 - 2. Inversión inicial	61
Tabla 7 - 3. Costos operacionales estimados	61
Tabla 7 - 4. Datos de análisis de TIR v/s Producción inicial	62
Tabla 7 - 5. TIR en función de margen de comercialización y de unidades producidas al inicio.....	65

INTRODUCCIÓN

Es difícil determinar con exactitud el origen de la domótica. Su definición aparece por primera vez a mediados de los años 50 cuando los franceses acuñaron el concepto de domotique, del latín domus (casa) unida a telemática (griego). Esto corresponde a un nuevo concepto de arquitectura orientado a desarrollar casas modernas en las que, tanto las funciones de operación como las de manutención, de ella, estén automatizadas. Lo anterior, junto con las tecnologías de la información y de las telecomunicaciones, transforma una casa o habitación en inteligente. Al considerar la complejidad de las aplicaciones que se incorporan, las nuevas tecnologías deben ser consideradas en el diseño arquitectónico de la casa e integradas a ésta, al momento de la construcción.

Hoy en día, aparatos electrónicos y dispositivos de toda índole permiten gobernar el entorno relacionado con variables como la temperatura del ambiente, la del agua y el volumen de la música de forma independiente. Alertas audibles de cocción de alimentos, apertura de ventanas, posición de las persianas, programas favoritos de televisión y alarmas perimetrales, son sistemas automatizados puestos al servicio del diario vivir. Sin embargo, la coordinación de tales acciones y la consolidación de la información que todos estos dispositivos generan no es tan común. El costo de su instalación, la complejidad de su integración, incluso, el desconocimiento de que estas funciones pueden ser agrupadas en un sólo sistema centralizado, son las razones por las que la domótica, a la fecha, no ha tomado el rol que le corresponde en la vida cotidiana del hombre moderno.

A pesar de que los indicadores económicos señalan que existe un grupo de consumidores que invierten en domótica, este es considerablemente bajo y no ejerce la presión suficiente en la demanda para que los precios bajen en favor de otros grupos económicos de menor poder adquisitivo.

Respecto de este último grupo, se observa que representan el grueso del mercado de los hogares y que es asiduo consumidor de equipos de seguridad de costo medio bajo, menos de 400US, pudiendo reafirmar que a pesar de conocer las

ventajas de la domótica, no la ha integrado a su cotidianidad, principalmente, por su elevado costo. El aumento de la comercialización de alarmas domiciliarias es fruto de la creciente sensación de inseguridad que sufre la clase media del país y que ha creado un fuerte nicho de mercado: el mercado de la seguridad puertas adentro.

Este fenómeno, podemos considerarlo como una tendencia si tenemos en cuenta la Encuesta Nacional Urbana de Paz Ciudadana, efectuada en Junio del 2008. En ella se establece que desde el 2007, en Santiago, el tipo de delitos que afectan a domicilios ha aumentado su ocurrencia de uno por cada quince días, hasta la actual de dos por día. Hecho que, al ser publicado por los medios de comunicación masivos, difunde el temor nacional de ser asaltado.

De lo anterior, y dado que uno de los fuertes de la domótica es la gestión de información centralizada de sensores distribuidos, es conveniente desarrollar un sistema domótico innovador que tenga un precio similar a un kit comercial de alarma de seguridad por zona. Así, será posible llamar la atención del mercado objetivo y lograr que la inversión que se hace, en seguridad convencional, se enfoque hacia una tecnología domótica moderna que permita futuras expansiones en su funcionalidad.

En consecuencia, se propone desarrollar un prototipo mejorado de un sistema de seguridad domótico denominado "Indoor" orientado al control de la actividad dentro de un recinto cerrado.

Esta propuesta es innovadora, a diferencia de los sistemas convencionales de seguridad indoor, exhibe la capacidad de detectar la presencia de un individuo u objeto y reaccionar según sean las credenciales de seguridad asignada a esa persona en específico. Por ejemplo, al implementarlo en una casa o edificio, la alarma no debe actuar, en un área específica, cuando un usuario autorizado esté presente en ella evitando tener que apagar todo el sistema de seguridad durante largos períodos de tiempo.

Además, se especifica que el sistema, en cada uno de los módulos, incluya contactos secos adicionales controlados por software, para que, a través de ellos, se pueda interactuar sobre la operación de los dispositivos no inteligentes que existan en un área determinada.

CAPÍTULO 1: DEFINICIÓN DEL PROBLEMA Y PLANTEAMIENTO DE UNA SOLUCIÓN

1.1 DEFINICIÓN DEL PROBLEMA

Una de las maneras de eliminar eventuales problemas de instalación y operación de cualquier sistema eléctrico-electrónico pasa por reducir o eliminar el cableado. Por ello, una base de domótica y sus módulos de interacción deben ser fáciles de instalar y de aprender a usar para que puedan ser abordados por el público masivo con familiaridad.

El dispositivo de identificación de presencia innovador que se proyecta, permite que el jefe de seguridad, no sólo sepa que hay alguien presente, sino que ahora puede individualizarlo y reaccionar frente a los permisos previamente definidos para el usuario.

Debido a la importancia de las funciones que esta solución entrega, se puede aspirar a un mercado más exclusivo dispuesto a pagar más, sin embargo, esta óptica no debe ser considerada al momento de la evaluación económica del proyecto pues es incompatible, a corto plazo, con el alcance hacia el mercado masivo al que el presente proyecto postula.

En función de lo anterior, tomando en cuenta específicamente el nicho de mercado que representa la seguridad bajo techo, es posible proyectar un sistema que opere sobre algún otro sistema de seguridad cableado disponible o, en su defecto, desarrollar uno completamente nuevo en que sea autónomo, inalámbrico y de fácil instalación.

1.2 SOLUCIÓN PROPUESTA

Puesto que el entorno, en que se decide implementar este proyecto es exclusivamente bajo techo, particularmente en pasillos de industrias, bodegas, hospitales, domicilios, etcétera y cuyo objetivo relevante es determinar la posición, e identidad, del individuo que se desplaza en esos lugares, se decide diseñar y construir unidades remotas que contengan el sensor correspondiente y su módulo de comunicación.

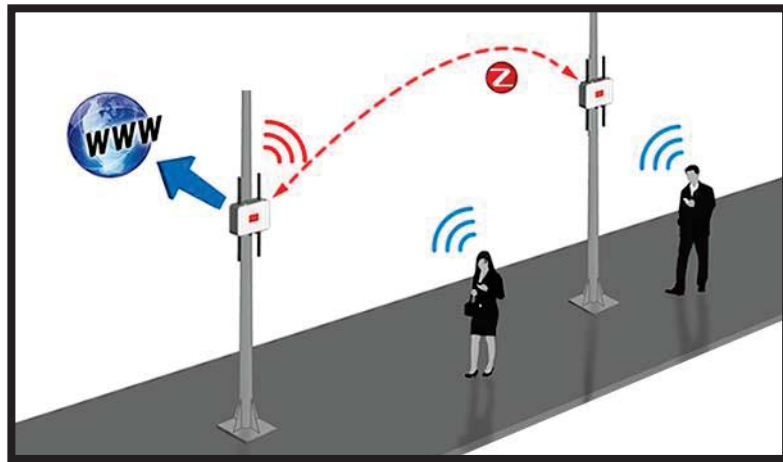


Figura 1 - 1 Ejemplo típico de implementación

Como se muestra en la Figura 1-1, en una serie de unidades sensoras puestas en cascada, siempre ocurrirá que el individuo que circula entre ellos, estará más cerca de un módulo estructural que de otro. En base a esas diferencias finalmente se puede determinar la zona donde éste está ubicado. Para conseguirlo es necesario conocer la magnitud de la señal irradiada por el transmisor del individuo. Esta es recibida, y medida para su análisis, por las antenas de los receptores contenidos en cada uno de los módulos de la cascada.

Basado en el criterio expuesto se estima, que en la primera etapa del proyecto, lo principal es determinar y ejecutar la comunicación del módulo flotante, módulo personal, con el módulo estructural, módulo sensor. La información obtenida de la red, permisos de acceso y configuración de los módulos deben ser administradas por un servidor, hardware y software, del menor costo posible.

De las tecnologías disponibles para la comunicación entre los módulos, se selecciona la plataforma XBee dado que, por su adecuado perfil de radio comunicación, se ajusta a las condiciones técnicas y económicas del fin que se persigue. Para el software del servidor, se opta por software de uso libre.

1.3 OBJETIVOS DEL PROYECTO

1.3.1. Objetivo General

- Diseñar un sistema de seguridad indoor que controle actuadores en función de parámetros predefinidos ante la detección de presencia dentro de una zona de seguridad en particular.

1.3.2. Objetivos Específicos

- Definir la tecnología que cumple mejor con los requisitos planteados.
- Familiarizarse con los protocolos y el funcionamiento general de la tecnología escogida.
- Construir un prototipo de la plataforma en hardware.
- Construir un prototipo de la plataforma del servidor.
- Evaluación económica bajo la perspectiva del mercado nacional.

1.3.3. Utilidades y fortalezas propuestas en el proyecto

- Localización de usuarios u objetos dentro de un área controlada
- Activación inteligente de actuadores controlados desde una central, en función de los privilegios del módulo que porta el usuario u objeto.
- En modo siempre encendido funciona como alarma común.
- Fácil instalación, administración y gestión de recursos relativos al sistema.
- Factibilidad de control y operación del sistema integrado a redes LAN y WAN.

1.4 DESCRIPCIÓN DEL SISTEMA A REALIZAR Y SU FUNCIONAMIENTO

El sistema en su modelo general en bloques aparece en el siguiente diagrama:

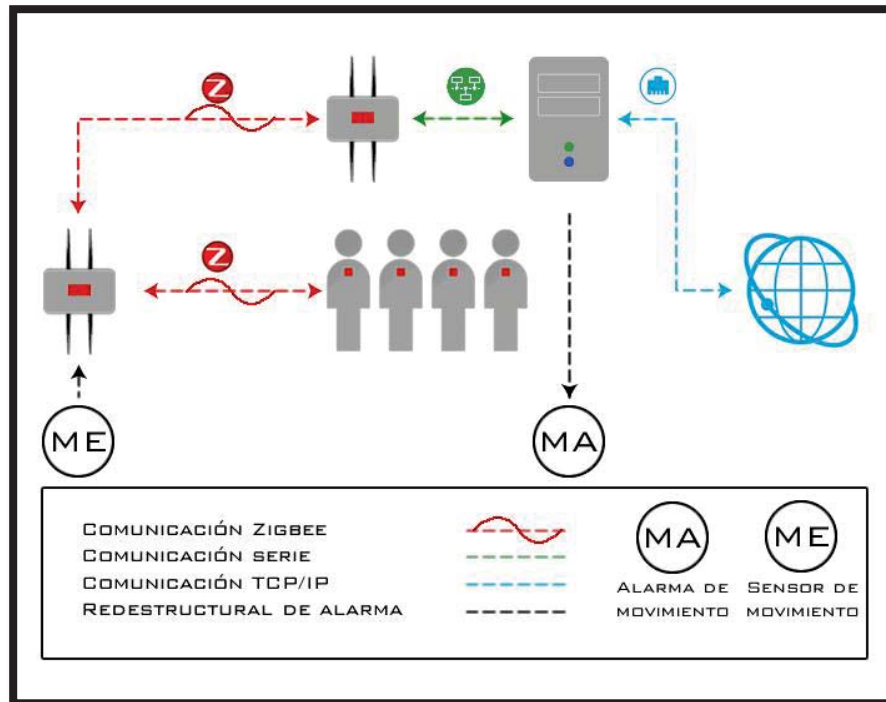


Figura 1 - 2. Diagrama general del sistema

Los módulos de comunicación Zigbee, operan de dos formas: función reducida (RFD) y de función completa (FFD), las que se distinguen por su consumo y capacidades. En cuanto a las capacidades, y a modo de ejemplo, en una misma red donde hay varios módulos de comunicaciones y; teniendo una determinada información que debe ser transportada desde una MAC de origen a una MAC de destino, en el caso de los RFD, el enlace ocurre solo entre dos módulos por lo que el transporte de la información queda entre esas dos unidades. En el caso de un módulo de forma FFD, toma la información, aunque no sea él el destinatario y establece comunicación con otros módulos, para pasarla de un módulo a otro hasta encontrar el módulo con la MAC de destino. En términos más coloquiales, el modo RFD puede asociarse a lo que es una carrera de 100 metros planos y la FFD a lo que es una posta, siendo la posta la información que se genera en la salida de la carrera y pasa de mano en mano (de módulo en módulo) hasta llegar a la meta. En cuanto al consumo, ambos tipos de módulos alternan períodos de bajo consumo

(tiempo de reposo) y de consumo nominal (tiempo de alerta), momento durante el cual el módulo consulta a los módulos vecinos si requieren que permanezca en operación, ciclo que en inglés es llamado *beaconing* y en español, balizamiento o balizaje.

En el caso de este proyecto, sus roles están definidos por la siguiente denominación:

- **Módulo Zigbee Flotante:** respondiendo a un modelo RFD, será integrado a la red Zigbee usando como mecanismo de seguridad una llave de 128 bits preinstalada y administrada por el servidor. Con un intervalo de alerta y reposo determinado por el coordinador de red, debe ser fijado para asegurar un máximo de rendimiento de las baterías, con intervalos de alerta no mayores a 0,7 segundos (valor máximo sugerido por el fabricante). Este proyecto centra el análisis de la autonomía (Cap 4.1.2) en fijar un parámetro de tiempo de reposo adecuado para alcanzar los 75 días de servicio.



Figura 1 - 3. Representación del módulo flotante

- **Módulo Zigbee Estructural:** respondiendo al modelo RFD, su alcance define físicamente una zona y es el dispositivo que recibe la información de presencia de uno o más elementos controlados. Además, Incorpora un relé normalmente abierto (NO) con capacidad de 10[A] de libre disposición.

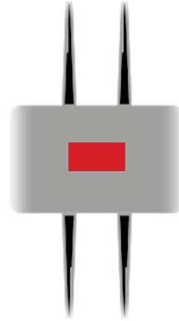


Figura 1 - 4. Representación del módulo estructural

- **Módulo Zigbee Central:** bajo un modelo RFD, es asignado como coordinador de red y datalogger de la red completa, alimentando a un servidor de forma continua. Es el módulo más complejo de todos. Puede incluirse en él una EEPROM para el almacenamiento de la información de autorización temporal, en caso de cortes de energía, mejorando así a la autonomía del sistema.

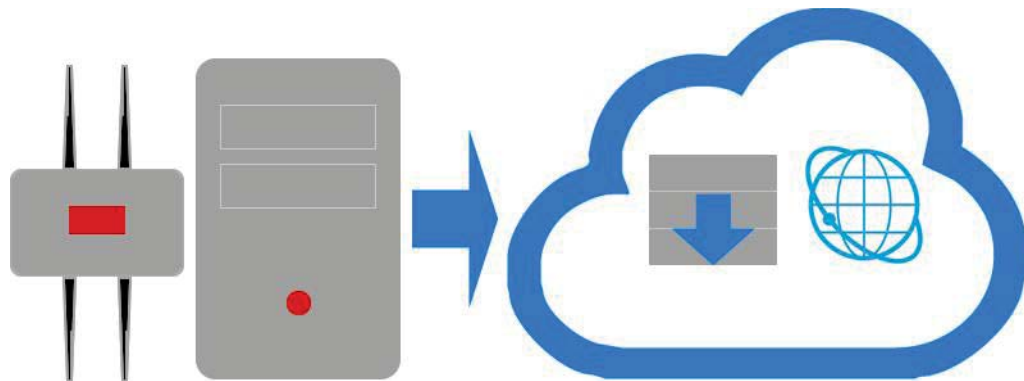


Figura 1 - 5. Representación del módulo central, servidor y comunicaciones

- **Servidor:** bajo la plataforma LAMP, un servidor de bajo nivel (64 MB RAM, 300Mhz) puede gestionar la información para ser consultada vía LAN a través de un navegador web compatible con W3Clo que lo convierte en un sistema multiplataforma. De esto se profundiza en el capítulo 4.2.

CAPÍTULO 2: ELECCIÓN DE LA TECNOLOGÍA

2.1 POR QUÉ PIC

En un módulo alimentado por baterías es importante observar los elementos que afectan al consumo de energía en cada componente. En el caso particular de los microcontroladores el consumo es, principalmente, función de factores tales como la carga en los pines de entrada y salida, la tasa de conmutación, el tipo de frecuencia, el patrón de ejecución del código interno y, por supuesto, la temperatura de operación. No obstante lo anterior, el fabricante entrega valores promedio de consumo en condiciones de trabajo estándar, 5VDC y oscilador reloj de 32khz-16Mhz, en tres modos de operación distintos de funcionamiento que son los llamados estados “activo” e “Idle” o “ahorro de energía”.

