



**ESCUELA SUPERIOR POLITÉCNICA DE  
CHIMBORAZO**

**FACULTAD DE INFORMÁTICA Y ELECTRÓNICA**

**ESCUELA DE INGENIERÍA ELECTRÓNICA**

**“Estudio de los sistemas Web embebidos y su aplicación en un  
sistema de control domótico con microcontroladores”**

**TESIS DE GRADO**

***Previa la obtención del título de Ingeniera Electrónica***

***Irma Paulina Robalino Layedra***

***Riobamba-Ecuador***

***2009***

## **AGRADECIMIENTO**

*Quiero agradecer en primer lugar a Dios, dador de la vida, por permitirme alcanzar uno de mis más grandes sueños, gracias a mis padres y a mi familia en general por todo su apoyo y cariño incondicional. Gracias a los señores docentes, por compartir sus conocimientos y permitir que pueda forjarme como profesional.*

## **DEDICATORIA**

*Dedico este triunfo a mis padres Irma y Eduardo por ser el instrumento de Dios para darme la vida, por todo el amor, la confianza y el invaluable apoyo que han sabido brindarme en cada etapa. A mi hermano Carlitos, por ser el compañero de risas y llanto y porque su presencia me inspira cada día a esforzarme para ser mejor. A mis abuelitas, que aún tengo a mi lado, por sus consejos sabios y su especial amor. A mis abuelitos que aunque no están presentes, me dejaron enseñanzas que han marcado mi vida. A mis tíos, por su ayuda cuando lo he necesitado, a mis primos y primas por todos esos momentos compartidos que son inolvidables, y a todos quienes en su momento aportaron con valiosos granitos de arena para ayudarme a crecer.*

*“Yo, Irma Paulina Robalino Layedra, soy responsable de las ideas, doctrinas y resultados expuestos en esta Tesis, y el patrimonio intelectual de la misma pertenecen a la ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO”.*

---

*IRMA PAULINA ROBALINO LAYEDRA*

# INDICE

<b>CAPITULO I .....</b>	<b>5</b>
<b>INTRODUCCION.....</b>	<b>8</b>
1.1 Antecedentes.....	8
1.2 Objetivos .....	11
1.2.1 General.....	11
1.2.2 Específicos .....	11
1.3 Justificación.....	12
<b>CAPITULO II .....</b>	<b>15</b>
ESTUDIO DE LA DOMÓTICA .....	15
2.1 INTRODUCCIÓN A LA DOMÓTICA .....	15
2.2 CARACTERISTICAS DE UN SISTEMA DOMOTICO.....	20
2.3 FUNCIONES DEL SISTEMA DOMÓTICO .....	23
2.4 DESCRIPCIÓN DE LOS SISTEMAS DOMOTICOS.....	26
2.5 VENTAJAS Y DESVENTAJAS DE LA DOMOTICA.....	35
<b>CAPÍTULO III .....</b>	<b>38</b>
ESTUDIO DE LOS SERVIDORES WEB EMBEBIDOS Y MICROCONTROLADORES. ..	38
A. Componentes de un Sistema Embebido .....	54
B. Aplicaciones de un sistema embebido .....	55
C. Ventajas de un sistema embebido sobre las soluciones industriales tradicionales .....	57

<b>CAPITULO IV</b> .....	<b>58</b>
BASIC STAMP 2.....	58
4.1 INTRODUCCION .....	58
4.2 FUNCIONAMIENTO INTERNO DEL BASIC STAMP 2.....	62
4.3 ORGANIZACIÓN DE MEMORIA DEL BS2.....	69
4.4 ESTRUCTURA DE PROGRAMACIÓN EN BS2.....	81
<b>CAPITULO V</b> .....	<b>89</b>
Parallax Internet Netburner Kit.....	89
<b>CAPITULO VI</b> .....	<b>98</b>
DESARROLLO DEL PROYECTO.....	98
6.1 FASE DE ANALISIS .....	98
6.2 FASE DE DISEÑO .....	100
6.3 FASE DE IMPLEMENTACION.....	133
6.3.1 CONEXIÓN TÍPICA PARA EL FUNCIONAMIENTO DEL BS2.....	133
6.3.2 MODO DE PROGRAMACIÓN DEL BASIC STAMP 2.....	133
6.3.4 MODULO PINK.....	140
<b>CAPITULO VII</b> .....	<b>143</b>
PRUEBAS Y RESULTADOS .....	143
ANALISIS DE RESULTADOS.....	156
<b>CONCLUSIONES</b> .....	<b>158</b>
<b>RECOMENDACIONES</b> .....	<b>159</b>

<b>RESUMEN.....</b>	<b>160</b>
<b>SUMMARY .....</b>	<b>161</b>
<b>GLOSARIO .....</b>	<b>162</b>
<b>ANEXOS .....</b>	<b>165</b>
ANEXO 1 .....	166
ANEXO 2 .....	167
ANEXO 3 .....	169
ANEXO 4 .....	170
ANEXO 5 .....	1
<b>TABLA COMPARATIVA DE ALTERNATIVAS DE MICROCONTROLADORES.....</b>	<b>1</b>
<b>BIBLIOGRAFIA.....</b>	<b>1</b>

# CAPITULO I

## INTRODUCCION

### 1.1 Antecedentes

Vivimos en un entorno sensorial del que podemos recabar datos e información valiosa. Desde siempre la humanidad ha buscado formas creativas de simplificar su vida y en el transcurrir del tiempo los logros tecnológicos han permitido que muchos de estos sueños hoy sean una realidad. Con el paso de los años nuevas tecnologías han sido diseñadas para permitir un control óptimo de diferentes aspectos en nuestra vida cotidiana, es así que hoy en día disponemos de herramientas como la domótica, los microcontroladores y los sistemas Web para satisfacer varias de nuestras necesidades.

El objetivo del ser humano de poder mantener control sobre su vivienda tanto en el aspecto de la seguridad como en el de manejo inteligente de sus espacios hoy en día es una realidad.