Como lo que se busca es abaratar costos, se opta por evaluar los microcontroladores más baratos, utilizados y disponibles. Entre ellos, se encuentran dos: el PIC16F87XA de Microchip TR (5,93USD) y el ATMEGA328P-PU de ATMEL (13,80USD).

Tabla 2 - 1. Consumo por uC en base a frecuencias de hasta 5Mhz y VDC de 5V

Microcontrolador	Activo	Idle	Ahorro de energía
ATMEGA328P-PU	1,7-0,9mA	0,3mA	0,8uA
PIC16F87XA	1,6-0,6mA	0,4mA	-

La Tabla 2 - 2 muestra el comportamiento en consumo típico que exhiben los microcontroladores seleccionados en cada fase de funcionamiento. Dada la permanente gestión de datos que le corresponde al microcontrolador se debe usar el medio de mayor consumo que es el activo. Además, teniendo en cuenta lo que respecta a precio el modelo PIC16F877A es la opción más adecuada.

2.2 POR QUÉ XBEE

En los sistemas de domótica, además de la medición de una o varias variables, fundamental es la intercomunicación entre los módulos. Usar cables o radiofrecuencia es una decisión que afecta directamente al costo de la solución planteada. Mientras que el cable entrega, en gran medida, la garantía de que el enlace se lleva a cabo sin problemas, su instalación en un entorno donde es relevante el confort y calidad de las terminaciones, obliga a ser instalado durante la construcción, o por lo menos, al momento de hacer los tendidos básicos (teléfono, TV cable, intercomunicadores). Es así como, en un entorno en que no se consideró inicialmente este tipo de instalación, el enlace de datos RF cobra fuerza. Es entonces cuando surge el primer problema: el precio.

A mediados de los años noventa se crean las arquitecturas y protocolos de comunicación que son adecuados para el tipo de implementación que persigue este proyecto. Bluetooth y ZigBee, ambas en los 2.4 Ghz, consiguen un desempeño aceptable sin sobredimensionar potencia en la transmisión bajo un concepto que unifica la compatibilidad entre dispositivos, además de la seguridad y bajo consumo. Sin embargo, Bluetooth encuentra su fuerte en la comunicación entre dispositivos móviles, no pudiendo competir con la capacidad de autoreparación (autohealing) de los enlaces de comunicación propio de las redes ZigBee y el bajo consumo característico y necesario en los módulos distribuidos para domótica.

Tabla 2 - 3. Comparativa Bluetooth y ZigBee

Tecnología	Consumo	Velocidad Max.	Nodos	Autonomía (días)
Bluetooth	40m – 0.2mA	2Mb	8	30
ZigBee	30m – 3uA	256kb	+ 65.000	100+

El estándar tecnológico ZigBee (aprobado como 802.115.4 por la IEEE el año 2004), busca cubrir las necesidades industriales de conectividad para sistemas de monitoreo remoto de bajo tráfico de datos y de corto alcance, especialmente para aplicaciones indoor. Desde su salida al mercado, la gran acogida de la tecnología

dio interesantes frutos en soluciones industriales, móviles y a nivel hogareño. Sin embargo, y debido a las continuas actualizaciones de las estructuras de datos (2004 y 2006), cada fabricante desarrolló sistemas de interconexión en red propietaria, de tal manera que, al adquirir un dispositivo de determinada compañía, la compatibilidad con dispositivos ZigBee más antiguos, o de otra marca, no estaban garantizadas. ZigBee solucionó el problema de la conectividad, pero no de la compatibilidad.

En Noviembre del año 2007, ZibBee Alliance hace pública la ZigBee Cluster Library bajo el nombre comercial de ZigbeeHA (ZigBee Home Automation), que fija un denominador común para los ingenieros que quieran desarrollar soluciones para la automatización residencial, siendo ésta la modificación más importante introducida a la tecnología.

La empresa Maxstream toma este estándar y se enfoca en brindar a los desarrolladores una herramienta sencilla de utilizar y muy robusta, creando XBee.

Hoy XBee es un sistema propietario de Digi (antes Maxstream), el cual ofrece muchas de las cualidades de ZigBee, a menor precio, con la ventaja de estar más cerca del origen, y de los principios, de la tecnología, dotándolo de una muy amplia compatibilidad con los nuevos dispositivos que aparecen en el mercado.

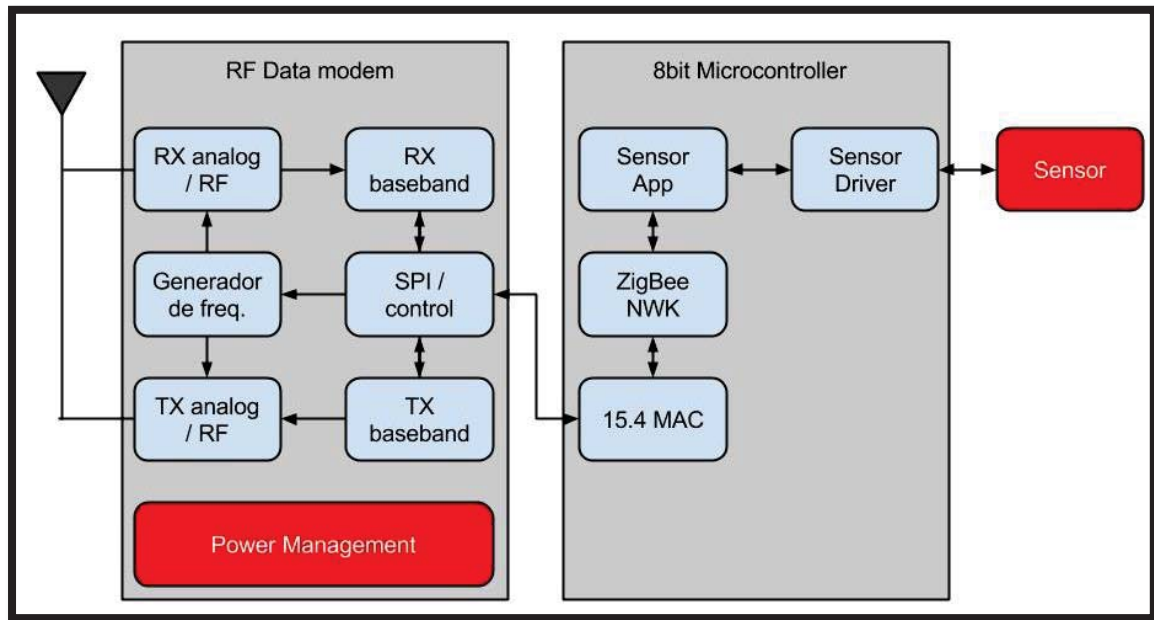


Figura 2 - 1. Implementación típica de XBee y microcontrolador

2.3 XBEE O XBEE PRO

Las diferencias principales entre XBee y VBee Pro están en su consumo y sensibilidad del receptor (1.25mW a 10mW y sensibilidad de -95dBm a -102dBm respectivamente) y, por ende, en la eficaz cobertura dentro de un radio determinado.

Difieren marginalmente en sus dimensiones, funcionalidad, configuración, distribución de pines y código de control, operando ambos a 2.4 Ghz sin necesidad de licencia, banda ISM (Industrial, Scientific, Medical). Ambos módulos disponen de una antena impresa en la tarjeta PCB o un conector UFL para antenas externas.

El XBee irradia hasta 1mW de potencia, mientras el XBee Pro lo hace hasta los 60mW, además su receptor exhibe la capacidad de recibir señales más débiles que el XBee. Esto redundaría en que el XBee Pro puede establecer comunicaciones de mayor alcance.

En el módulo XBee es más simple la modulación y demodulación de las señales que recogemos desde el sensor y hacia el actuador, relacionados por medio

del microcontrolador que entrega digitalizada la señal al módulo XBee, como muestra la figura 2-1.

2.3.1. Calidad de enlace

Una herramienta para la decisión de qué módulo utilizar es la tabla 2-3 que distribuye la empresa Digi, relaciona el alcance de las antenas disponibles en la tecnología de los modem según el tipo de espacio en que se van a usar, ya sea en espacios cerrados o abiertos (indoor y outdoor, respectivamente).

Con el objeto de caracterizar el radio de cobertura en espacios cerrados, el fabricante comparó el comportamiento de los módulos en bodegas y oficinas y, para la comparación en espacios abiertos, usó un espacio próximo a sus instalaciones. Tomaron medidas que ordenaron geográficamente para obtener un patrón de radiación considerando la máxima distancia a la que se envía y recibe un paquete de datos de forma íntegra. Este paquete de datos consta de un número incremental transmitido desde el emisor a un computador portátil programado para cuantificar el porcentaje de las recepciones correctas con respecto del total de los paquetes enviados. Las pruebas fueron hechas en base a que el emisor y el receptor correspondían al mismo tipo de dispositivo y antena, ambos montados en interfaz USB (para la interacción con el computador) sin esperar confirmación, pues el paquete se enviaba solo una vez.

Los resultados del experimento se muestran en la Tabla 2 - 4.

Tabla 2 - 5. Pruebas de enlace XBee v/s XBeePro

	Antena	Outdoor	Indoor	Indoor
Módulo	Type	Con línea Vista	Oficinas	Bodega
XBee	Chip	143 [m]	24 [m]	-
	Alambre	258 [m]	24 [m]	26 [m]
XBee-PRO	Chip	515 [m]	43 [m]	-
	Alambre	1335 [m]	43 [m]	108 [m]

Un dipolo de λ medio también fue usado para el experimento, sin embargo, el monopolo y el dipolo tuvieron resultados muy parecidos especialmente en condiciones bajo techo.

Algo interesante respecto a la diferencia de rendimiento es el referente a la orientación del módulo.

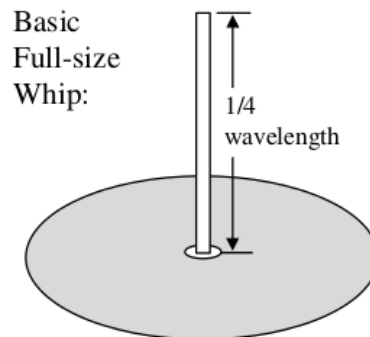


Figura 2 - 2. Antena de 1/4 de largo de onda

De acuerdo a la información del fabricante, el diagrama de radiación para la antena monopolo es parecido a la del dipolo, sin embargo, el rendimiento de un módulo usando la antena de alambre es relativamente insensible a su orientación con respecto al plano al que es perpendicular, como muestra la figura 2-5. Por otro lado, el rendimiento de radiación de la antena de chip (antena integrada) no es el mismo que el de la antena de alambre. En la figura 2-4 y 2-5 se aprecia que ciertas orientaciones alcanzan mejor distancia que otras. Se ajustó la orientación hasta

determinar la mejor señal y, finalmente, fueron éstas las que se usaron para recoger las medidas que se despliegan en los gráficos posteriores. Se recalca que los resultados pueden haber sido afectados por interferencias con señales provenientes de antenas vecinas.

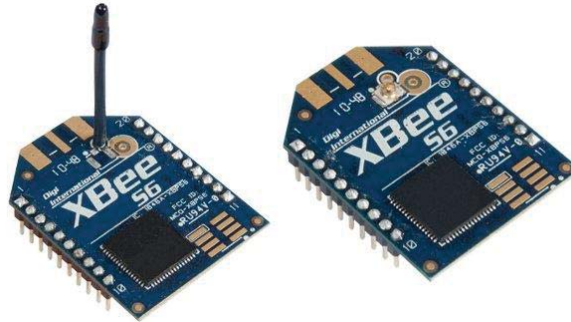


Figura 2 - 3. Imagen de modelos

2.3.2 Antena

Se adelantan conclusiones al ver la tabla 2-3 pues se puede fácilmente decidir por la antena de alambre; sin embargo, hay que considerar el uso del espacio que demanda ese tipo de implementación.

En el caso particular del módulo flotante, sería muy inconveniente integrar una antena de esas características a un dispositivo diseñado para ser portable y, en cierto modo, sometido a los típicos maltratos que sufren los artículos de bolsillo.

A pesar de que estas pruebas no muestran los enlaces con antenas combinadas (antenas monopolo e integradas en transmisor y receptor), la relación entre los módulos estructurales y flotantes mostrarían características similares a las pruebas realizadas únicamente con antenas integradas.

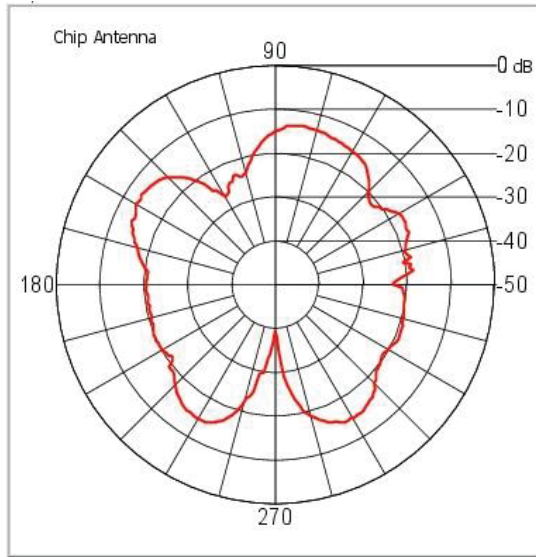


Figura 2 - 4. Pattern horizontal de XBeePro usando antena integrada, normalizada a la mayor ganancia registrada del dipolo (0dB)

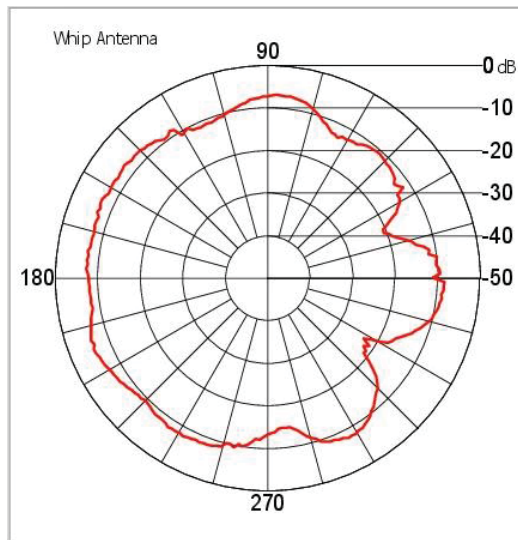


Figura 2 - 5. Pattern horizontal de XBeePro usando antena de alambre, normalizada a la mayor ganancia registrada por el dipolo (0dB)

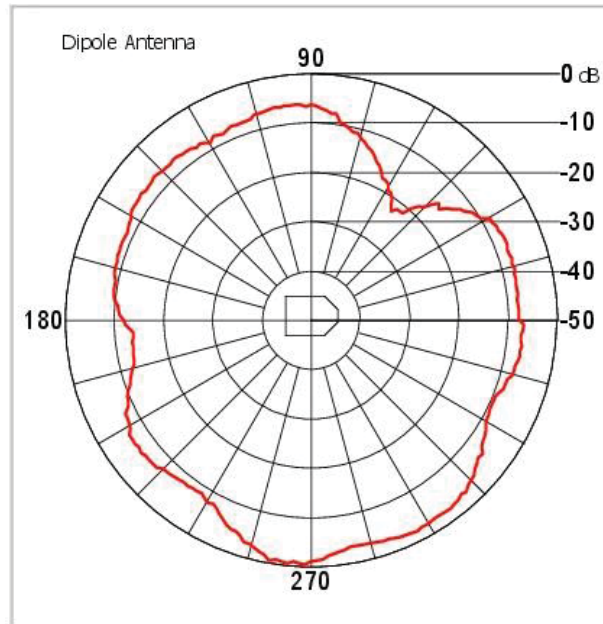


Figura 2 - 6. Pattern horizontal de XBeePro usando antena dipolo normalizada a la mayor ganancia registrada por el dipolo (0dB)

Para el caso de la antena integrada (chip antenna), el proveedor no ofrece un diagrama de cobertura en orientación vertical. Se replicó la metodología de prueba que usó el fabricante, esta vez, normalizado al mejor enlace conseguido por el propio dispositivo en función de su orientación respecto al receptor y se anotaron los resultados para la gráfica 2-7. En ella, 0 equivale al 100% de los paquetes recibidos, -10 equivale al 90% de los paquetes recibidos, y así consecutivamente hasta llegar hasta -50, que es cuando el software de prueba declara demasiadas fallas mostrando el mensaje “enlace interrumpido”.