Es así que se puede señalar que la domótica es una serie de sistemas tecnológicos que aportan diferentes servicios al hogar, estos servicios pueden ser de seguridad, bienestar, comunicación, de gestión energética. etc. La domótica está integrada por redes de comunicación tanto interiores como exteriores ya sea de forma inalámbrica o alambrada. Esta no solo va dirigida a las viviendas, sino también a los comercios, edificios, granjas. etc.

De una manera general, un sistema domótico dispone de una red de comunicación y diálogo que permite la interconexión de una serie de equipos a fin de obtener información sobre el entorno doméstico y, basándose en ésta, realizar determinadas acciones sobre dicho entorno.

Los elementos de campo (detectores, sensores, captadores, etc.), transmiten las señales a una unidad central inteligente que tratará y elaborará la información recibida. En función de dicha información y de una determinada programación, la unidad central actúa sobre determinados circuitos de potencia relacionados con las señales recogidas por los elementos de campo correspondientes.

La domótica se ha implantado desde hace décadas, pero, desde que se creó el Internet este ha tomado un giro controversial, los modelos tecnológicos relacionados a este han progresado y forman parte del futuro de la domótica. Las tecnologías inalámbricas WiFi y las redes de Internet, se han constituido como las tecnologías del entorno digital que evolucionarán, y sobre las cuales la domótica debería mantenerse para poder aumentar el uso de las tecnologías en los hogares.

Por otro lado un sistema integrado (a veces traducido del inglés como embebido, empotrado o incrustado) es un sistema informático de uso específico construido dentro de un dispositivo mayor. En un sistema integrado la mayoría de los componentes se encuentran incluidos en la placa base.

Un sistema embebido es un ordenador más, que a diferencia de un ordenador personal (PC) carece de teclado y pantalla en la mayoría de los casos. Dicho de otra manera, un sistema embebido consiste de una electrónica programable especialmente diseñada para soluciones específicas. Estas soluciones pueden ser:

- Sistemas de telefonía fija o móvil, por ejemplo un Terminal telefónico móvil.
- Automatización de procesos de producción.
- Equipos de instrumentación industrial.
- Sistema de transporte, desde cintas transportadoras y sistemas robotizados, hasta vehículos de transporte de todo tipo.

- Control de electrodomésticos de todo tipo.

El sistema embebido está especialmente diseñado para la solución óptima de tareas o tareas a resolver. Generalmente es una “pieza especializada” instalada en un sistema anfitrión. A diferencia de un PC, el sistema embebido se dota con los módulos estrictamente necesarios para su función, de ahí su costo óptimo.

La incorporación de un sistema embebido proporciona un valor añadido a una aplicación, esto es posible gracias a que el sistema proporciona una solución más precisa y rápida en su especialidad.

Es así que se puede citar que un servidor Web Embebido (incrustado en el circuito del módulo) es capaz de alojar páginas Web, desde las cuales poder mostrar datos que pueda gestionar un módulo de control, gracias al soporte de variables y registros especiales que permitirán además interactuar con dichos datos.

Además permite el control de procesos on line a través de la red. Se puede integrar de forma sencilla en muchos proyectos y aplicaciones que requieran un control remoto.

Con la irrupción de Internet en el hogar y, en general, las denominadas TIC (Tecnologías de Información y las Comunicaciones), se ha forjado una nueva forma de entender la aplicación de tecnología en la vivienda. Con ello, la tecnología es algo transparente para el usuario, el cual no tiene interés técnico sino simplemente de utilidad. El usuario no está interesado en la tecnología sino en resolver su problema. Por ello, se considera que el paso decisivo para implementar un sistema domótico en una vivienda debe ser la utilización de un servidor Web además de microcontroladores para un correcto manejo de la información.

## **1.2 Objetivos**

### **1.2.1 General**

Realizar un estudio de los sistemas Web embebidos y microcontroladores y a partir de este implementar un sistema de control de seguridad que permita monitorizar una vivienda a través de Internet permitiendo al usuario estar informado en todo momento de los eventos que sucedan en su hogar y poder actuar en consecuencia a los mismos.

### **1.2.2 Específicos**

- Estudiar el funcionamiento de los servidores Web embebidos y su utilidad para un sistema de control domótico.
- Investigar la familia de microcontroladores para determinar la mejor alternativa para establecer un óptimo intercambio de datos.
- Establecer la relación que puede existir entre los microcontroladores, el Internet y las herramientas de control domótico.
- Diseñar un sistema de control domótico que permita una interacción bidireccional entre el usuario y su hogar desde cualquier lugar.
- Implementar un sistema de monitoreo en una vivienda que permita sensor una zona determinada detectar la presencia de intrusos.

## 1.3 Justificación

La domótica se encarga de gestionar principalmente los siguientes cuatro aspectos del hogar:

- **Energía eléctrica:** En este campo, la domótica se encarga de gestionar el consumo de energía, mediante temporizadores, relojes programadores, termostatos, etc.
- **Confort:** La domótica proporciona una serie de comodidades, como pueden ser el control automático de los servicios de:
  - Calefacción
  - Agua caliente
  - Refrigeración
  - Iluminación

y la gestión de elementos como accesos, persianas, toldos, ventanas, riego automático, etc.

- **Seguridad:** La seguridad que proporciona un sistema domótico es más amplia que la que nos puede proporcionar cualquier otro sistema, pues integra tres campos de la seguridad que normalmente están controlados por sistemas distintos:
  - Seguridad de los bienes: Gestión del control de acceso y control de presencia, así como la simulación de presencia.
  - Seguridad de las personas: Especialmente, para las personas mayores y los enfermos. Mediante el nodo telefónico, se puede tener acceso

(mediante un pulsador radiofrecuencia que se lleve encima, por ejemplo) a los servicios de Policía, etc.