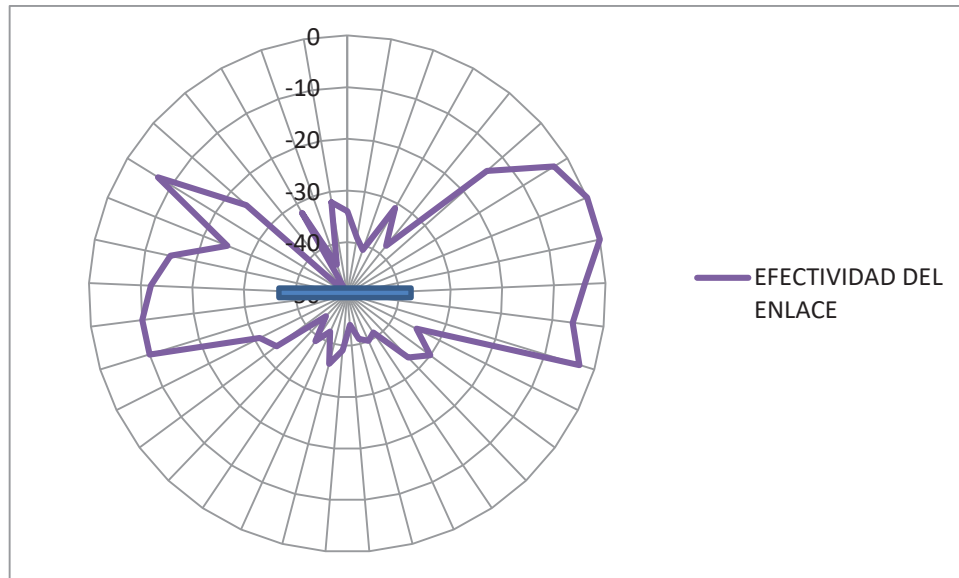


Figura 2 - 7. Pattern vertical de XBee usando antena integrada (chip antenna) normalizada al mejor rendimiento obtenido

CAPÍTULO 3: DESARROLLO DE HARWARE Y SOFTWARE DEL SISTEMA

3.1 INTERCONEXIÓN

La estructura general del sistema que se ha determinado está constituida por módulos que cumplen las funciones indicadas en la Tabla 3-1 siguiente.

Tabla 3 - 1.Módulos y sus componentes

Función	Partes
Módulo Flotante	Módulo XBee, Microcontrolador
Módulo Estructural	Módulo Xbee, Microcontrolador, Actuadores.
Módulo Central	Módulo Xbee, Microcontrolador, Max232, Actuador.
Servidor	LINUX, APACHE, PHP, MYSQL.

Los procesos más importantes del desarrollo modular son la comunicación entre el módulo XBee, y su respectivo microcontrolador, la participación del C.I. Max232, en el caso del Módulo Central, y el actuador. Estos se describen en el presente capítulo.

3.1.1. Conexión entre XBee y PIC

Para la comunicación serial del PIC con el módulo Xbee central, se utilizan los terminales de recepción y transmisión de éste, el pin RC6 para la transmisión y el RC7 para la recepción. Además, se deben interconectar las tierras comunes de ambos dispositivos.

En las placas de desarrollo, la alimentación de los módulos no es un problema. A pesar de que aparece una diferencia entre la alimentación del 16F877A (5v) y la recomendada para el XBee (3,3V). Esta no es una dificultad debido a que, además, el fabricante del XBee indica que éste tolera 5V de alimentación. En consecuencia, tal como representa la figura 3-1, se opta por una conexión directa entre las terminales I/O de estos componentes.

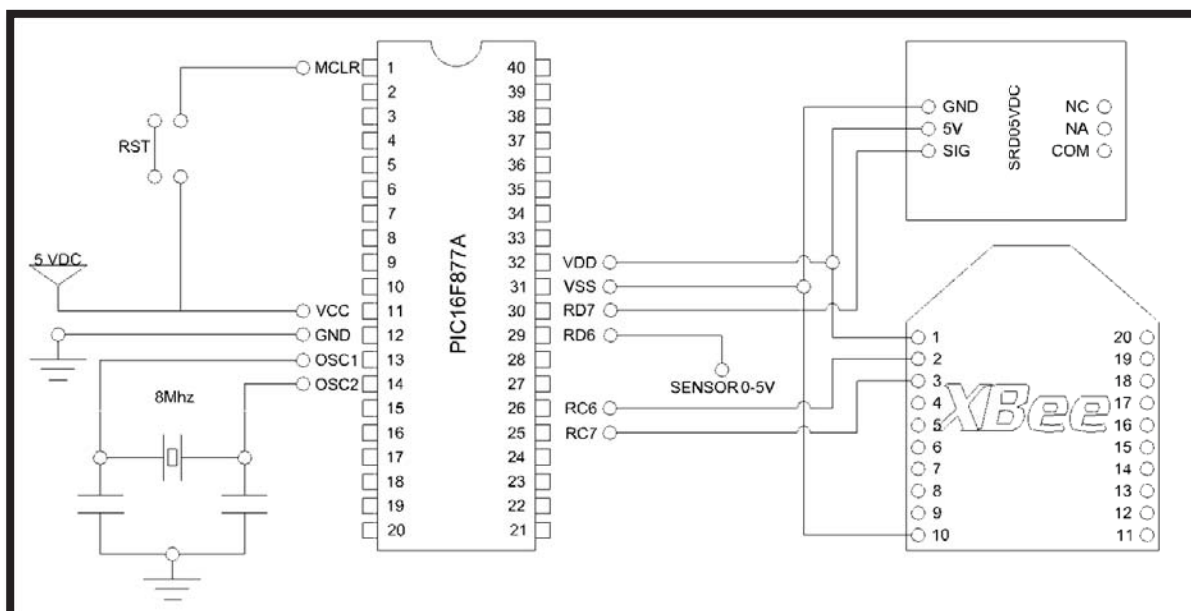


Figura 3 - 1. Conexión PIC y Xbee

Lo que se persigue es, básicamente, conseguir que un relé se accione cuando ocurra un determinado estímulo entregado por el PIC. Esto debe, a su vez, suceder cuando la información proveniente de un sensor lo indique.

El diagrama de comunicaciones entre los módulos servidor-estructural y sus respectivos microcontroladores se muestra en la Figura 3-2.

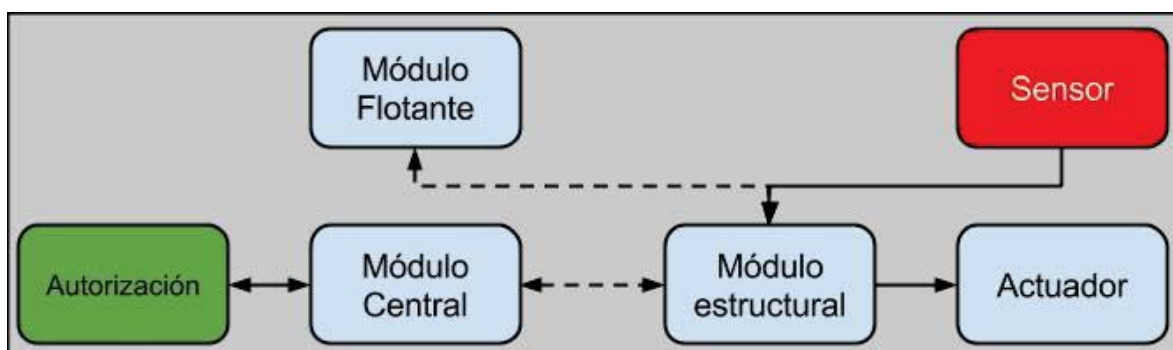


Figura 3 - 2. Diagrama de comunicaciones

La operación del sistema es la siguiente; el Sensor envía una señal al Módulo estructural. Este transmite una solicitud de autorización de presencia al Servidor (Autorización en Figura 3-2) a través del Módulo Central.

La autorización, o no-autorización, viajará de vuelta por el mismo canal para su procesamiento terminando en el comando de un relé vía el PIN B1, que se ha configurado como salida del microcontrolador.

3.1.2 Conexión típica del actuador

El actuador, figura 3.3, consiste en un relé SPDT cuyos contactos permiten cerrar o abrir un circuito. El relé es alimentado por un transistor que lo acciona toda vez que la señal del microcontrolador produce la corriente de base suficiente. Un diodo volante de protección se incluye para evitar sobretensiones, no deseadas, sobre el VCE del transistor, cuando éste deja de alimentar la bobina.

La corriente en la base del transistor, causada por la señal de salida desde el PIC, es suficiente para que el transistor entregue los 71mA necesarios para el accionamiento del relé. Por la parte de “fuerza” del circuito se tiene el conmutador SPDT que tiene un borne denominado “común” (COM) y los terminales “normalmente abierto” y “normalmente cerrado” (NA y NC, respectivamente), Figura 3-3. Para esta etapa del sistema se usa un transistor 2N3906, el diodo volante 1N4001 y el relé SRD05VDC, manufacturado por empresa SOGLE RELAY.

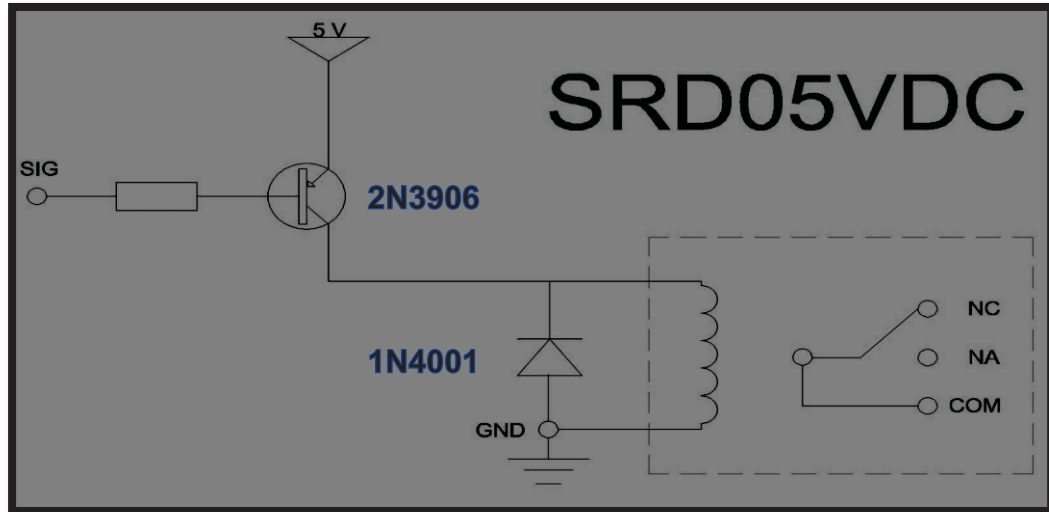


Figura 3 - 3. Conexión de un relé activado por una señal digital de salida del PIC

3.2 SOFTWARE

3.2.1 Lenguaje C

En relación a la imagen 3-1, el código grabado en el microcontrolador PIC (Microchip Inc., 2001) es desarrollado usando lenguaje C. Su objetivo es integrar el actuador y el módulo de comunicaciones XBee.

Dos de las funciones más utilizadas para el control de los módulos se detallan a continuación. El resto de las funciones y su código fuente se encuentran en Apéndice.

- **XBeeRF_str (APIID, IDF, ACK, ADDR64, "STRING", NUM, numbytes)**

Esta función construye y envía al puerto serie el paquete API de forma automática, alimentándola con los argumentos que se detallan:

- **APIID:** (1:64bits, 2:16bits) para comunicación en 64bits, este método considera como dirección de destino del paquete, a la MAC del módulo

XBee. Este número debe estar compuesto por 8Bytes, siendo algunas veces necesario completar la cifra con ceros a la izquierda.

- **IDF:** (0:Disable, 2 para 16bits, 1byte:default) Identificador de cuadro, se usa para etiquetar los comandos que se envían. Esta función cobra utilidad al momento de enviar varios comandos al mismo tiempo y hacer posible la identificación de las respuestas a cada una de las instrucciones.
- **ACK:** (0: deshabilitado, 4:broadcast, 1byte:habilitado). Su habilitación entrega una respuesta con el aviso de ocurrencia de un error.
- **ADDR64:** Dirección de 8bytes que componente la MAC del módulo de destino que debe ser ingresada byte a byte al arreglo ADDR64.
- **STRING:** Palabra que será enviada al módulo remoto en modo ASCII byte a byte. Esta opción es muy práctica para enviar código AT sin necesidad de hacer la traducción manual.
- **NUM:** Palabra compuesta por números que generalmente componen el argumento de un código AT, con el fin de fijar un parámetro. Al igual que ADDR64, este argumento debe ser previamente ingresado a un arreglo.
- **Numbytes:** El número de bytes válidos que componen el argumento anterior.

XBeeRF_str (APIID, IDF, ACK, ADDR64, "STRING", NUM, numbytes)

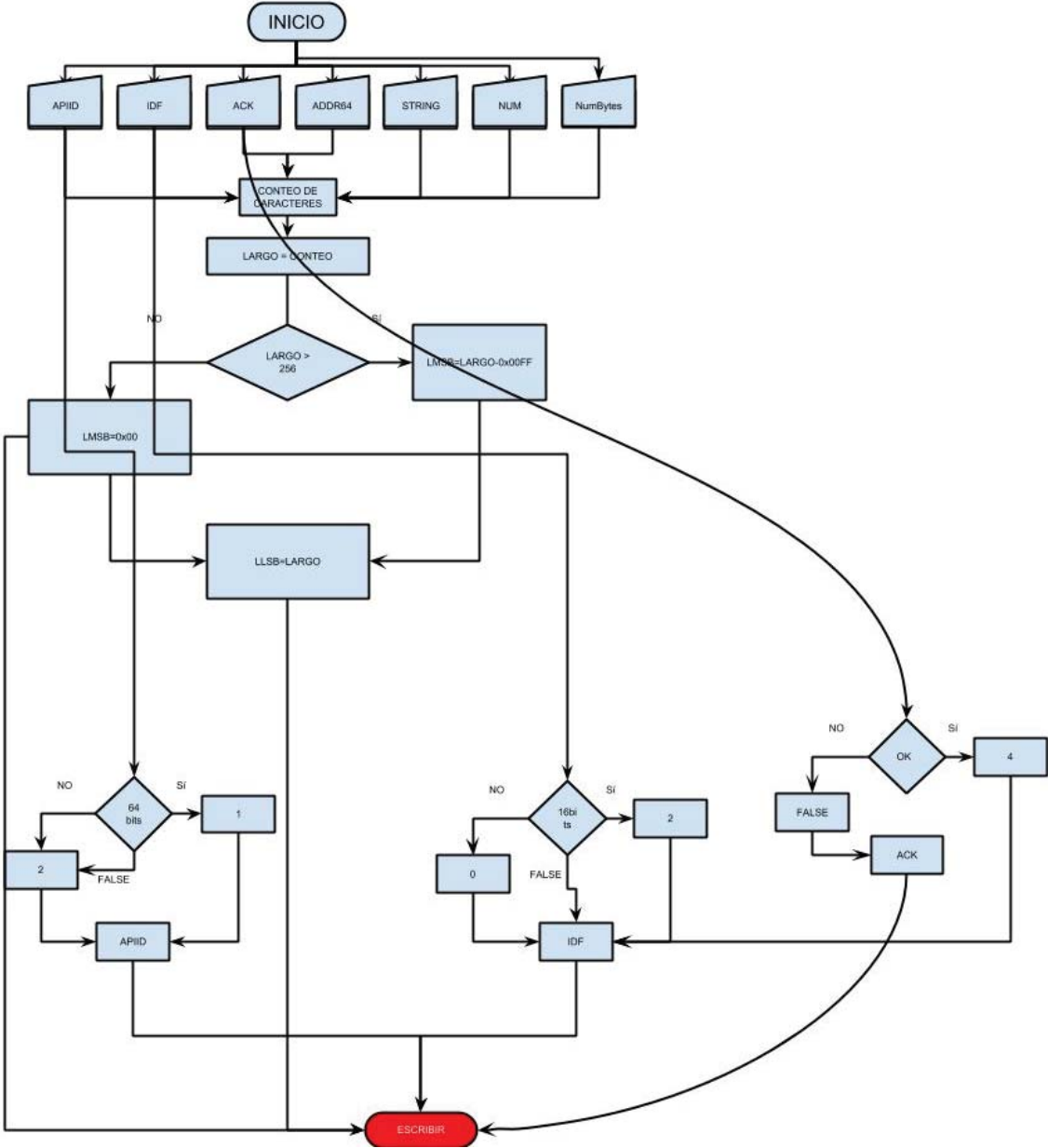


Figura 3 - 4.Función XBeeRF_str

- **XBee_str(APIID, ACK, "STRING", NUM, numbytes)**

Permite enviar un paquete en formato API a un módulo directamente conectado al PIC.

- **APIID:** (1:AT COMMAND, 2:AT QUEUE o 'cola de espera'). Define si se desea que el paquete sea procesado inmediatamente o bien, dejado en la cola para una eventual ejecución.
- **ACK:** (0: deshabilitado, 1: habilitado). Habilita, o noó, la recepción de una confirmación desde el módulo local. Si el APIID está fijado en 2, entonces esta variable se convierte en un identificador de cuadro, a la manera del IDF en 64 bits.
- **STRING:** Define el string que será enviado byte a byte con su correspondiente código ASCII.
- **NUM:** Arreglo numérico de 1 byte, que normalmente compone el argumento de un comando AT.
- **Numbytes:** Debe contener el número de bytes relevantes en el arreglo NUM.

El siguiente diagrama (figura 3-5) de flujo describe el algoritmo. Un detalle del código de las funciones explicadas en este capítulo y otras, se encuentra en el “Apéndice 1 Código en C”.

XBee_str(APIID, ACK, "STRING", NUM, numbytes)

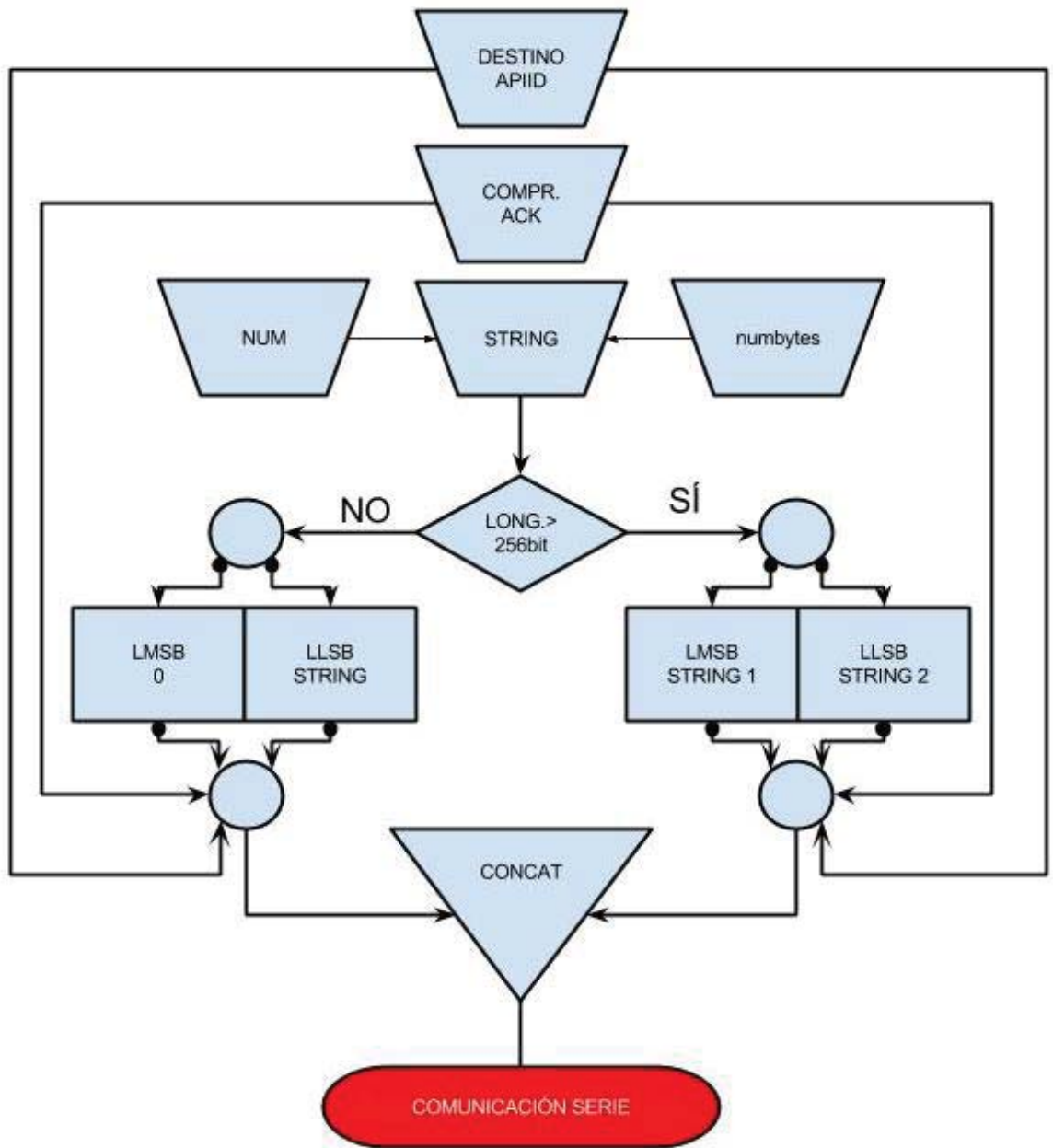


Figura 3 - 5. Función XBee_str

3.2.2 Lenguaje PHP (PHP INC.)

Este lenguaje, PHP, es un código de interpretación. Esto quiere decir que es el intermediario, o dicho de otra forma, el traductor entre el código HTML y las bases de datos MySQL (MySQL Inc). Dos de ellas están publicadas bajo la norma GNU/GPL y las tres son de libre disposición.

A pesar de tener un rol bastante relevante esta opción en el proyecto, no se describen mayores detalles respecto a ese código; pues se considera un paso fuera de la temática que envuelve al proyecto. Al igual que con el lenguaje de microcontroladores, fue necesario generar funciones que permitieran trabajar la transmisión y recepción de paquetes, de contenido de carga variable, de forma sencilla

La función destacada dentro de las construidas es:

- **ApiOut(\$API)**

Toma los datos recibidos y extrae la información más relevante en ellos.

\$API: Contiene una palabra de varios bytes que obedece a la estructura API que se usa en la comunicación entre módulos XBee. El uso de esta función permite separar los aspectos más relevantes de la información recibida, tales como: largo de la palabra, potencia de la señal entre el módulo terminal y el primer receptor, la MAC y, por supuesto, la información contenida.

Ejemplo:

La palabra "i" enviada desde el módulo 0013A20040063337

```
$datos = apiOut(7E000C80000013A200400633372E006983);
```

Ahora, \$datos es un arreglo que contiene:

- \$datos [addr] contiene 0013A20040063337
- \$datos [rssi] contiene -46dBm (0x2E)
- \$datos [content] contiene i.

- ApiOut(\$API)

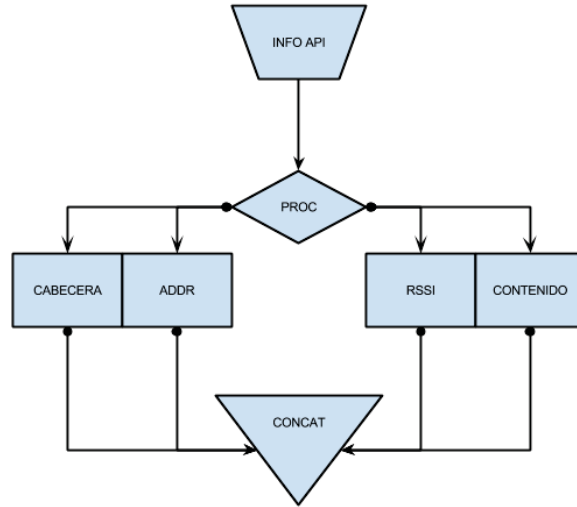


Figura 3 - 6. apiOut(\$API)

- **txApi(\$datos, \$addr, \$ack).**

Conformación de una palabra en modo API y envío byte a byte por el puerto serie.

- \$datos: está compuesta por la palabra escrita en hexadecimal que se quiere enviar.
- \$addr: la MAC del módulo de destino.
- \$ack: (0: deshabilitada, 1: habilitada). Controla la confirmación de recepción.

Ejemplo:

Se quiere enviar "ID" al módulo 0013 A20040088ACD.

```
txApi("4944", "0013 A20040088ACD", 1);
```

Se habrá enviado 7E000D00010013A20040088ACD0149441C al módulo local, pudiendo éste interpretar el redireccionamiento que se le solicita.

- **txApi(\$datos, \$addr, \$ack).**

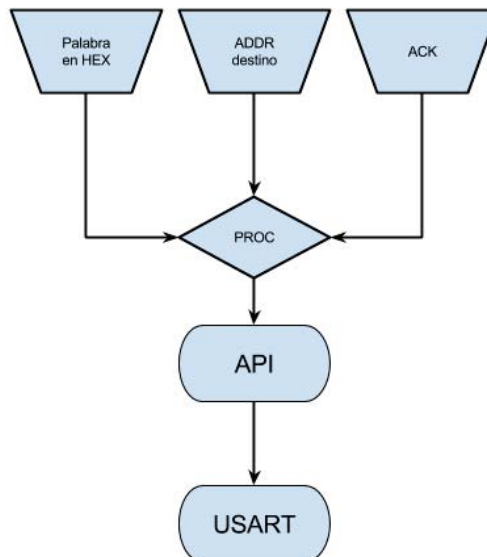


Figura 3 - 7. txApi(\$datos, \$addr, \$ack)

3.2.3 Instrucciones y gestión MySQL

Es uno de los más populares gestores de bases de datos que se pueden encontrar hoy en día. Es en extremo flexible y cuenta con una cantidad enorme de funciones que permiten gestionar la información de forma sencilla a un costo de proceso realmente bajo.

La estructura de base de datos que se usó para el presente proyecto es la que sigue:

- Autorizado, contiene los datos de los módulos que están autorizados.
 - Id: índice
 - addr: MAC del módulo autorizado (no es key sensitive)
 - nombre: Nombre del usuario del módulo
 - rango: Información personal respecto al usuario
 - zona: zonas sobre las cuales el usuario tiene privilegio de circulación
 - fecha: fecha en la cual fue ingresado.
- Registro, contiene la bitácora de actividad de todas las zonas con fechas detalladas en segundos.
 - Id: índice
 - largo: largo de la palabra recibida
 - addr: MAC del módulo
 - rssi: potencia de enlace
 - content: información contenida dentro del API

3.3 INTERCONEXIÓN GENERAL

En el diagrama de la figura 3-8 se aprecian los componentes, MAX 232, Pic y módulo XBee en la interconexión montada en el módulo central. El más complejo y completo de todos.

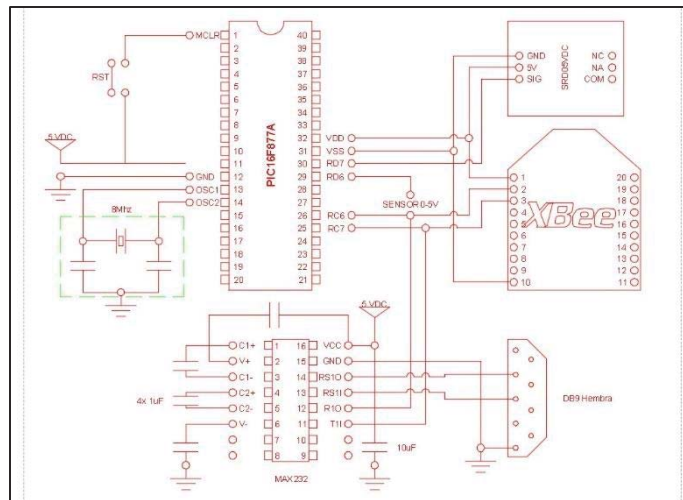


Figura 3 - 8. INTERCONEXIÓN GENERAL

CAPÍTULO 4. MONTAJE DEL SISTEMA

4.1 DETERMINACIÓN DE REQUISITOS DEL SISTEMA

Para implementar el sistema completo son necesarias algunas exigencias mínimas para el servidor, y ciertos márgenes para garantizar que las lecturas de nivel de señal, en las que se basa la detección de proximidad entre el módulo estructural y el módulo flotante, maximicen su confiabilidad. Además de la correcta interacción con cada módulo, el computador, que cumple el rol de servidor, no debe significar una fuerte inversión. Se comprobó que es factible utilizar las plataformas que emplean procesadores tipo INTEL PENTIUM 2 de 400Mhz con, a lo menos, 128MB de RAM. Estas especificaciones también permiten que se pueda trabajar con ordenadores de placa reducida, tales como Raspberry Pi o similares.

4.1.1. Distancia entre módulos

Empíricamente, la potencia de la señal recibida entre módulos separados por 0,5m es de -46dBm; llegándose a los -90dBm cuando la distancia es de 45m, con línea vista. La misma lectura (-90dBm) también se consigue a los 10m de separación, habiendo dos murallas de concreto de un espesor, de 30cm, cada una, obstruyendo en el paso de la comunicación.

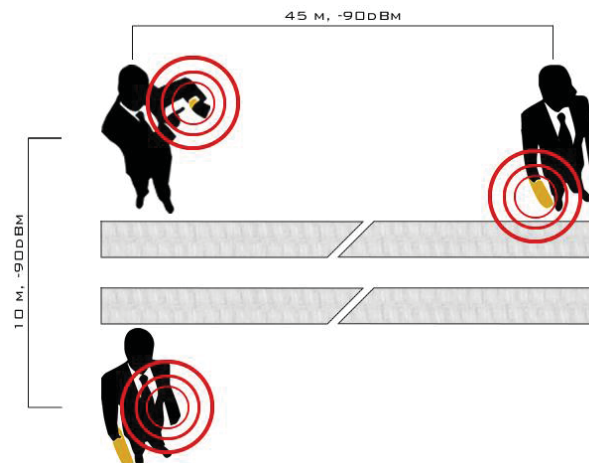


Figura 4 - 1. Esquema de prueba de potencia en el enlace

A diferencia de lo expuesto en la sección 2.3.1 del Capítulo 2, respecto a la calidad de enlace, se asume que los resultados experimentales son las especificaciones técnicas de la instalación del producto, es decir, que para conseguir un enlace seguro, por sobre los $-90[\text{dBm}]$, los módulos no deben estar separados, el uno del otro, por más de 45 [m] en cuanto exista línea vista y, por más de 10 [m] cuando existen obstáculos.

4.1.2. Autonomía

El sistema móvil remoto completo (XBee + uC 16F877A) es alimentado con 5 VDC. Para este voltaje de alimentación, se tienen las corrientes de consumo, aproximadas, que se indican en las tablas 2-1 y 2-2 en los estados de alerta y de reposo.

El soporte técnico de la Empresa Digi, por medio de sus foros públicos en línea, recomienda que el intervalo de tiempo para el estado de alerta no sea menor a 20[ms]. Bajo estas condiciones, usando la herramienta "BLC" (Battery Life Calculator), entregada gratuitamente por Digi, a los suscriptores de su sitio web, arroja una autonomía de 15 meses para baterías de 2000 [mAh]. Agregando a ese dato el consumo promedio del del microcontrolador PIC, que es 600 uA en modo activo, disminuye la autonomía respecto a lo indicado por Digi.

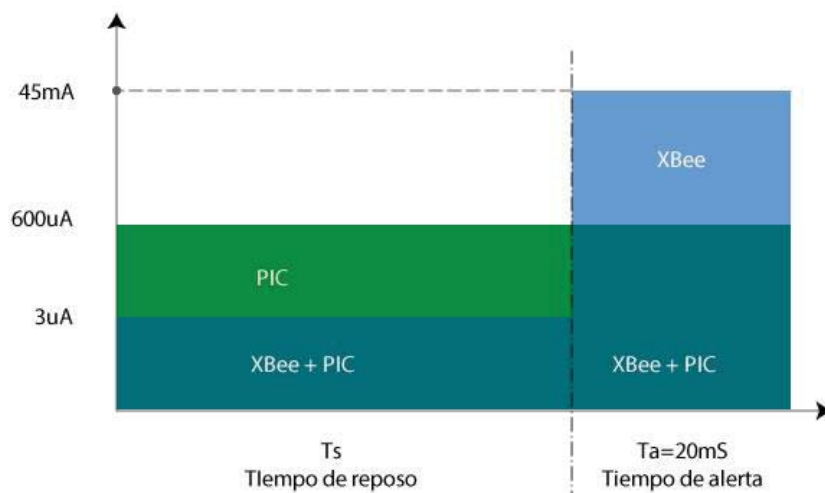


Figura 4 - 2. Diagrama de consumo promedio para un ciclo de reposo-transmisión y recepción

Dada la gráfica anterior, la ecuación del consumo de corriente promedio en función del tiempo de reposo del sistema es:

$$Q_{XBee} = \int_0^{ts} 3 * 10^{-3} dt + \int_{ts}^{ts+ta} 45 dt [mAh]$$

$$Q_{PIC} = \int_0^{ts+ta} 600 * 10^{-3} dt [mAh]$$

$$I_{promedio} = \frac{Q_{XBee} + Q_{PIC}}{ts + ta} [mA]$$

Como el intervalo de alerta es de 20ms, tenemos:

$$ta = \frac{20E - 3}{3600} = 5.56E - 6 [h]$$

$$I_{promedio} = \frac{603E - 3 * (ts + 420E - 6)}{ts + 5.56E - 6} [mA]$$

Si observamos estas ecuaciones, podemos tomar algunas decisiones para optimizar el rendimiento del sistema, sin perder funcionalidad, usando algunas directivas:

- Un menor tiempo de reposo aumenta la resolución del sistema. Al generar más datos el usuario puede contar con información más precisa.
- Un mayor tiempo de reposo aumenta la autonomía del sistema. Para fines comerciales del proyecto, una autonomía mayor a los 2,5 meses es apropiado.
- El sistema es alimentado por 1 pack de celdas de NIMH que almacenan 2200 mAh y entregan los 4,8V que se requieren. (Vonlin Industry CO, Modelo B4.8VNIMH)
- El rendimiento real de las baterías es estimado a 80%