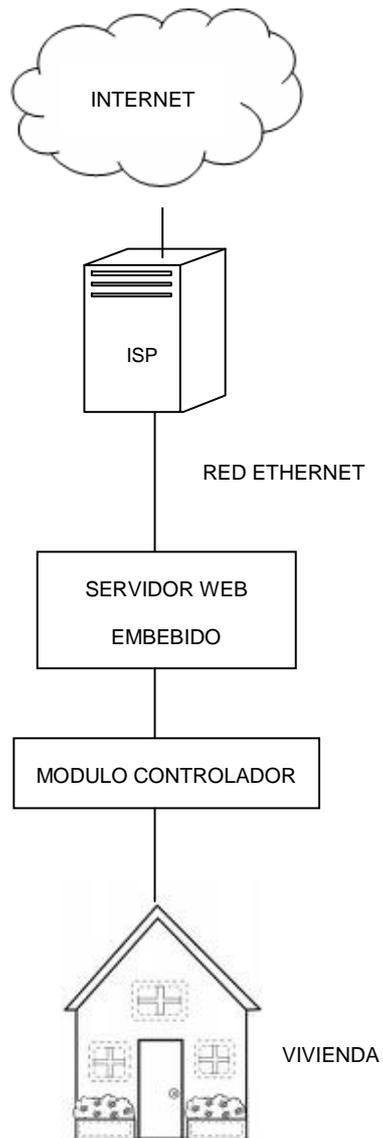
- Incidentes y averías: Mediante sensores, se pueden detectar los incendios y las fugas de gas y agua, y, mediante el nodo telefónico, desviar la alarma hacia los bomberos, por ejemplo.
  - También se pueden detectar averías en los accesos, en los ascensores, etc.
- **Comunicaciones:** La domótica tiene una característica fundamental, que es la integración de sistemas, por eso hay nodos que interconectan la red domótica con diferentes dispositivos, como la red telefónica, el video portero, etc.

Como información adicional se puede citar que un servidor Web permite, entre otras cosas, conectar un microcontrolador a una red Ethernet, como Internet, e intercambiar datos. Este servidor Web integrado permite el control de un conjunto de dispositivos desde una red de área local o Internet.

El sistema que se propone desarrollar, se creará en primer lugar con la finalidad de ser una herramienta de seguridad. El objetivo principal de esta herramienta es el de desarrollar un sistema que dé al usuario la posibilidad de estar informado en todo momento de lo que sucede en su vivienda y poder reprogramar la configuración o actuar directamente sobre los dispositivos de una manera rápida y sencilla.

Con la información antes mencionada, se puede decir que nos encontramos en un momento en el que las tecnologías evolucionan a un ritmo excesivamente rápido y los modelos de aplicaciones deben cambiar al mismo ritmo, es por eso que se plantea el

desarrollo de un sistema de control domótico para viviendas que estará controlado vía Internet (Figura 1).



**Figura 1: Diagrama del sistema de control domótico**

## **CAPITULO II**

### **ESTUDIO DE LA DOMÓTICA**

#### **2.1 INTRODUCCIÓN A LA DOMÓTICA**

El concepto de automatización tiene muchos años de existencia como tal, desde que a un estudiante se le ocurrió conectar dos cables eléctricos a las manecillas de un reloj despertador, para que poco tiempo después, y movidos por dichas manecillas, los cables cerraran un circuito formado por una pila y una lámpara. Ese pudo ser el momento en que nació la idea de temporizar una función eléctrica. Más adelante se fueron perfeccionando los sistemas, primitivos al principio y mucho más sofisticados más tarde, hasta llegar al momento actual donde fundamentalmente las industrias basan gran cantidad de fases de producción en distintos tipos de elementos automáticos o temporizados, desde el sonido de la sirena de entrada de los trabajadores, hasta el precalentamiento de hornos para que cuando lleguen los distintos operarios encuentren sus puestos de trabajo en condiciones óptimas.

La idea de la moderna automatización del hogar para proporcionar a los usuarios mayor comodidad, ahorro de energía y, por supuesto, dinero, tiene pocos años, y fue desarrollada y patentada por una empresa escocesa utilizando un novedoso sistema de transmisión de señales a través de la red eléctrica. Más tarde se fue perfeccionando dicha idea y se utilizaron una serie de emisores que se enchufaban en una parte de la red eléctrica y que eran capaces de emitir una señal que circulaba a través de ella. A su vez, otra serie de receptores, que igualmente iban enchufados en otra parte de la red, eran los encargados de recibir dicha señal y de transformarla en una acción, por ejemplo activar un relé o contacto eléctrico.

Los avances tecnológicos que se vienen realizando en las áreas de telecomunicación, informática y electrónica han propiciado el desarrollo de productos y sistemas para el control y la supervisión de los equipamientos de viviendas y edificios, permitiendo a su vez una mayor comunicación bidireccional entre los usuarios y sus viviendas o edificios.

Hasta hace poco el usuario de una instalación eléctrica podía contentarse con iluminarse, calentarse y muy poco más. El trabajo del electricista consistía en proteger las líneas y las personas contra riesgos eléctricos.

Después han aparecido productos y funciones que nos proporcionan confort, los fabricantes de material eléctrico han desarrollado una serie de productos que nos permiten programar la calefacción, regular la temperatura ambiental, TV por cable, satélite, Internet.

La instalación se ha vuelto más compleja, los cables se suman a los cables, una modificación en la utilización de los locales se traduce en largas y costosas operaciones.

Es así que la tecnología ha avanzado tan rápidamente que hoy en día se tiene a disposición la posibilidad de automatizar una vivienda de tal forma que se obtenga mayores beneficios.

Las posibilidades que ofrece la domótica e una vivienda son innumerables, estando su limitación en las necesidades e imaginación de cada usuario.

El término Domótica viene del latín "domus" que significa casa y de la palabra "automática", por lo tanto la Domótica se refiere a una casa automática o como se le ha llamado más comúnmente una casa inteligente. En inglés a la Domótica se la conoce más como "home networking" o "smart home". Una casa inteligente es aquella cuyos elementos o dispositivos están integrados y automatizados a través de una red (principalmente Internet) y que a través de otro dispositivo remoto o inclusive interno se puede modificar

sus estados o los mismos dispositivos están diseñados para realizar ciertas acciones cuando han detectado cambios en su propio ambiente.