Dado lo anterior, tenemos:

$$Autonomía = \frac{2200 * 80\%}{I_{promedio}} [h] = 2,5 * 24 * 30$$

$$t_s = 661E - 6 * 3600 = 2.38s$$

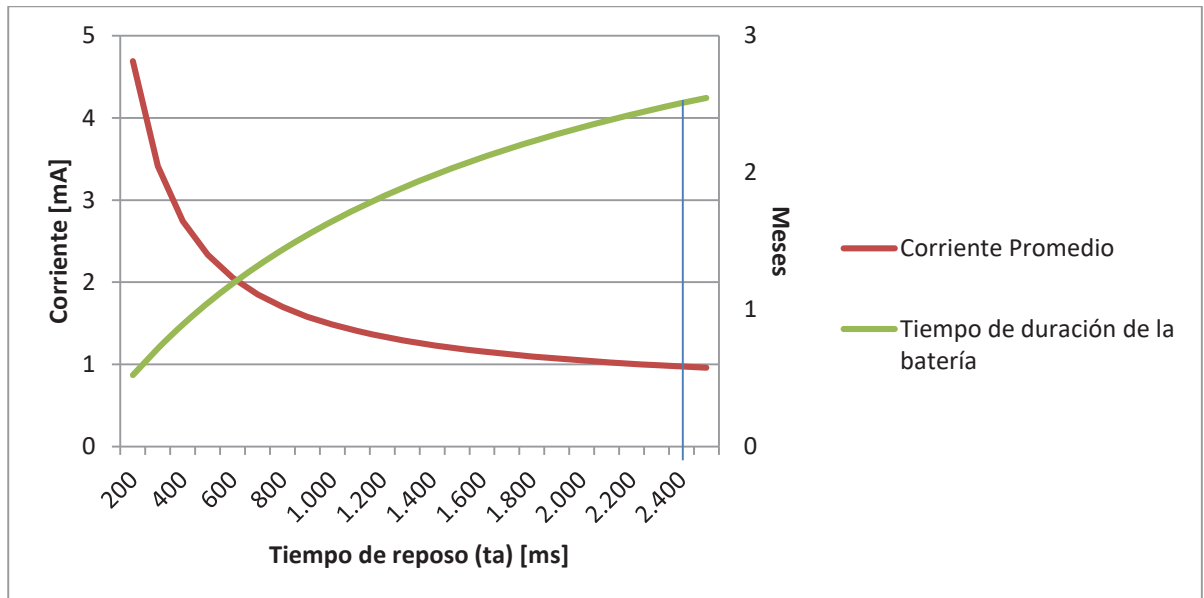


Figura 4 - 3. Gráfico de Autonomía del sistema

Se confirma en la Figura 4-3 que el valor de 2380ms cumple con los criterios definidos en un principio.

En términos del funcionamiento del módulo, este tiempo significa que cada dos segundos y medio, aproximadamente, el sistema sensorá los módulos que están presentes en la red. Para las características de este proyecto eso es suficiente.

4.1.3. Plataforma para el servidor

De las pruebas con el servidor usando chipset 440BX INTEL, se demuestra que es una tecnología del año 2002, muy estable. Además, se garantiza que existe el amplio espectro de computadores que son capaces de llevar a cabo la tarea de administración que se necesita.

Los requerimientos computacionales mínimos son:

- CPU 800Mhz
- 128 MB RAM
- 2GB en disco duro

- Tarjeta de Red 10b T

No existen costos adicionales asociados a softwares derivados de este sistema, pues todo está montado bajo software GPL.

4.2 SOFTWARE DE SERVIDOR

4.2.1 Sistema operativo

Linux es un sistema operativo de código libre que destaca por su enorme estabilidad comparada, incluso, con las versiones más populares de software para estos fines proveniente de Microsoft. Una “distribución” es un conjunto de paquetes de software que facilita la instalación, la configuración y el mantenimiento de un sistema GNU/Linux.

Existen numerosas distribuciones Linux (también conocidas como "distros", diminutivo de "distribuciones"), ensambladas por individuos, empresas y otros organismos.

Entre las distribuciones de GNU/Linux, destacan el proyecto Debian/GNU. Debian nace como una iniciativa no comercial de la Fundación de Software Libre (Fundación sin fines de lucro), aunque luego se independiza de ésta y va más allá del propio sistema GNU/Linux. Es la única de las distribuciones importantes que no tienen intereses comerciales ni empresariales. Son sus propios usuarios, quienes mantienen la distribución de modo comunitario, incluidas todas sus estructuras de decisión y funcionamiento. Su objetivo es recopilar, difundir y promover el uso del software libre. Reúne el mayor catálogo de software libre, todos ellos probados, mantenidos y documentados por algún desarrollador voluntario.

Una distribución contiene todo el software necesario para instalar en un ordenador personal; servidor, correo, ofimática, fax, navegación de red, seguridad, etc. Una lista de las distribuciones más conocidas para distintos usos se encuentra en <http://www.getgnulinux.org>.

4.2.2 Cómo se instala Linux

Linux es un sistema operativo fácil de instalar, tan sólo basta con descargar la imagen iso de la distribución usar como Debian, SuSE, Ubuntu o Yellow Dog (la mayoría son gratuitas), grabarla en un CD o DVD, o en un pendrive con capacidad superior a 4Gb o, en el caso de no contar con ninguna de esas posibilidades, instalarlo por medio de un servidor PXE.

Existen versiones Linux para máquinas de 32 bits, y también para 64 bits. Existen también distribuciones para plataformas Macintosh con sus procesadores PPC (hasta 200Mhz), G3, G4, G5 y los más actuales. Cabe destacar que el sistema OSX de APPLE está basado en una versión más elaborada de Unix y, por esta razón, es considerado, por algunos, como una distribución comercial de Linux.

Linux ha tenido éxito en desarrollar una plataforma de instalación muy sencilla tal que hoy es incluso más simple de instalar que Windows. Algunas distribuciones permiten entrar al escritorio Linux, sin hacer cambios en el disco duro, para dar al usuario la experiencia de uso antes de decidir la instalación definitiva en la unidad de almacenamiento interna. Este es el caso de GUbuntu y Kubuntu. Estas versiones son también llamadas Live, del inglés "en vivo".

Como un sistema operativo sin soporte de software carece de utilidad, Linux cuenta con una plataforma de soporte de software llamados repositorios, donde se encuentran miles de programas de código libre o propietario que están vinculados con cada sistema y disponibles, a tan solo un par de pasos, desde cualquier sistema conectado a internet.

Linux es una alternativa muy buena frente a los demás sistemas operativos. Más allá de las ventajas evidentes de costo ofrece algunas características muy notables. Entre otras razones está su estabilidad, el acceso a las fuentes, la independencia del proveedor, la seguridad, la rapidez con que incorpora los nuevos adelantos tecnológicos a la escalabilidad (se pueden crear clusters de cientos de computadoras), la activa comunidad de desarrollo que hay a su alrededor, su interoperabilidad y a la abundancia de documentación relativa a los procedimientos.

Todo esto entrega condiciones óptimas para montar un sistema al que le importa principalmente la seguridad.

Hay varias empresas que comercializan soluciones basadas en Linux: IBM, Novel, Red Hat, Rxtart, Canonical (Ubuntu), Rxtart, así como miles de PYMES que ofrecen productos o servicios basados en esta tecnología.

Para el caso del presente Proyecto se escoge la distribución DSL, acrónimo de la frase DAMN SMALL LINUX (Community Damn Small Linux), la que destaca por tener mínimas funcionalidades y ofrecer múltiples instancias de expansión, lo que finalmente entrega una plataforma que obedece al perfecto balance entre lo que necesitamos y la demanda de hardware inherente a la tarea encomendada.

4.2.3 Linux como servidor

Antes de definir el software que se usará y, habiendo considerado el factor costo, también es necesario determinar el tipo de “servicio” que este software debe brindar al cliente, es decir, al usuario. Asumimos que este usuario dispone de un Smartphone o un computador que tiene un navegador web instalado. Estas premisas son consistentes con el modelo de servidor escogido pues permite procesar parte del código por el lado del servidor (base de datos, scripts php) y también por el lado del cliente (javascript, AJAX).

En este proyecto no se desarrolla esto último, tal como muestra la Figura 4-4. Se resuelve montar un sistema que sea servidor web (Apache) y servidor MySQL, que soporte un intérprete que sirva de comunicación entre ambas plataformas (PHP).

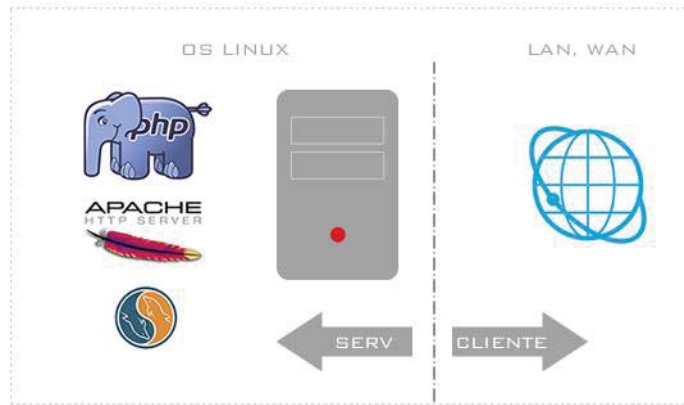


Figura 4 - 4. Relación Servidor/Cliente

La integración de PHP y MySQL tiene cierto grado de complejidad para el operador sin experiencia en manejo de servidores, sin embargo, herramientas como XAMPP instalan todas las opciones necesarias para cualquier tipo de necesidad que envuelva las tres plataformas. Su instalador está disponible de forma gratuita, incluso para versiones de Windows 32 bits (Apache Friends, <https://www.apachefriends.org>).

Una de las ventajas que incorpora esta versión de XAMPP es que, además de instalar el software básico necesario, propone instalar también un mail Server, abriendo así un sinnúmero de posibilidades en cuanto a la notificación por correo electrónico del comportamiento de la red de seguridad.

Una imagen de la interfaz de control de XAMPP se muestra en la figura 4-5.

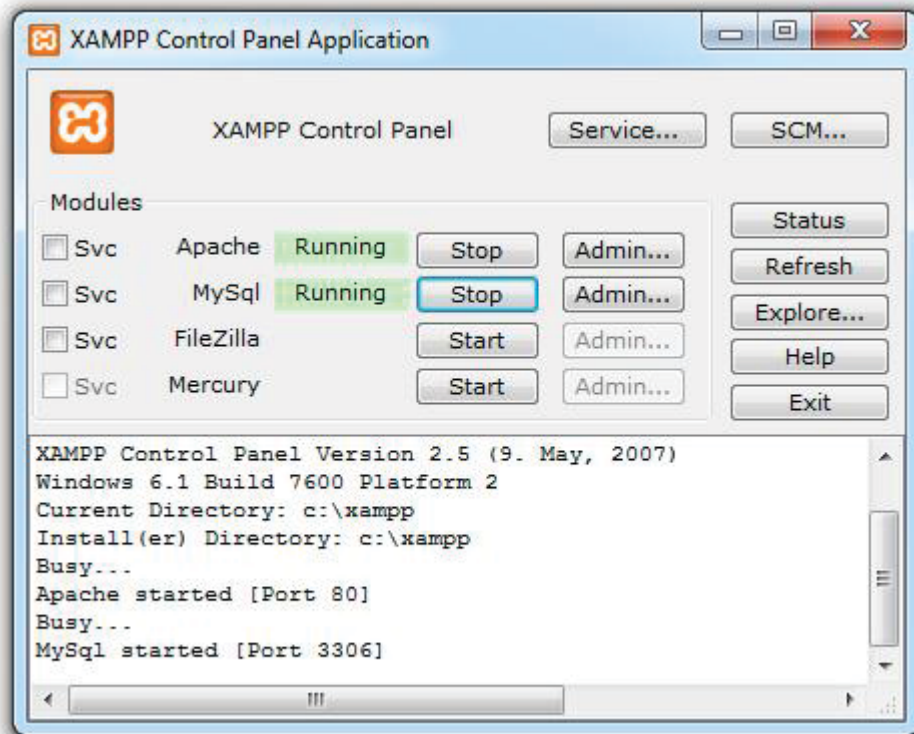


Figura 4 - 5. Herramienta XAMPP

CAPÍTULO 5: MANEJO DEL SOFTWARE

5.1 ACCESO

Para el acceso al sistema hace falta, únicamente, un intérprete de código HTML/XTML tal como Internet Explorer, Mozilla, Safari, o cualquiera de los muchos navegadores para Internet que hay disponibles en cada plataforma. Esto corresponde a uno de los objetivos principales planteados al principio del proyecto: el control del sistema de forma sencilla y multiplataforma.

5.1.1 Menú principal

Luego de asignar la dirección IP del servidor aparecen las dos opciones principales del sistema, en la pantalla del inicio, como se observa en la figura 5-1.



Figura 5 - 1. Menú principal de gestión de sistema

5.1.2 Revisar actividad

En esta opción de menú, el usuario puede monitorear cada una de las entradas que se han hecho y registrado en el sistema.

En la actual versión del software es posible revisar toda la información desde esta interfaz, tomando información de la red a intervalos de tiempo definidos por el usuario.



Figura 5 - 2. Despliegue de actividad registrada

En la figura 5-2 se identifican:

- **Fecha:** Indica la fecha exacta en la que se registró la actividad.
- **Largo:** Representa el número de bytes que fueron enviados por el módulo flotante a la central. Este dato se refleja aquí a modo de ejemplo para demostrar que es posible enviar datos de mayor complejidad y gestionarlos con miras a una futura implementación orientada a la domótica.

- **Origen:** Muestra la MAC del módulo XBee. Este número es el identificador único de hardware y que es imposible de modificar.
- **RSSI:** La fuerza de la señal con la que el módulo flotante se comunica con el módulo estructural. Este valor es de vital importancia para determinar si el usuario está, o no, dentro de una determinada área de seguridad y poder, así, aplicar la política de restricción que corresponde.

5.1.3 Autorizar nuevos módulos

Esta opción permite actualizar, modificar e ingresar nuevos módulos flotantes al sistema de seguridad. Cuando la actividad registrada no aparezca debidamente autorizada en el sistema, a través de esta herramienta, es considerada inválida y, por ende, se ejecutará el protocolo de alarmas. El ingreso se realiza en pantalla de acuerdo a la Figura 5-3 siguiente:



MENU PRINCIPAL DE CONTROL DE PRESENCIA
BASADO EN XBEE

ID RANGO ZONA

MAC

NOMBRE FECHA DE INGRESO

Figura 5 - 3. Autorizar nuevo módulo

En la Figura se identifican:

- **ID:** Muestra el tipo de solicitud especificado en el protocolo API.

- **MAC:** Determina la MAC del módulo que se está autorizando.
- **NOMBRE:** Nombre del usuario, o bien, del objeto que se autoriza.
- **Rango:** Con fines administrativos, este espacio puede ayudar a individualizar u obtener mayores datos respecto al individuo identificado.
- **Zona:** Código hexadecimal, de un byte, en el que se especifica el tipo de acceso, Fecha de Ingreso: La fecha actual.
-

5.2 SEGURIDAD DEL SISTEMA

En el presente prototipo no se contemplan medidas de seguridad informática para el sistema. Es más, con el fin de hacer más simple la intervención o modificación del código, no existen protecciones de ninguna especie al acceso directo vía FTP. Sin perjuicio de lo anterior, es posible establecer un protocolo de instrucciones para disminuir el riesgo de que la plataforma sea afectada por intrusiones desde la red.

5.2.1 Seguridad de acceso con navegadores

- **Seguridad a nivel de HTTPD:** Modificando el archivo httpd.conf del servidor Apache se puede limitar el acceso de los usuarios, basándose en direcciones IP.
- **Cookies:** A través de la autenticación del usuario por medio de un nombre de usuario y contraseña, se puede autorizar a un computador para que tenga acceso a la administración del sistema.

5.2.2 Seguridad de acceso vía FTP

- **Contraseña:** Asignar una contraseña
- **SFTP:** FTP sobre SSH es un método sencillo de implementar transferencias de archivos usando encriptación simétrica.

5.2.3 Seguridad contra uso de acceso no autorizado a administración de DB

Para impedir que personas no autorizadas lleguen a intervenir en la administración del sistema, se ejecutan las siguientes acciones:

Limitar conexiones directas: Autorizar únicamente el acceso a la administración del sistema, es decir, conexiones directas sólo desde la dirección 127.0.0.1 (localhost)

Filtro de IP: A través de la aplicación integrada en Linux IPFilter, es posible, a través de comandos, inspirados en las técnicas de switching, determinar si se está siendo víctima de una ataque informático, tomar apropiado registro de ello y, también, aplicar alguna de las reglas predefinidas para cada caso por el administrador quien es el único usuario con permisos plenos para determinar políticas de seguridad de los datos.

CAPITULO 6: PRUEBAS EN FUNCIONAMIENTO

El diagrama de flujo de la Figura 6-1, muestra el funcionamiento general del sistema y la integración de comunicación por las redes de los dispositivos ZigBee, la comunicación serial, la administración de la información con servidores LAMP y, finalmente, el accionamiento remoto de los relés SPDT de alarma sectorizados .

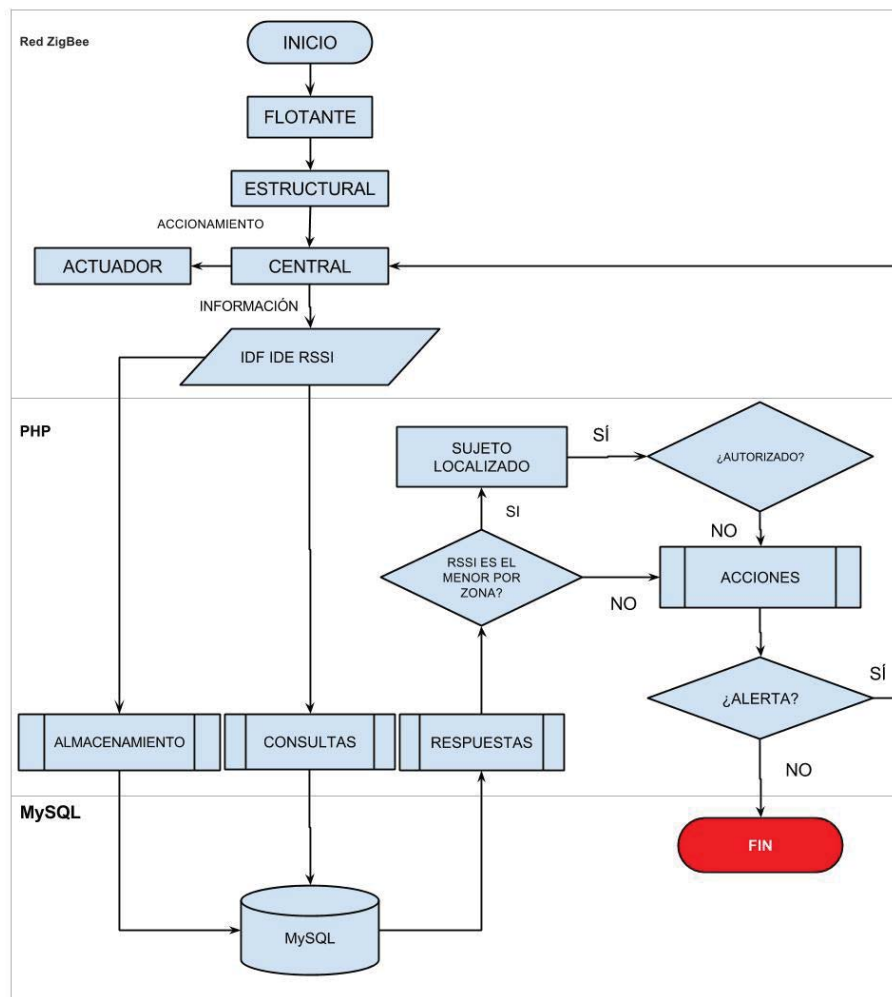


Figura 6 - 1. Flujo general del sistema

Para realizar las pruebas de operación del sistema proyectado, se construyeron cuatro prototipos de módulos de red, con sus respectivos programas incorporados.

La integración del sistema completo fue satisfactorio con la participación de los cuatro módulos interconectados en red, comunicándose entre sí y con el servidor. A partir de estos resultados, se estima que es posible escalar esta solución tal como está.

También, dado que, teóricamente la red ZigBee puede soportar una cantidad de 65535 dispositivos, en la práctica, distintos usuarios y estudiosos de este tipo de redes recomienda utilizar sólo un máximo de 100 unidades dentro de la misma red. Esa es una recomendación que deberían tener en cuenta nuestros clientes.

CAPÍTULO 7: EVALUACIÓN ECONÓMICA

7.1 PREMISAS

El producto diseñado, comercialmente se puede llamar MiniCat, éste aspira a convertirse en la llave de entrada al mercado nacional de la domótica, utilizando la seguridad domiciliar e industrial como plataforma para ello.

Un MiniCat promete ser mínimamente invasivo a los sistemas de seguridad ya instalados y de fácil empalme, pudiendo quedar bajo la responsabilidad de personal sin entrenamiento técnico calificado.

Mientras la domótica lucha por entrar en el mercado nacional siempre se encuentra con la misma barrera, “el costo de inicial del producto para el cliente”, que para los estándares norteamericanos y europeos no es tan grande; pero, para la realidad nacional es un disuasivo importante. Una red MiniCat puede hacer un invaluable aporte con un bajo costo de producción y total dispensación en mano de obra calificada, rompiendo así la dinámica de un mercado que lucha por su prevalencia, desde mediados de los años noventa. Este factor costo, es uno de los aspectos más importantes en la evaluación económica del proyecto, siendo por esto uno de los pilares constituyentes en las proyecciones del negocio.

Respecto al costo del proyecto, incluyendo su desarrollo, y tomando en cuenta que este sistema funciona bajo el concepto de red, no sería adecuado para el cliente considerar únicamente comprar un solo módulo. Todavía más, por las características de comercialización de las piezas que los distribuidores norteamericanos, europeos y chinos han fijado, no tiene una justificación financiera el importar únicamente un módulo. El sobreprecio que se paga por comprar una cantidad menor a los 5 módulos puede llegar al 100%.

Como es ahora transparente, la compra de módulos en pequeño número, esto puede afectar, de forma importante, la rentabilidad del negocio.

El punto anterior, nos obliga a elevar la cantidad de componentes a cubrir en la inversión inicial, y para esos propósitos se estima recurrir a las iniciativas CORFO bajo su fondo denominado INNOVA, las que contemplan el apoyo a proyectos de

emprendimiento que puedan aportar, con un enfoque diferente, a una determinada demanda de mercado. Esta es evaluada por comités integrados por capitalistas y funcionarios de Corfo con experiencia en proyectos de innovación.

Importante es recalcar que los fondos, de apoyo al emprendimiento, que se desprenden de la gestión de INNOVA son préstamos no retornables, lo que quiere decir que, en su primera fase, son considerados capital de riesgo.

La adjudicación del préstamo INNOVA consta de dos etapas: La primera por un monto no mayor a los 6 millones de pesos y la segunda por un monto no mayor a los 50 millones de pesos. Lo anterior implica aportes del 10% de lo prestado por parte del solicitante y el pago de los honorarios de los "patrocinadores" que son fijados por la ley.

El procedimiento para la adjudicación de los fondos es a través del análisis del mérito del negocio por especialistas, miembros de fundaciones intermedias que gestionan el préstamo con la CORFO, los que son llamados "patrocinadores". Estos patrocinadores forman parte de un directorio de empresas consultoras, entre las cuales se encuentran varias universidades a lo largo del país. Para la Quinta Región destaca la Universidad Técnica Federico Santa María con su oficina 3ie, incubadora bastante exitosa que ha financiado varios proyectos que hoy se erigen como solventes y bien constituidas empresas.

7.2 ELEMENTOS Y CONDICIONES PARA EL ANÁLISIS

Habiendo ya definido en etapas anteriores las características técnicas y las aspiraciones comerciales del producto, queda por definir un modelo de negocios de tienda (producción, venta sin restricciones ni servicios asociados) que se escoge como el más apropiado para el desarrollo de la idea.

El costo de desarrollo del dispositivo está constituido por el valor de horas hombre más las placas de desarrollo disponibles desde Digi.com.

El servidor que gestionará la información será considerado un requerimiento del sistema, siendo responsabilidad del cliente proveerse de uno que cumpla con las especificaciones técnicas descritas anteriormente.

Para facilidad de cálculo y, para lograr cubrir a lo menos dos zonas de control de presencia, se considera ofrecer un paquete básico que incluye diez módulos y la venta futura de módulos independientes que sean integrables a redes con MiniCats existentes. Cualquier módulo que compone el sistema MiniCat estará compuesto por los elementos que se muestra en en la tabla 7-1:

Tabla 7 - 1. Costos por unidad para construcción de módulo de comunicaciones MiniCat

Costo por unidad	USD
Módulo Xbee	\$ 24,95
PIC16f877a	\$ 5,93
MAX232	\$ 1,95
Placa	\$ 1,00
Caja	\$ 3,00
Relé	\$ 1,50
Otros	\$ 3,83
Manufactura	\$ 1,00
Total costo USD	\$ 43,16
Dólar observado	550
Total costo CLP	\$ 23.740

El valor de los módulos que integran el sistema está calculado bajo los valores para importación directa con un margen de comercialización de 26% tal que alcance un valor aproximado por MiniCat de \$30.000CLP. Este punto es muy importante, pues una de las premisas fundamentales para entrar al mercado implica: un costo asociable a otras soluciones existentes de seguridad domiciliaria.

Para el problema de las cajas, en primera instancia, se decide comprar cajas prefabricadas.

En futuras etapas se debe incorporar matricería y serigrafía para fortalecer la imagen corporativa y el valor de la marca. En la tabla 7-2 están los costos generales para el inicio de la producción.

Tabla 7 - 2. Inversión inicial

Ingeniería	\$2.500.000
Placas de desarrollo	\$200.000
Servicios	\$200.000
Insumos	\$100.000
Desarrollo Web	\$600.000
Iniciación de actividades	\$400.000
Vehículo	\$800.000
Inversión Inicial	\$12.000.000

Se persigue llegar al cliente a través de publicaciones y de un sitio web adecuadamente diseñado, integrado al sistema de pago en línea PayPal, que ampliaría las posibilidades de comercio fuera de las fronteras de la ciudad, e incluso del país, a cambio de un costo prácticamente nulo.

Dentro de los honorarios se incluye, para una mayor confiabilidad de cálculo, la demanda de honorarios del propietario de la iniciativa, monto que le dará más herramientas para flexibilizar las apreciaciones del rendimiento una vez que el proyecto esté en marcha. El marketing es esencial para la promoción constante del producto.

Lo expresado está detallado en cifras en la tabla 7-3

Tabla 7 - 3. Costos operacionales estimados

Costos operacionales en fase de producción	
Honorarios especialistas	\$ 1.200.000
Honorarios Administrativos	\$ 1.000.000
Honorarios ventas	\$ 500.000
Insumos	\$100.000
Servicios	\$100.000
Marketing	\$200.000
Costos operacionales	\$3.450.000

Bajo las condiciones anteriores, un presupuesto típico para un cliente que espera controlar 2 zonas y manejar 4 usuarios, debería ser de aproximadamente \$ 210.000 más el impuesto IVA legal.

Se estima un aumento en la producción, optimista, de un 10% con respecto al mes anterior.

7.3 ANÁLISIS DEL PERFIL ECONOMÉTRICO

El estudio económico financiero realizado al proyecto comercial MiniCat considera la estimación de costos de operación e ingresos por ventas del producto para un horizonte de evaluación de 6 meses. La razón para este plazo radica en que la base del emprendimiento está casi totalmente en los capitales no retornables CORFO y sus límites de aporte (50 millones de pesos chilenos que representen el 90% del aporte total de capital) más el aporte de los socios (6 millones sin interés sobre el capital) permiten construir, al arranque, una cantidad máxima aproximada 2000 MiniCat.

Teniendo como objetivo una rentabilidad del 20%, un estudio de sensibilidad a esta variable con respecto a la cantidad de unidades producidas al inicio, determina los siguientes datos:

Tabla 7 - 4. Datos de análisis de TIR v/s Producción inicial

Venta primer mes	TIR	Aporte Corfo	Capital privado/Riesgo
1000	2%	\$35.271.000	\$ 3.919.000
1500	8%	\$45.954.000	\$5.106.000
2000	11%	\$56.000.000	\$6.930.000
2500	11%	\$56.000.000	\$18.800.000
3000	10%	\$56.000.000	\$30.670.000
3500	10%	\$56.000.000	\$42.540.000
4000	10%	\$56.000.000	\$54.410.000
4400	11%	\$56.000.000	\$63.906.000

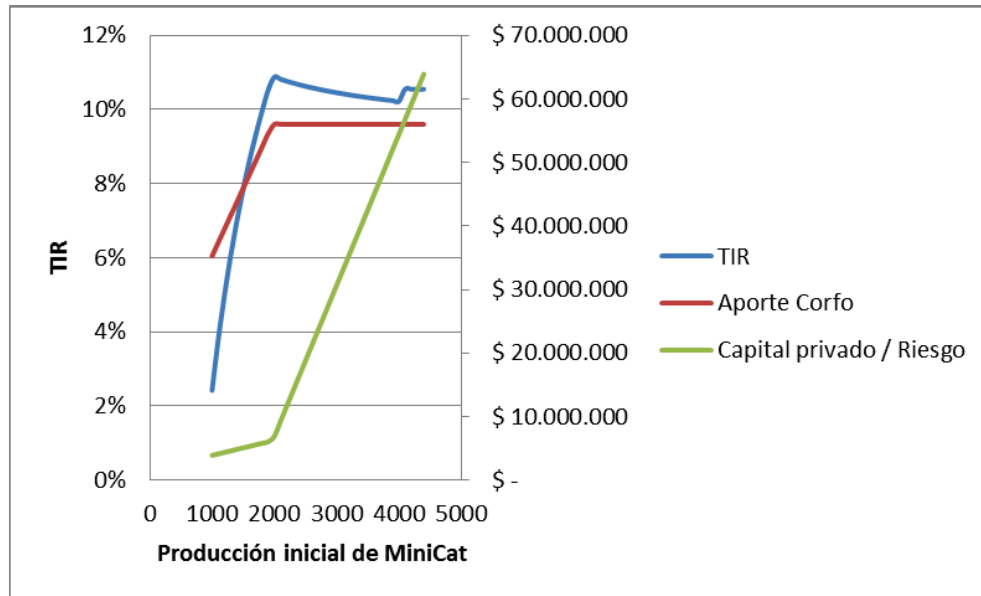


Figura 7 - 1. Gráfico de Sensibilidad de TIR v/s producción inicial

En la Tabla 7-4, se aprecia que la rentabilidad del negocio es afectada positivamente por el aporte inicial de CORFO, tergiversando la real tendencia del negocio, que se aprecia en la baja de la rentabilidad tan pronto como el aporte de CORFO llega a su límite. La columna llamada “CAPITAL PRIVADO/RIESGO” es el aporte que hacen los socios y que, para los efectos del estudio se han incorporado, la tabulación muestra que, casi al final de la gráfica, la rentabilidad del negocio va de la mano de los montos de inversión. Es muy importante si consideramos que se trata de una empresa que emprende pues, para soportar el nivel de producción propuesto, demanda de aportes privados y un total de capital de trabajo de \$3.919.000 de pesos. Obviamente, este negocio presenta características típicas del mercado de comercialización masiva y, es claro, que el porcentaje de comercialización, de 26%, no es suficiente para alcanzar el nivel de rentabilidad que se le exige al proyecto. Tal afirmación se ratifica con posterioridad en este documento, desde un punto de vista más a largo plazo.

7.4 FINANCIAMIENTO

El proceso para acercarse a Innova de CORFO, es a través de un patrocinador quien debe representar el proyecto ante varias fases en un comité en donde se decide, en un proceso relativamente corto, si el proyecto merece ser financiado.

El primer paso es presentar una solicitud para reunirse con un 'ejecutivo de cuenta'. Esta incluyendo algunos detalles relevantes para la oficina de gestión ocasión en que recogerá información básica del proyecto y que les servirá de instrumento para plantear las preguntas en la primera reunión.

El formato de la hoja de solicitud, para la oficina 3ie de la UTFSM, se encuentra adjunto en este informe.

Cabe señalar que el proyecto INNOVA no financia prototipos, es decir, que los solicitantes deben contar con un módulo funcional, que en este caso debieran ser entre 4 y 6 módulos.