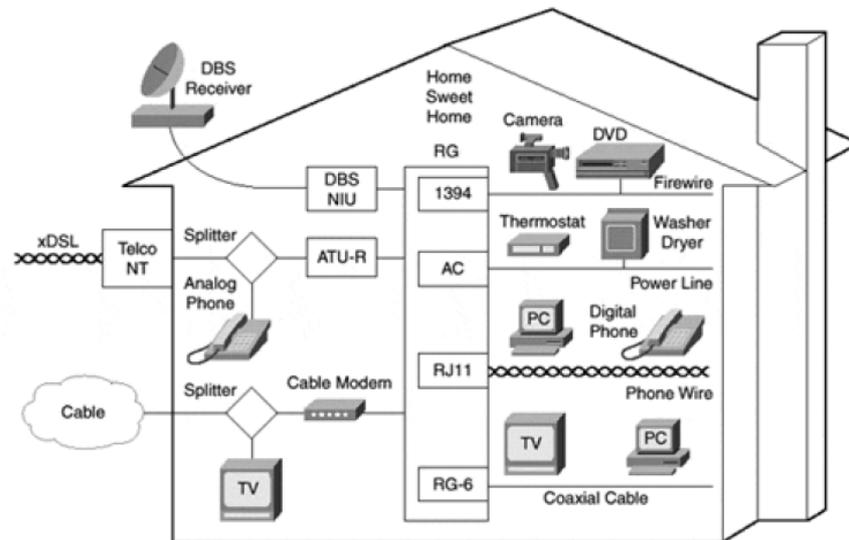
La enciclopedia Larousse definía en 1988 el término Domótica como: “el concepto de vivienda que integra todos los automatismos en materia de seguridad, gestión de la energía, comunicaciones, etc.”. Una definición más técnica del concepto sería: “conjunto de servicios de la vivienda garantizado por sistemas que realizan varias funciones, los cuales pueden estar conectados entre sí y a redes interiores y exteriores de comunicación. Gracias a ello se obtiene un notable ahorro de energía, una eficaz gestión técnica de la vivienda, una buena comunicación con el exterior y un alto nivel de seguridad”.

Para que un sistema pueda ser considerado “inteligente” ha de incorporar elementos o sistemas basados en las Nuevas Tecnologías de la Información (NTI) que van surgiendo día a día. El uso de las NTI en una casa genera nuevas aplicaciones y tendencias basadas en la capacidad de proceso de información y en la integración y comunicación entre los equipos e instalaciones.

La definición de casa domótica o inteligente presenta múltiples versiones en diferentes países e idiomas, pero los términos más utilizados son: “casa inteligente” (smart house), automatización de viviendas (home automation), domótica (domotique), sistemas domésticos (home system), entre otros.

De manera general, un sistema domótico dispondrá de una red de comunicación y diálogo, que permite la interconexión de una serie de equipos a fin de obtener información sobre el entorno doméstico y, basándose en ésta, realizar unas determinadas acciones sobre dicho entorno. El funcionamiento de una casa inteligente consistirá a grandes rasgos de lo siguiente: los elementos de campo (detectores, sensores, captadores, etc.),

transmitirán señales a una unidad central inteligente que tratará y elaborará la información recibida. En función de dicha información y de una determinada programación, la unidad central actuará sobre determinados circuitos de potencia relacionados con las señales recogidas por los elementos de campo correspondientes.



**Figura 2: Esquema de una casa conectada en red.**

La figura 2 muestra el esquema de un sistema domótico o en este caso de una casa. A continuación se detallan las diferentes definiciones que ha ido tomando el término Domótica en los últimos años:

1. La nueva tecnología de los automatismos de maniobra, gestión y control de los diversos aparatos de una vivienda, que permiten aumentar el confort del usuario, su seguridad y el ahorro en el consumo energético.
2. Un conjunto de servicios en las viviendas, asegurados por sistemas que realizan varias funciones, pudiendo estar conectados, entre ellos, y a redes internas y externas de comunicación.

3. La informática aplicada a la vivienda. Agrupa el conjunto de sistemas de seguridad y la regulación de las tareas domésticas designadas a facilitar la vida cotidiana automatizando sus operaciones y funciones.

Una característica de las casas inteligentes es que deben tener la flexibilidad para asumir modificaciones de manera conveniente y económica.

Desde el punto de vista computacional, una Casa Inteligente sugiere la presencia de sistemas basados en técnicas de inteligencia artificial, programados, sistemas distribuidos, capaces de:

- Tomar las decisiones necesarias en un caso de emergencia.
- Predecir y auto diagnosticar las fallas que ocurran dentro de la casa.
- Tomar las acciones adecuadas para resolver dichas fallas en el momento adecuado.
- Monitorear y controlar las actividades y el funcionamiento de las instalaciones de la casa.

Para que esto sea entendible la figura 3 muestra las conexiones dentro de una casa para conocer sus ubicaciones y la forma de comunicación.



**Figura 3: Vista de conexión para domótica ejemplificada en una casa**

En la actualidad, existen multitud de sistemas diferentes de transmisión de señales vía red eléctrica (incluso internet), y por lo tanto hay gran cantidad de empresas dedicadas a esta actividad, tanto en el ámbito industrial como en el doméstico.

A muchos siempre les ha gustado ver cómo en las películas el protagonista, al llegar a su lujosa mansión, y con el simple gesto de apretar un botón, se encendían las luces al nivel adecuado, se activaba el equipo de música, etc. Todo esto no es más que AUTOMATIZACION DEL HOGAR.

El avance tecnológico en los procesos de fabricación y la entrada en el mercado de mejores, más rápidos, eficientes, económicos y sobre todo más pequeños componentes electrónicos, ha permitido que dichos elementos (emisores y receptores), al ser mucho más livianos, discretos y atractivos, tengan gran demanda por parte del público. De hecho, fue desde el momento de la miniaturización de dichos elementos cuando se ha experimentado un importante aumento de consumo del concepto de automatización doméstica.

Desde hace algunos años se habla de la "vivienda inteligente", asociándose a pisos de gran lujo. Actualmente hay soluciones económicas al alcance de cualquiera, tomando como base un autómata programable y un PC como complemento. Con este equipamiento se pueden automatizar todo tipo de viviendas y locales.

## **2.2 CARACTERISTICAS DE UN SISTEMA DOMOTICO**

De manera general, un sistema domótico dispondrá de una red de comunicación y diálogo que permita la interconexión de una serie de equipos a fin de obtener información sobre el

entorno doméstico y, basándose en ésta, realizar unas determinadas acciones sobre dicho entorno. Por ello las características fundamentales de este sistema son:

- Control remoto desde dentro de la vivienda: Tal posibilidad se consigue mediante un esquema de comunicación entre los distintos equipos que forman parte del hogar.

La consecuencia más inmediata de lo anterior consiste en una reducción considerable de la necesidad de moverse dentro de la vivienda, lo que puede resultar particularmente importante en el caso de personas de la tercera edad o minusválidos.

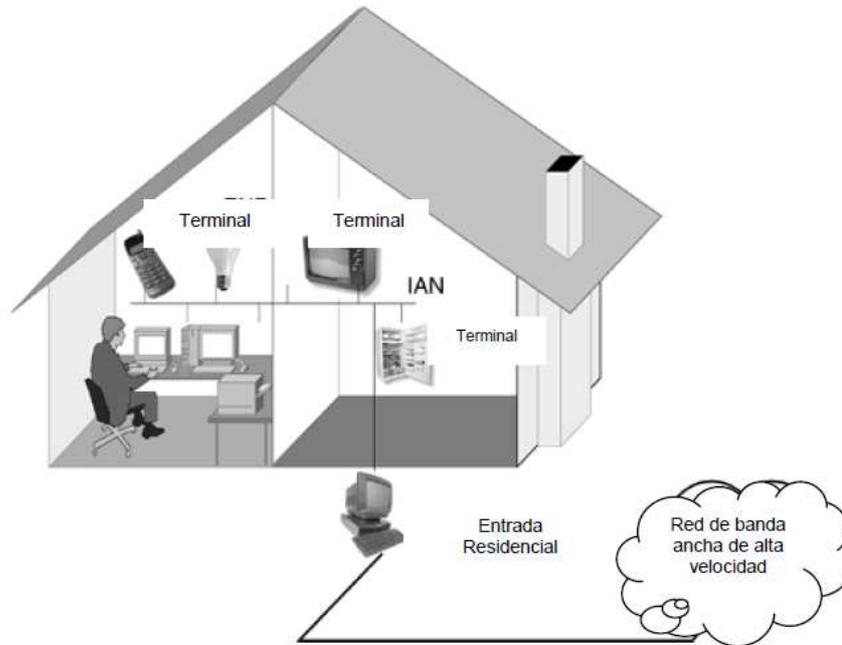
- Control remoto desde fuera de la vivienda: Presupone un cambio en los horarios en los que se realizan las tareas domésticas y permite al usuario un mejor aprovechamiento de su tiempo. En la actualidad ya existen numerosos electrodomésticos que incorporan esta faceta; por ejemplo la lavadora Margherita 2000 de la firma Ariston Digital que se puede manejar a través de Internet.
- Programabilidad: Permite que los sistemas se puedan programar ya sea para que realicen ciertas funciones con sólo tocar un botón o bien que las lleven a cabo en función de otras condiciones del entorno (hora, temperatura interior o exterior, etc.)
- Acceso a servicios externos: Aquí podemos incluir servicios como el telebanco o la telecompra y por supuesto el teletrabajo, siendo que este último ha sido uno de los principales motivos de la aplicación de las nuevas tecnologías a la vivienda.

Además de estas características, se pueden mencionar las que siguen:

- INTEGRACION: Todo sistema funciona bajo el control de una computadora. De esta manera, los usuarios no tienen que estar pendientes de los diversos equipos autónomos, con su propia programación, indicadores situados en diferentes lugares, dificultades de interconexión entre equipos de distintos fabricantes, etc.

- **INTERRELACION:** Una de las principales características que debe ofrecer un sistema domótico es la capacidad para relacionar diferentes elementos y obtener una gran versatilidad y variedad en la toma de decisiones. Así, por ejemplo, es sencillo relacionar el funcionamiento del aire acondicionado con el de otros electrodomésticos, con la apertura de ventanas, o con que la vivienda esté ocupada o vacía, etc.
- **FACILIDAD DE USO:** Con una sola mirada a la pantalla de la computadora, el usuario está completamente informado del estado de su casa. Y si desea modificar algo, solo necesitará pulsar un reducido número de teclas. Así, por ejemplo, la simple observación de la pantalla nos dirá si tenemos correo pendiente de recoger en el buzón, las temperaturas dentro y fuera de la vivienda, si está conectado el aire acondicionado, cuando se ha regado el jardín por última vez, si la tierra está húmeda, si hay alguien en las proximidades de la vivienda, etc.
- **CONTROL REMOTO:** Las mismas posibilidades de supervisión y control disponibles localmente (excepto sonido y música ambiental) pueden obtenerse mediante conexión telefónica desde otra computadora en cualquier lugar del mundo. De gran utilidad será en el caso de personas que viajan frecuentemente, o cuando se trate de residencias de fin de semana, etc.
- **FIABILIDAD:** Las computadoras actuales son máquinas muy potentes, rápidas y fiables. Si añadimos la utilización de un Sistema de Alimentación Ininterrumpida, ventilación forzada de CPU, batería de gran capacidad que alimente periféricos, apagado automático de pantalla, etc. Se debe disponer de una plataforma ideal para aplicaciones domóticas capaces de funcionar muchos años sin problemas.
- **ACTUALIZACION:** La puesta al día del sistema es muy sencilla. Al aparecer nuevas versiones y mejoras sólo es preciso cargar el nuevo programa en su

equipo. Toda la lógica de funcionamiento se encuentra en el software y no en los equipos instalados. De este modo, cualquier instalación existente puede beneficiarse de las nuevas versiones, sin ningún tipo de modificación.