7.5 ANÁLISIS FINANCIERO

Se ha postulado que la rentabilidad del negocio tiene una tendencia alcista; pero, con un margen de comercialización mayor desconocido al determinado en la hipótesis. Para responder esta interrogante se evalúa el proyecto con un horizonte de 6 años (Apéndice B) con varios márgenes de comercialización, obteniendo así la siguiente información:

Tabla 7 - 5. TIR en función de margen de comercialización y de unidades producidas al inicio
Margen de comercialización

		26%	30%	34%	38%	42%	46%
Unidades producidas al primer año	21384	3%	6%	10%	13%	16%	20%
	23522	3%	7%	10%	14%	17%	20%
	25875	4%	7%	11%	14%	18%	21%
	28462	4%	8%	11%	15%	18%	21%
	31308	5%	8%	12%	15%	19%	22%
	34439	5%	9%	12%	16%	19%	22%
	37883	6%	9%	13%	16%	20%	23%
	41671	6%	9%	13%	16%	20%	23%

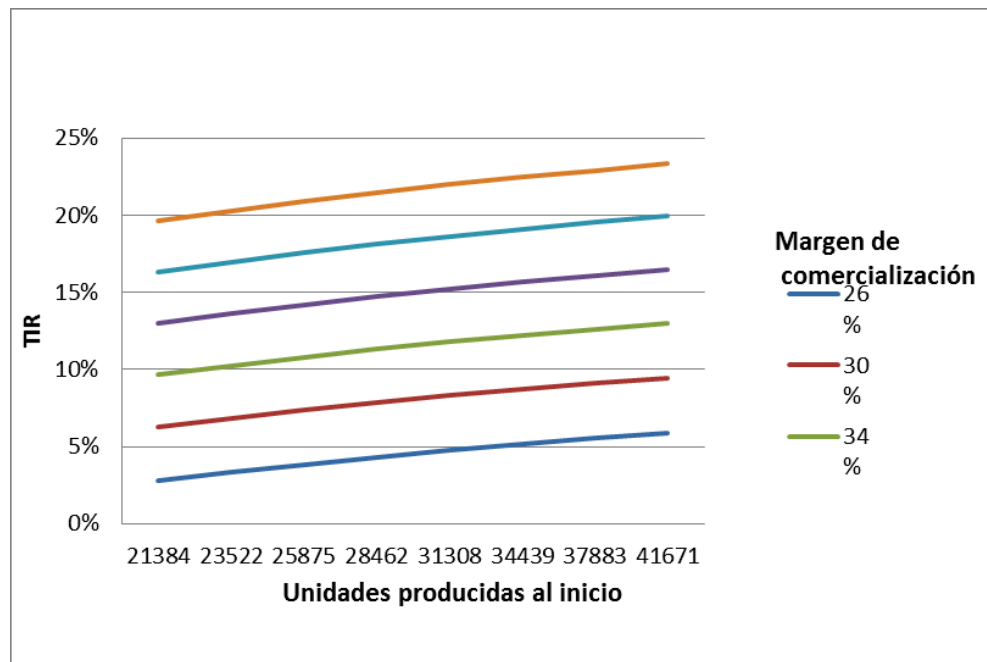


Figura 7 - 2. Gráfico TIR en función de margen de comercialización y de unidades producidas al inicio

La tabla muestra que el margen de comercialización del producto debe estar en el 46% y no en el 26%. Se supone que un aumento del 20% en el precio objetivo

nos aleja demasiado de la franja de mercado que no invierte en nuevas tecnologías. Si se tiene en consideración que, en la actualidad, los precios de kits de seguridad domiciliaria siguen bajando sus precios hasta por debajo de los 100USD y que se pueden importar directamente desde China, con un período de arribo de 60 días, hace que aquellos clientes que hoy compran en Chile lo hagan en el exterior. Logran adquirir productos de marcas de prestigio que están bien representadas y que, por el tamaño del mercado, siguen siendo pocas.

La asociación de marca con empresas de mayor envergadura y presencia actual en el mercado podría solucionar este problema, o bien, propiciar un cambio en la percepción del precio pero con una demanda de inversión en marketing mucho más poderosa.

Se puede decir que el mercado actual necesita los MiniCat's Pero, para lograr el ingreso de la domótica masivamente a las casas y a la industria, se necesita de una presión que no se ha hecho sentir después de más de 20 años de infructuosos intentos, por parte de la empresa privada, ya desencantada con este rubro.

CONCLUSIONES

En este proyecto se propuso diseñar un sistema teniendo éste los elementos de: detección de presencia, envío de información por radiofrecuencia, clasificación y despliegue de la información en sistemas multiplataforma, todo enfocado a la seguridad, revestido de simplicidad al mínimo costo posible.

El desarrollo de este proyecto y su proceso de programación bien estructurado, colabora con la generación, no sólo del producto final, sino que proporciona una técnica heredada a aquellos nuevos ingenieros y estudiantes que escojan esta como plataforma de desarrollo de sus actividades.

En el aspecto financiero, hay que aceptar que la domótica tiene, comparado con las soluciones en seguridad existentes, un alto costo. Mientras no se masifique su uso, ese costo adicional lo deberán pagar los sectores que se aventuren en ella.

La domótica tiene múltiples aplicaciones: seguridad, multimedia, ahorro de energía entre muchas otras, y se aplicará masivamente en hogares cuando mejore el ingreso promedio en el país y las personas puedan costearlo. Desde el principio se planteó que la seguridad "Indoor", es un factor importante al considerar inversiones en el hogar, por lo que esta podía ser la llave de entrada para la domótica, pero no. La nueva tecnología aún no puede competir con los precios vigentes en ese mercado.

Aunque, como se ha dicho, no se recomienda la fabricación de este Proyecto para su comercialización orientada al mercado domiciliario, considero que el Proyecto acierta y se adelanta a su tiempo en dos puntos esenciales: la gestión de la información por medio de una plataforma en línea y, en segundo lugar, la buena recepción que tiene el concepto de un sistema que cumple con localizar a personas y objetos en un espacio controlado usando redes tipo malla. Respecto a la primera característica la hemos visto consolidarse en todo software de planificación de recursos que existe hoy en día y, respecto a la segunda, ya existen proyectos que proponen este servicio con resultados interesantes (San Jose Medical Center Installs ZigBee-based RTLS Across 10 Buildings, RFID Journal, Marzo 19th of 2010).

Como conclusión, de la elaboración y prueba de los módulos y del sistema proyectado, vista la evaluación económica de éste, se recomienda, por ahora, no fabricar el modelo MiniCat pues no es factible por ser marginalmente rentable y de muy alto riesgo financiero.

BIBLIOGRAFIA

Aumento de inseguridad refleja encuesta Paz Ciudadana [Artículo] / aut. La Segunda // La Segunda. - Santiago : [s.n.], 2008.

¿Cuánto cuesta estar seguro en casa? [En línea] / aut. Vicente Manjavacas, http://www.consumer.es/web/es/vivienda/comunidades_vecinos_y_legislacion/2008/03/13/175393.php. -

Sitio corporativo [Informe] / aut. Ubuntu Community (www.ubuntu.com). - 2015.

Sitio principal de Microchip Inc: <http://www.microchip.com/downloads/en/DeviceDoc/39582C.pdf> [Datasheet 16F877A] / aut. Microchip Inc.. - 2001. - Vols. Pag 48-69

Sitio principal DSL (Damn Small Linux) <http://www.damnsmalllinux.org/> [En línea] / aut. Comunity Damn Small Linux.

XAMPP [En línea] / aut. Apache Friends, <https://www.apachefriends.org> // <https://www.apachefriends.org>.

SII, OPERACIÓN RENTA 2016, tabal de depreciaciones, páginas 40 – 45.

http://www.sii.cl/renta/suplemento/2006/pag_n041.pdf

Php Serial [En línea] / aut. Remy Sánchez, Instructions to developing applications based on PHP to serial projects // <http://hyperthese.net>. 2006

**APÉNDICE A: SOLICITUD DE FINANCIAMIENTO PARA CAPITAL SEMILLA
CORFO**

PERFIL DE PROYECTO

ANTECEDENTES PROYECTO

Nombre Proyecto	MiniCat
Nombre Emprendedor(es)	Rodolfo Flores
RUT (s),	13.265.498-1
Área del Proyecto(sector destino)	Seguridad, Herramientas para la Ingeniería, Domótica.
Teléfono	9xxxx
E-Mail	rodolfoflores@terra.cl
Dirección	Almagro XXX, Viña del Mar

DESCRIPCIÓN DE LA IDEA DEL PRODUCTO O SERVICIO

Un producto en una caja de dimensiones 6x6x4 que contiene un módulo de comunicaciones XBeePro (Digi Inc.) con tecnología Zigbee y un micro controlador PIC, orientados a ser interconectados a un sensor de salida análogo/digital. Se caracteriza por estar preparado para manejar hasta seis tipos de sensores al unísono, con una programación que define un comportamiento ad hoc para tales fines. Estos sensores podrán ser infrarrojos, sensores de nivel, de humo de barras tensoras y sensores volumétricos – para el prototipo, este último- constituyendo una solución modular para el mercado de instalación de redes de seguridad indoor. Los componentes de este producto se encuentran a la venta en el mercado nacional,; pero su precio es más atractivo en el mercado norteamericano y europeo, por lo que se espera contar con un nivel de inversión que permita acceder a la importación de ellos. Al adquirirlos directamente de distribuidores extranjeros como Digi, Motorola y Atmel se consigue una inmejorable calidad en los componentes y soporte postventa para nosotros e indirectamente, a nuestros

clientes. El diseño del programa y de la electrónica quedará en manos de la propia sociedad. La manufactura de la placa y la construcción del chasis, serán encargadas a empresas especialistas en Santiago. Una vez construidas las partes, en el taller, ubicado en el Cerro Alegre de Valparaíso, se procede al ensamblaje y puesta a punto para la venta. Se contempla la adquisición de un dominio .cl, la renta de un hosting y el diseño de un sitio web, definiendo así los canales mediante los cuales comercializaremos y promocionaremos el producto.

FACTORES DE INNOVACIÓN

Simpleza en su instalación por su modularidad, flexibilidad y bajo costo de mantenimiento.

Solución al problema de cableados para alarmas, sensores y actuadores de uso común. Las compañías de seguridad y sus clientes valorarán el hecho de poder instalar un sistema sin tener que hacer perforaciones adicionales. Las empresas de domótica verán en esto una plataforma de bajo costo que ayuda a la introducción de sus productos en el mercado nacional.

Ofrecemos un nivel de desarrollo avanzado para un sistema que muestra una masificación de características explosivas y que, como producto terminado, no existe en el mercado nacional. Técnicos y Profesionales podrán ampliar su gama de servicios de automatización gracias a esta herramienta.

COMPETENCIA Y SUSTITUTOS

Nuestro producto, implica la participación de ingenieros para desarrollar un sistema de utilidad directa para el cliente final. Hasta el momento se han desarrollado sistemas muy completos basados en la tecnología Zigbee. Pero, hoy en día, son conocidas estructuras básicas que entregan una solución genérica para las necesidades del mercado masivo, son reconocidas como una herramienta fácil de utilizar y que ponen las nuevas tecnologías en manos de un usuario no experto.

Nuestro producto persigue cubrir este mercado dotando a un sistema de red de sensores inalámbricamente interconectados, de una interfaz y de la metodología fácil de comprender y utilizarlo,. Aprovechando la confiabilidad, el bajo

costo de producción, la confiabilidad y el soporte de la tecnología Zigbee. Todo lo anterior puesto dentro del concepto de unidad bajo una marca reconocible.

Nuestro producto supera por mucho las prestaciones y fiabilidad de los sistemas de redes por cable, siendo éste el medio mayor difusión principalmente por el bajo costo de sus materiales.

DESCRIPCIÓN DEL MERCADO OBJETIVO

Con el tiempo, el "usuario final" es nuestro mercado objetivo; mercado que esperamos alcanzar brindando soluciones en el campo de la seguridad domiciliaria, para luego expandirnos a otros campos a partir de la gran versatilidad y modularidad de nuestra herramienta.

Esperamos poder contar con el apoyo de empresas de seguridad y de distribuidoras capaces de canalizar hacia sus clientes el gran potencial de nuestro producto. Ello constituye la forma innegable que puede ayudarles a potenciar su inversión en seguridad, control remoto, domótica, y un sinnúmero de otras funciones.

DESCRIPCIÓN DEL MODELO DE NEGOCIOS DEL PRODUCTO O SERVICIO

El primer paso es desarrollar una página web, en donde el producto sea mostrado como una solución práctica y de fácil instalación. Definiremos para el sitio web un esquema que mezcle adecuadamente información y compras en línea, palanca que nos brindará la posibilidad de alcanzar el mercado fuera de Chile usando PayPal.

Lo primero es el mercado nacional, y por eso planeamos distribuir publicidad en prensa escrita especializada y masiva, siempre apuntando al mercado objetivo primario que son las empresas de seguridad, potenciando la inversión en marketing. Buscando el enganche constante hacia la página web de nuestra empresa, en donde estará destacada la sencillez de la construcción de redes de múltiples sensores volumétricos, que hoy es el producto de mayor venta para nuestros clientes objetivos. El margen de ganancia se verá optimizado al momento de importar partes desde el extranjero. Además, dado el volumen de compra requerido para una importación deberemos contar con un stock de nuestros

productos lo que nos permite brindar un nivel de servicio de entrega muy conveniente.

Importante será el aglomerar bajo un determinado nombre comercial todos los dispositivos y componentes necesarios para construir una red de seguridad bajo el concepto "all-include" a un precio aún más atractivo, haciendo todavía más hincapié en su flexibilidad y fácil manejo.

ESTIMACIÓN DE LOS RESULTADOS ESPERADOS

Es demostrable que el margen de comercialización del producto debe estar en el 46%. Asumimos que tal margen de comercialización nos aleja demasiado de la franja de mercado que no invierte en experimentar con nuevas tecnologías. Si tomamos en consideración que, en la actualidad, los precios de kits de seguridad domiciliaria siguen bajando sus precios de venta hasta por debajo de los 100USD y que se pueden importar directamente desde China con un período de arribo de 60 días hace que aquellos clientes que hoy compran en Chile lo hagan basándose en el prestigio de marcas bien representadas y que, por el tamaño del mercado, siguen siendo unas pocas. La asociación de marca con empresas de mayor envergadura y presentes actualmente en el mercado, podría solucionar este problema. Quizás esto propiciar un cambio en la percepción del precio respecto a las funcionalidades esperadas para una de estas soluciones. Así la nueva demanda permitirá una inversión en marketing mucho más poderosa.

Se puede decir que el mercado actual necesita los MiniCat's para lograr el ingreso de la domótica a las casas y a la industria, y para hacerlo se requiere del aporte de capitales privados con líneas de mercado y distribución ya establecidas.

LISTADO DE LAS PRINCIPALES ACTIVIDADES Y RECURSOS NECESARIOS

Ingeniería	\$2.500.000
Placas de desarrollo	\$200.000
Servicios	\$200.000
Insumos	\$100.000
Desarrollo Web	\$600.000
Iniciación de actividades	\$400.000
Vehículo	\$800.000
Inversión Inicial	\$12.000.000

En fase de producción:

Costos Operacionales en fase de producción	
Honorarios especialistas	\$ 1.200.000
Honorarios Administrativos	\$ 1.000.000
Honorarios ventas	\$ 500.000
Insumos	\$100.000
Servicios	\$100.000
Marketing	\$200.000
Costos Operacionales	\$3.450.000

DESCRIPCIÓN DEL EMPRENDEDOR Y SUS SOCIOS

Rodolfo Flores, ... (Breve descripción personal)

Acerca del proyecto	I	O
Cuento con un prototipo o está presente en su etapa final de desarrollo		
La empresa está constituida(*)		
Desea postular a Incubarseen 3IE		
Desea postular Capital Semilla de CORFO		
Busca inversionistas para el proyecto		

(*) Si la empresa está constituida, indicar:

Fecha de Constitución			
-----------------------	--	--	--

APÉNDICE B: CÁLCULO DE RENTABILIDAD A 6 CICLOS

ANALISIS ECONOMETRICO DE PROYECTO MiniCat

		Margen Comercialización			26%				
		Costo Unitario \$			23.740				
Inversión	12.000.000								
Terreno y Edific.	0								
Maquinarias	8.200.000								
Otros	3.800.000								
		Gastos Mensuales							
		Administrativos	9000000						
		Honorarios	32400000						
Producción Mensual (Unidades)		21384	23.522	25.875	28.462	31.308	34.439		
		Meses							
		0	1	2	3	4	5	6	
Ingresos Oper.			639.646.762	703.611.438	773.972.582	851.369.840	936.506.824	1.030.157.506	
Costos Oper.			(507.656.160)	(558.421.776)	(614.263.954)	(675.690.349)	(743.259.384)	(817.585.322)	
Utilidad Bruta			131.990.602	145.189.662	159.708.628	175.679.491	193.247.440	212.572.184	
Gastos Adm.			(41.400.000)	(41.400.000)	(41.400.000)	(41.400.000)	(41.400.000)	(41.400.000)	
Depreciación			(166.667)	(166.667)	(166.667)	(166.667)	(166.667)	(166.667)	
Amort. Act Nom.			(120.000)	(120.000)	(120.000)	(120.000)	(120.000)	(120.000)	
Gastos Financieros			0	0	0	0	0	0	
Base Imponible			90.303.935	103.502.995	118.021.961	133.992.824	151.560.773	170.885.517	
Impuestos (20%)			(18.060.787)	(20.700.599)	(23.604.392)	(26.798.565)	(30.312.155)	(34.177.103)	
Utilidad Despues Impuestos			72.243.148	82.802.396	94.417.569	107.194.259	121.248.619	136.708.414	
Depreciación			166.667	166.667	166.667	166.667	166.667	166.667	
Pago Cap Prestado			(84.176.027)	(84.176.027)	(84.176.027)	(84.176.027)	(84.176.027)	(84.176.027)	
Aporte Cap Trabajo			(549.056.160)						
Recup Cap Trabajo								549.056.160	
Ahorros/desahorros			0	0	0	0	0	0	
Préstamo(15% anual)			0						
Flujo Caja Neto			(561.056.160)	(11.766.212)	(1.206.964)	10.408.209	23.184.899	37.239.259	60.1755.214
TIR			3%						
VAN			(169.360.898)						
TIRM			5%						