*Figura 4: Componentes de la infraestructura en un hogar doméstico*

## 2.3 FUNCIONES DEL SISTEMA DOMÓTICO

### A. Gestión de la confortabilidad:

Esta gestión se consigue mediante el control de la iluminación y la temperatura. Algunas aplicaciones serían:

- Accionamiento automático de ventanas, persianas y toldos (con sensores crepusculares y de viento).

- Recepción de mensajes de mal funcionamiento de aparatos, del videoportero automático, etc. a la pantalla del televisor, el monitor de un PC u otro monitor de la casa.
- Distribución de señales de audio y video para las habitaciones (desde cualquier habitación se puede controlar el televisor, el video, el equipo de música, etc.).
- Accionamiento automático de electrodomésticos.
- Posibilidad de dar un conjunto de órdenes programadas instantáneamente, por ejemplo, para activar el programa de vacaciones.
- Accionamiento especial de aparatos para personas con minusvalías físicas.
- Centralización de la información sobre el estado del edificio en paneles o monitores.
- Activación del riego automático en función de la humedad, la lluvia y el viento.
- Unidades de gobierno activadas por voz.

A parte de los sensores, presentes en cualquier operación automatizada, existen elementos orientados a mejorar el confort llamados comandamientos remotos y temporizadores. El comandamiento más común es el de infrarrojos, con una capacidad muy limitada (5-20m). Con más capacidad están los comandamientos por radiofrecuencia. Las nuevas tecnologías de la información permiten la posibilidad del control remoto a través de módem telefónico, que se puede dirigir con una llamada o desde internet.

La incorporación de temporizadores horarios permite la generación de órdenes de actuación a receptores como puntos de luz, calefacción, riego, simulación de presencia, etc.

## **B. Gestión de la seguridad:**

Esta gestión se consigue mediante distintos sistemas:

- Sistemas de alarmas técnicas que se disparan cuando un parámetro sale de un valor determinado. Sirven para detectar incendios, inundaciones, fugas de gas, etc. En caso de un escape el sistema provoca el cerramiento de las válvulas de paso.
- Sistemas antirrobo orientado no solo a impedir que entren personas al edificio o a la vivienda, sino a disuadirlos de sus intentos. Mediante la colocación de detectores de presencia se pueden detectar movimientos y también el calor de los cuerpos. La simulación de presencia consiste en el alumbrado y apagado de las luces aleatoriamente y temporizada para dar la impresión de que hay gente en el edificio.
- Control de accesos que restringen la entrada y salida de personas en un edificio, permiten su identificación o bien indican la existencia de objetos no permitidos con detectores de metales y barreras de infrarrojos.
- Alarmas médicas que permiten realizar una consulta médica desde casa, vía módem o en forma de colgantes o llaveros con pulsadores que permiten el aviso a familiares o a los teléfonos adecuados.

### **C. Gestión de la energía:**

Esta gestión se consigue mediante:

- Regulación para mantener una magnitud dentro de valores fijados.
- Programación de los valores en función del tiempo.
- Optimización realizando un diseño a partir de todas las variables y condiciones para asegurar el costo mínimo del usuario.
- Desconexión para detener un aparato en el caso de que su funcionamiento resulte un costo superior al adecuado.

- Seguridad, el sistema interviene en caso de peligro de sobrecarga de equipos.

## **2.4 DESCRIPCIÓN DE LOS SISTEMAS DOMOTICOS**

### **• ILUMINACIÓN**

La forma de encender y apagar la iluminación de la vivienda puede ser automatizada y controlada de formas complementarias al control tradicional a través del interruptor clásico. Se puede en esta manera conseguir un incremento del confort y ahorro energético. A continuación se indican las principales funciones de control domótico de la iluminación.

La iluminación puede ser regulada en función del nivel de luminosidad del ambiente, evitando su encendido innecesario o adaptándola a las necesidades del usuario. La activación de ésta se realiza siempre y cuando el nivel de luminosidad pasa un determinado umbral, ajustable por parte del usuario. Esto garantiza un nivel de iluminación mínima, que puede ser esencialmente útil para por ejemplo un pasillo o la iluminación exterior.

La iluminación puede ser activada en función de la presencia de personas en la estancia. Se activa la iluminación cuando un sensor detecta presencia. Esto garantiza una buena iluminación para por ejemplo zonas de paso como pasillos. Asegura que las luces no se queden encendidas en habitaciones cuando no hace falta.

Activación de la iluminación según otros eventos, por ejemplo al pulsar el mando a distancia del garaje la iluminación exterior de acceso y el del garaje se puede encender por un tiempo limitado para poder tener un acceso seguro y confortable. O si salta la

alarma de seguridad en el exterior de la vivienda se puede encender toda la iluminación exterior.

El encendido o apagado de una luminaria puede temporizarse a voluntad del usuario, permitiendo su actuación al cabo de determinado tiempo. Su uso puede ser variado, estando sujeto a las necesidades y deseos del usuario. Por ejemplo que se encienda la luz de forma graduada del dormitorio a cierta hora de la mañana, o que se apague toda la iluminación del jardín a cierta hora por la noche.

La iluminación también puede realizarse a través de mandos a distancia, con independencia del tradicional mecanismo de mando eléctrico. Un mismo mando a distancia puede controlar distintas luminarias a la vez que otras funciones del hogar digital.

Es preciso indicar que un sistema domótico debería garantizar siempre la posibilidad de encender y apagar la iluminación de forma tradicional, es decir, de forma voluntaria y manual mediante interruptores tradicionales por parte del usuario.

## • CLIMATIZACIÓN

La forma más básica de controlar la climatización de una vivienda es la conexión o desconexión de todo el sistema de climatización. Se puede realizar esto según una programación horaria, según presencia de personas en el hogar o de forma manual. Con estos modos de funcionamiento el sistema sólo garantiza el establecimiento de una temperatura de consigna única para toda la vivienda, de forma parecida a la existencia de un termostato de ambiente convencional. Sin embargo se puede hacer muchísimo más para alcanzar un alto nivel de confort y ahorrar energía.