**APÉNDICE C: FUNCIONES DE CÓDIGO PHP PARA GESTIÓN DE
INFORMACIÓN API XBEE**

```

////////////////////////////////////
//CONTROL DE PUERTO SERIE
////////////////////////////////////

function sendEnter (){
    $output = array();
    $return_var = 0;
    Exec("sendEnter.exe", $output, $return_var);
}

Function openSerial(){
    sendEnter();
    ser_open("COM3", 9600, 8, "None", 1, "None");
}

function closeSerial(){
    ser_close();
    sendEnter();
}

//GENERA EL dumpHexSerial.txt
function dumpHex(){
    exec("sfk151 hexdump logSerial.txt>dumpHexSerial.txt");
}
//DEVUELVE UN ARRAY CON TODAS LAS API QUE HAN LLEGADO
function extractDumpHex(){
    $dump =
$_SERVER['DOCUMENT_ROOT']."/XBee/dumpHexSerial.txt";

//ABRO EL ARCHIVO $DUMP Y TABULO TODAS LAS ENTRADAS QUE
EMPIECEN CON 7E
    $file = fopen($dump, "r") or exit("Unable to open file!");

    While(!feof($file))
    {

```

```

        $linea = fgets($file);
        $linea = ereg_replace(">",",",$linea);

        $contenido = explode('<', $linea);
        $linea = ereg_replace(' ',",",$contenido[0]);
        $temp.= $linea;
    }
    $contenido = explode('7E', $temp);

    Fclose($file);

    //ELIMINO LA PRIMERA ENTRADA EN FALSO
    $comando = array();
    for($i=1; $i<count($contenido); $i++){
        $comando[$i-1]=$contenido[$i];
    }

    return $comando;
}
//DEVUELVE UN ARRAY DE EL API QUE SE LE INGRESE

function apiWord($API){
    aquí falta

        $odd=(strlen($API)%2);

        for($i=0; $i<(strlen($API)/2)+$odd;$i++){
            $byte = substr($API, 2*$i,2);

            $word[$i]=$byte;
        }

    return $word;
}

//IDENTIFICA LOS COMPONENTES DE UNA API
function apiOut($API){

        $byte = apiWord($API);

        switch($byte[2]){

        case 80:

            $apitype= 80;
            $lenght = hexdec($byte[1]);

```

```

        for($i=3; $i<3+8; $i++){
            $addr. = $byte[$i];
        }
        $rsi = hexdec($byte[11]);
        for($i=13; $i<$lenght+2; $i++){
            $content. = $byte[$i];
        }

        $label = array(largo, addr, rssi, content);
        $dato = array($lenght, $addr, $rsi, $content);
        $salida = array_combine($label, $dato);

        break;

        //SE DESCARTA EL PAQUETE COMPLETO
        break;
    }

return $salida;
}

////////////////////////////////////
//                                TX API
////////////////////////////////////

//GARANTIZA LA ESTRUCTURA EN BYTES
function minLenByte($lsb){
    $lsb          = (strlen($lsb)%2!=0)?'0'.$lsb:$lsb;

return $lsb;
}

//CONSTRUYE LA PALABRA API Y ENTREGA UN STRING
function constructApi($datos, $addr, $ack){
    $msb = '00';
    $lsb          = dechex((strlen($datos)/2) + (strlen($addr)/2 +
3);

```

```

$lsb          = minLenByte($lsb);
$ack          = ($ack==0)?'00':'01';

$inicio       = '7E';
$largo        = $msb.$lsb;
$sapiType     = '00';
$idf          = '01';

$ack          = $ack;
$datos        = $datos;

$word = $inicio.$largo.$sapiType.$idf.$addr.$ack.$datos;

$temp = apiWord($word);

    for($i=3; $i<count($temp); $i++){

        $byte = $temp[$i];
        $byte = hexdec($byte);
        $chksum += $byte;

    }

$chksum = 255-$chksum;

$chksum = dechex($chksum);
$chksum = substr($chksum, -2);
$chksum = minLenByte($chksum);

$word = $word.$chksum;

return $word;
}

//ENVIA BYTE A BYTE POR EL PUERTO SERIE
function txApi($datos, $addr, $ack){

    $sapiWord=constructApi($datos, $addr, $ack);

    $sapiWord = apiWord($sapiWord);

    for($i=0; $i<count($sapiWord); $i++){

        $byte = $sapiWord[$i];
        $byte = hexdec($byte);
        ser_writebyte($byte);

    }
}

```

**APÉNDICE D: FUNCIONES EN CÓDIGO C PARA GESTIÓN DE INFORMACIÓN
API XBEE**

```

//_____ INICIACIÓN DE VARIABLES

unsigned short i, n, LMSB, LLSB, CHK, TX, tempn, alarma, compara;
char tempw;
unsigned int LARGO;
unsigned short ADDR64[] = {0x00,0x13,0xA2,0x00,0x40,0x08,0x8A,0Xcd};
unsigned short NUM[] = {0x00,0x13,0xA2,0x00,0x40,0x06,0x33,0x37};
unsigned short RCV [10];

////////////////////////////////////
//      INTERRUPTACIONES
////////////////////////////////////

//_____ INT SETUP

void setup(void) {
INTCON = 0b11000000;
PIE1   = 0b00100000;
}
//_____ INTERRUPT

void interrupt(){
compara = Usart_Read();
if(compara == 0x01){
    alarma = 0x00;
}
//PORTD=0xFF;
RCREG = 0;
PIE1 = 0b00100000;
INTCON=0b11000000;
}
////////////////////////////////////
//      FUNCIONES
////////////////////////////////////
//_____ SERIAL CHAR

void USART_str(char *g){
    while (*g){
        Usart_Write(*g++);
    }
}
//_____ ARRAY POR SERIAL

void Array2Serial(unsigned short NUM[], int NumBytes){
for(i=0;i<NumBytes;i++){
        Usart_Write(NUM[i]);
    }
}
//_____ CLEAN RCV

void cleanRCV(unsigned short i){

    for(i; i<sizeof(RCV); i++){

        RCV[i]=0x00;

    }
}
//_____ Filtrar RCV

void extractRCV(){
    unsigned short a, b, c, d, firstByteApi, lengthApi, lastByteApi;

```



```

for(a=0; a<sizeof(RCV); a++){
    if(RCV[a]==0x7E){
        lengthApi=a+2;
        lengthApi=RCV[lengthApi]; //Largo API
        firstByteApi=a+3 //Primer Byte relevante
        lastByteApi=firstByteApi+lengthApi; //Último Byte relevante
        c=firstByteApi;
        d=lengthApi;
        b=0;
        for(c; c<lastByteApi; c++){ //Muevo los Bytes relevantes
            RCV[b]=RCV[c]; //hacia el principio del array
            b++;
        }
        CleanRCV((lengthApi)); //rellena con ceros
    }
}
}
}
//_XBEE_local////////////////////////////////////
void XBee_str(short APIID, short ACK, char *g, unsigned short NUM[], int NumBytes){
    If(LARGO > 0x00FF){
        LMSB = LARGO-0x00FF;
    }
    else{
        LMSB = 0x00;
    }
    LLSB = LARGO;
    switch(APIID){ //API ID:
        case 1: //1: AT Command
            APIID = 0x08; //2: AT Queue Parameter Value
            break; //3: ETC...
        case 2:
            APIID = 0x09;
        default:
            APIID = 0x09;
    }
    switch(ACK){ //ACK:
        case 0: //0: DISABLE
            ACK = 0x00; //1: ENABLE
            break; //Cuando APIID=2, ACK=FRAMEID
        case 1:
            ACK = 0x01;
            break;
        default:
            ACK = ACK;
    }
    CHK=0;
    while(*g){
        CHK += *g++;
    }
    g=tempw; //REW
    for(i=0; i<NumBytes;i++){
        CHK += NUM[i];
    }
    CHK = 0xFF-(CHK+APIID+ACK); //Cálculo el Checksum

//A PARTIR DE AQUI SE MANDAN LOS DATOS AL USART
    Usart_Write(0x7E); //Encabezado
    Usart_Write(LMSB); //MSB
    Usart_Write(LLSB); //LSB
    Usart_Write(APIID); //API ID
    Usart_Write(ACK) //ACK
}

```

```

USART_str(g); //Escribo el comando AT
Array2Serial(NUM, NumBytes); //Escribo el argumento
Usart_Write(CHK); //Checksum
}

//_XBEE_Remoto////////////////////////////////////

Void XbeeRF_str(short APIID, short IDF, short ACK, unsigned short ADDR64[], char *g, unsigned short
NUM[], short NumBytes){

LARGO = strlen(g) + 3 + 8 + NumBytes; //LEN,APIID,RSPFRAME(IDF),ACK,ADDR64 y NumBytes
If(LARGO>0x00FF){
LMSB = LARGO-0x00FF
}
else{
LMSB = 0x00;
}
LLSB = LARGO;
switch(APIID){ //API ID:
case1: //1: 64 bits
APID = 0x00; //2: 16 bits
break; //Default: 64 bits
case 2:
APIID = 0x01;
default:
APIID = 0x00;
}

switch(IDF){ //ID FRAME:
case 0: //0: DISABLE
IDF = 0x00, //2: 16 bits
break;
case 2:
IDF = 0x01;
default:
IDF = IDF;
}

switch(ACK){ //ACK:
case 1: //0: DISABLE
IDF = 0x01; //4: Send Package with broadcast PanID

break; //(Otros bits ==0)
case 4:
IDF = 0x04;
default
ACK = ACK;
}

CHK=0;
tempw=g; //Variable para REW
while(*g){
CHK +=*g++;
}
g=tempw; //REW

for(i=0;i<NumBytes;i++){
CHK += NUM[i];
}
for(i=0;i<8;i++){
CHK +=ADDR64[i];
}
}

```

```

}

CHK = 0xFF-(CHK+APIID+ACK+IDF);    //Cálculo el Checksum

//A PARTIR DE AQUI SE MANDAN LOS DATOS AL USART
Usart_Write(0x7E);                //Encabezado
Usart_Write(LMSB);                //MSB
Usart_Write(LLSB);                //LSB
Usart_Write(APIID);               //API ID
Usart_Write(IDF);                 //IDF
Array2Serial(ADDR64, 8); //Escribo el destino del paquete (SHySL).
Usart_Write(ACK);                 //ACK
USART_str(g);                     //Escribe la palabra
Array2Serial(NUM, NumBytes);      //Escribe los datos numéricos
Usart_Write(CHK);                 //Checksum

//_____MAIN

void main(){
setup();
TRISD = 0xFF;
TRISB = 0x00;
TRISE = 0x00;
//_____C. INICIALES

PORTE = 0;
PORTB = 0;
alarma=1;                        //SE ENCIENDE POR DEFECTO
Usart_init(9600);                //SerialCom a 9600
while(1){
    if(PORTB){
        Usart_Write(alarma);
        delay_ms(10000);
        if(alarma==1){
            while(i<2000){
                PORTE=0xFF;
                delay_ms(1);
                PORTE=0x00;
                delay_ms(1);

                i++;
            }
            else{
                alarma=0;
            }
        }
    }
}
}

```

**APÉNDICE E: FUNCIONES EN CÓDIGO PHP PARA GESTIÓN DE
INFORMACIÓN API XBEE**

```

Void XBee_str(short APIID, short ACK, char *g, unsigned short NUM, int NumBytes){

LARGO = strlen(g) + 2 + NumBytes; //largo + API ID + CHK + NumBytes
If(LARGO > 0x00FF){
LMSB = LARGO-0x00FF;
}
else{
LMSB = 0X00;
}
LLSB = LARGO;
switch(APIID){          //API ID:
case 1:                 //1: AT Command
APIID = 0x08;          //2: AT Queue Parameter Value
break;
case 2:
APIID = 0x09;
default:
APIID = 0x09;
}

switch(ACK)             //ACK:
case 0:                 //0: DISABLE
ACK = 0x00;            //1: ENABLE
break;                  //Cuando APIID = 2, ACK = FRAMEID
case 1:
ACK = 0x01;
break;
default:
ACK = ACK;
}

CHK=0;
tempw=g;                //Variable para REW
while(*g){
CHK +=*g++;
}
g=tempw;                //REW

for(i=0;i<NumBytes;i++){
CHK +=NUM [i];
}

CHK = 0xFF-(CHK+APIID+ACK); // Cálculo el Checksum

//A PARTIR DE AQUÍ SE MANDAN LOS DATOS AL USART
Usart_Write(0x7E);      //Encabezado
Usart_Write(LMSB);     //MSB
Usart_Write(LLSB);     //LSB
Usart_Write(APIID);    //API ID
Usart_Write(ACK);      //ACK
USART_str(g);          //Escribo el comando AT
Array2Serial(NUM, NumBytes); //Escribo el argumento
Usart_Write(CHK);      //Checksum
}

```

```

Void XbeeRF_str(short APIID, short IDF, short ACK, unsigned short ADDR64[], char *g, unsigned short
NUM[], short NumBytes){

```

```

    LARGO = strlen(g) + 3 + 8 + NumBytes;
    //LEN, APID,RSPFRAME(IDF),ACK,ADDR64 y NumBytes
    If(LARGO>0x00FF){
    LMSB = LARGO-0x00FF
    }
    else{
    LMSB = 0x00;
    }
    LLSB = LARGO;

    switch(APIID){
    case1: //API ID:
            //1: 64 bits
            APID = 0x00; //2: 16 bits
            break; //Default:64 bits
    case 2:
            APIID = 0x01;
    default:
            APIID = 0x00;
    }

    switch(IDF){
    case 0: //ID FRAME:
            IDF = 0x00, //0: DISABLE
            break; //2: 16 bits
    case 2:
            IDF = 0x01;
    default:
            IDF = IDF;
    }

    switch(ACK){
    case 1: //ACK:
            IDF = 0x01; //0: DISABLE
            break; //4: Send Package with broadcast PanID
            //(Otros bits ==0)
    case 4:
            IDF = 0x04;
    default:
            ACK = ACK;
    }

    }

    CHK=0;
    tempw=g; //Variable para REW
    while(*g){
        CHK +=*g++;
    }
    g=tempw; //REW

    For(i=0;i<NumBytes;i++){
        CHK += NUM[i];
    }
    for(i=0;i<8;i++){
        CHK +=ADDR64[i];
    }

    CHK = 0xFF-(CHK+APIID+ACK+IDF); //Cálculo el Checksum

```

```

//A PARTIR DE AQUI SE MANDAN LOS DATOS AL USART
Usart_Write(0x7E);           //Encabezado
Usart_Write(LMSB);          //MSB
Usart_Write(LLSB);          //LSB
Usart_Write(APIID);         //API ID
Usart_Write(IDF);           //IDF
    Array2Serial(ADDR64, 8); //Escribo el destino del paquete (SHySL).
Usart_Write(ACK);           //ACK
USART_str(g);               //Escribe la palabra
Array2Serial(NUM, NumBytes); //Escribe los datos numéricos
Usart_Write(CHK);           //Checksum

}

```

```

Function constructApi($datos, $addr, $ack){

    $msb = '00';
    $lsb = dechex ((strlen($datos)/2) + (strlen($addr)/2) + 3);
    $lsb = minLenByte($lsb);
    $ack = ($ack==0)?'00': '01';

    $inicio = '7E';
    $largo = $msb.$lsb;
    $apiType = '00';
    $idf = '01';
    $addr = $addr;
    $ack = $ack;
    $datos = $datos;

    $word = $inicio.$largo.$apiType.
    $idf.$addr.$ack.$datos;
    $temp = apiWord($word);

    for($i=3;$i<count($temp);
    $i++){
        $byte = $temp [$i];
        $byte =
Hexdec ($byte);
        $chksum += $byte;
    }
    $chksum = 255-$chksum;

    $chksum = dechex($chksum);
    $chksum = substr($chksum, -2);
    $chksum =
minLenByte($chksum);

    $word = $word.$chksum;
}

```

```

return $word
}
.
.
//ENVIA BYTE A BYTE POR EL PUERTO SERIE
Function txApi($datos, $addr, $ack){
    $apiWord=constructApi($datos, $addr, $ack);

    $apiWord = apiWord($apiWord);
    for($i=0; $i<count($apiWord); $i+
+){
        $byte = $apiWord [$i];
        $byte = hexdec($byte);
        Ser_writebyte($byte);
    }
}

```


**APÉNDICE F: CD DE MÁQUINA VIRTUAL CON DAMN SMALL LINUX,
SERVIDOR LAMP Y SISTEMA MINICAT**