## • ZONIFICACIÓN

Cada zona definida en la vivienda tiene requisitos de uso o condiciones térmicas distintas, que hacen conveniente el ser gestionadas de forma independiente. Esta gestión por zonas puede realizarse siguiendo una misma programación para cada una de ellas, o bien ser controlarlas de forma independiente, incrementando, con ello, las posibilidades de uso y confort para el usuario.

Es importante resaltar que en instalaciones de climatización sin zonificación, algunas estancias de la vivienda pueden climatizarse por exceso como consecuencia de su tamaño, orientación, uso, etc., creando una reducción del confort para el usuario. Así mismo, otras estancias de la vivienda pueden climatizarse por defecto, es decir, sin alcanzar la temperatura deseada, creando una misma situación.

Los criterios seguidos para definir una zonificación de la vivienda pueden ser variados. De entre los posibles, los más habituales son los dos siguientes:

- El uso dado a las dependencias, creando lo que se denomina como zona día (uso habitual durante el día como el comedor, el salón, etc.) y zona noche (habitualmente limitada a las habitaciones); y
- La orientación de la vivienda, considerando los aportes energéticos solares, creando las dos zonas siguientes: la zona norte (estancias no expuestas a la radiación solar) y la zona sur (con incidencia solar).
- Incremento del grado de confort al asegurar la temperatura deseada por el usuario en cada una de las zonas disponibles. Asociadamente, esta aplicación permite también reducir el consumo de energía al incrementar la eficiencia global de la instalación. Sólo se climatizan aquellas zonas de la vivienda que son necesarias.

El número y tipo de niveles de temperatura más comúnmente utilizados son los siguientes:

- Nivel de temperatura de confort. Es el estado habitual de funcionamiento de la climatización, que se da, por lo general, cuando los usuarios se encuentran en la vivienda (por ejemplo, una temperatura de consigna de 21°C para calefacción).
- Nivel de temperatura de economía. Estado de funcionamiento que se da cuando, o bien los usuarios salen de casa por un corto período de tiempo, o bien durante aquellos períodos en los cuales no se requiere un nivel de temperatura tan elevado (si se considera la calefacción) o tan bajo (si se considera el aire acondicionado). Un ejemplo de ello sería el uso de calefacción durante la noche al acostarse, con una temperatura de economía, por ejemplo, 18°C).
- Nivel de temperatura antihelada. Con el objeto de evitar que el agua contenida en las conducciones de agua de la vivienda se hiele en invierno y produzca roturas en las mismas, el sistema de calefacción se puede poner en marcha para alcanzar una temperatura mínima establecida por el sistema (por ejemplo, una temperatura de 5°C).

Los beneficios son el aumento del confort doméstico y optimización del consumo energético al asegurar que solamente se mantiene la temperatura necesaria durante un período concreto. Con el nivel antihelada se evita, además, la rotura de conducciones de agua por el efecto de la temperatura.

## • DEROGACIÓN DE NIVELES DE TEMPERATURA

El sistema domótico gestiona el funcionamiento de la climatización siguiendo el programa introducido en el sistema domótico, es decir, acorde con el perfil de temperatura. Este

seguimiento supone un determinado número de cambios entre los niveles de confort y economía. Sin embargo, el usuario puede modificar en cualquier momento el nivel de temperatura existente (de confort a economía, o viceversa), por diversos motivos, forzando un cambio puntual en el perfil de temperatura. A este cambio puntual se le conoce como derogar el nivel de temperatura existente.

Es preciso indicar que este cambio puntual no afecta al desarrollo del perfil de temperatura ni lo modifica. El sistema domótico seguirá el perfil de temperatura una vez se restablezca el nivel programado.

- **PUERTAS Y VENTANAS**

En el caso de tener Puertas y Ventanas motorizadas estas pueden ser integradas con el sistema de domótica.

Un área de aplicación principal es para gente con discapacidades físicas. La automatización puede ayudar al usuario tanto abrir como cerrar las puertas y ventanas a través del sistema de domótica. Cada puerta o ventana puede ser controlado de forma individual y por zonas.

Además las puertas de acceso a la finca, al garaje, etc. pueden ser abiertos por el sistema integrado de domótica en combinación de otras actuaciones como el encendido de la iluminación o como consecuencia de la desconexión del sistema de seguridad, etc.

Las puertas y ventanas pueden ser controladas para temas climatológicos. Es decir se pueden abrir y cerrar para crear corrientes de aire para la ventilación natural. Las puertas y ventanas también pueden ser programadas para ser controladas de forma automática

para el tema de seguridad si se detecta fuego, humo o gas, es decir cerrarse o abrirse automáticamente.

Pero no solo pueden ser controlados los motores para abrir y cerrar las ventanas y puertas. También pueden ser controladas las cerraduras, con cerraduras electrónicas se puede abrir y cerrar cerraduras de forma local y remota a través del sistema de domótica.

## • PERSIANAS Y TOLDOS

En el caso de tener persianas y toldos motorizados hay varias formas de controlar estos a través de los sistemas de domótica.

Las persianas y toldos pueden ser controlados según la temperatura interior o la situación climatológica del exterior. Es decir si se quiere que entre el sol y la luz para calentar el interior de la vivienda a través de las ventanas, las persianas pueden de forma automática abrirse según una programación horaria o según los datos de sensores de luz. En la misma manera se puede asegurar que estén bajadas para que la luz solar no dañe el interior. También sensores de lluvia y viento pueden obligar a los toldos a recogerse para que no sean dañados.

Las persianas pueden ser controladas de forma automática según una programación horaria o un escenario por el tema del confort y el ahorro energético, para minimizar el uso de la iluminación artificial. Pueden por ejemplo subirse de forma automática por la mañana y bajarse por la noche para temas de confort o por el tema de seguridad. En la misma manera pueden tener una actuación programada para el tema de la seguridad si se detecta por ejemplo humo, fuego, gas o una intrusión.

- **APARATOS**

Hay muchos sistemas y aparatos dentro y en el exterior de la vivienda que pueden ser controlados por el sistema de domótica. Cafeteras, radios, y otros aparatos que se encienden simplemente activando la alimentación suelen ser integrados con facilidad y pueden para muchos usuarios ser prácticos y curiosos. Esto se puede realizar según una programación horaria, para simular presencia, para escenarios o según se disparan otros eventos.

- **RIEGO**

El riego automático es una aplicación muy utilizada por la gente que vive en viviendas unifamiliares. Este puede ser gestionado por un controlador que normalmente se limita a regar según la programación horaria. Pero el riego puede ser más desarrollado y avanzado que eso. Puede ser activado de forma automática según programación horaria, pero también según la humedad en el césped, el día de la semana o cualquier otro valor. Además si el riego está integrado en el sistema de domótica tele controlada, se puede controlar el mismo de forma remota o según otros eventos como incendios o robos. Además existe la posibilidad de realizar actualizaciones puntuales y personalizados por ejemplo regar por la tarde, en vez de por la noche, etc.

- **ELECTRODOMÉSTICOS**

Una forma básica para controlar algunos electrodomésticos (como la lavadora, el lavaplatos, el horno, o el aire acondicionado) es a través de la conexión y desconexión de la alimentación eléctrica. Esto podría servir con electrodomésticos más antiguos donde la

interrupción de la alimentación simplemente significaba que se paraba el programa actual y al volver a conectar la alimentación se ponían en marcha otra vez. Esta actuación puede hacer la mayoría de los sistemas domóticos pero no siempre es muy aconsejable debido a que por un lado no es bueno conectar y desconectar la alimentación de un electrodoméstico. Por otro lado ya no es seguro que un electrodoméstico moderno se vuelva a poner en marcha en el estado en que se encontraba al ser desconectado. Para las calderas de calefacción y sistemas de aire acondicionado es mejor controlar la entrada del termostato para su activación / desactivación. Aire acondicionados tipo split se suelen poder controlar a través de señales de infrarrojo. Aparatos como hornos y lavadoras ya vienen con sistemas avanzadas para poder realizar la programación directamente en ellas.

Algunos modernos electrodomésticos domóticos empiezan ya a ser controlables a través de un sistema más amplio. En esos tipos de electrodomésticos no solo suele ser posible controlar el encendido y apagado de forma remota o automática, sino también permiten funciones como avisos remotos de un malfuncionamiento, como una puerta abierta del frigorífico, o un filtro sucio; telegestión para poder diagnosticar de forma remota un malfuncionamiento de un aparato o cargar de forma remota un nuevo software.

## ○ **SEGURIDAD**

- **INCENDIO:** Los detectores pueden ser de humo, temperatura o manuales, ubicados en hall, oficinas, escaleras, cocheras, depósitos, etc. En caso de incendio el sistema avisara con mensajes en pantalla, en el teclado alfanumérico y con sirenas en las escaleras de los pisos. También podrá llamar a una cantidad de números que pueden ser del personal, bomberos, policía, etc.
- **ANTIRROBO:** Al ser un sistema integrador de distintas clases de sensores y dispositivos, los sistemas inteligentes tienen la ventaja de poder programar a la

misma unidad para distintas funciones, como por ejemplo para encender una luz o una alarma de intrusos. Por lo tanto, la misma instalación que se uso para la automatización de la luminaria ahora sirve para la de seguridad y viceversa.

El teclado alfanumérico sirve para ingresar el código de armado o la exclusión/inclusión de zonas, etc. Se pueden colocar: switch para la apertura y comprobación del estado de las puertas, detector de vidrio roto, sensor de movimiento o cualquier detector comercial de cualquier tipo, así como sirenas, etc. Todos los dispositivos se visualizarán en un plano para saber su estado.

#### **SERVICIOS DE SEGURIDAD:**

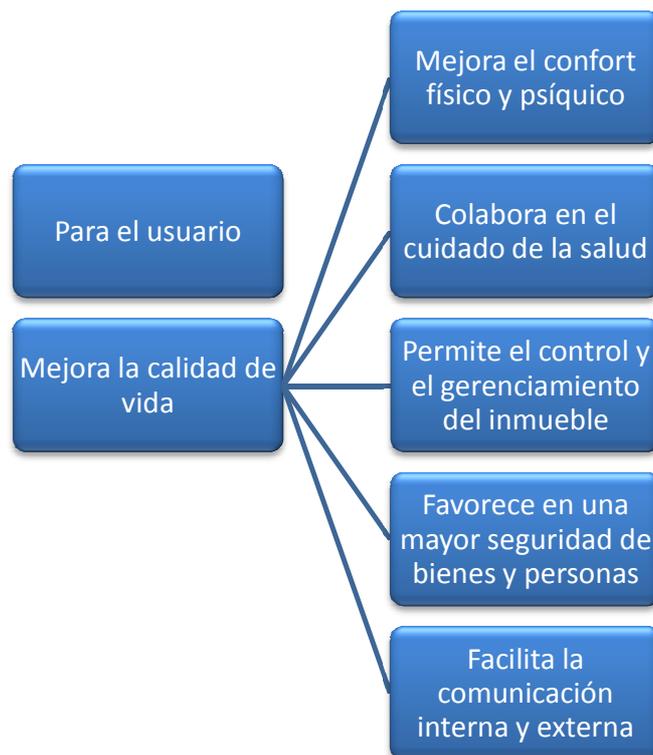
- Detectores de presencia
- Circuitos cerrados de televisión
- Comprobación del estado de las puertas.
- Vigilancia perimetral y periférica
- Control y bloqueo de accesos
- Protección anti-intrusos
- Control/comprobación de rondas de vigilancia
- Detección de incendios (humo y fuego)
- Detección de escapes o fugas de gas
- Evacuación automática de humo
- Señalización y megafonía de emergencia
- Telefonía de emergencia (interna o externa)
- Conexión con las fuerzas del orden, bomberos u otras

## 2.5 VENTAJAS Y DESVENTAJAS DE LA DOMOTICA

### 2.5.1 VENTAJAS:

Luego de analizar los sistemas domóticos se puede llegar a la conclusión de que se pueden obtener ventajas desde dos puntos de vista distintos como se muestra a continuación.

- Desde el punto de vista del Usuario:

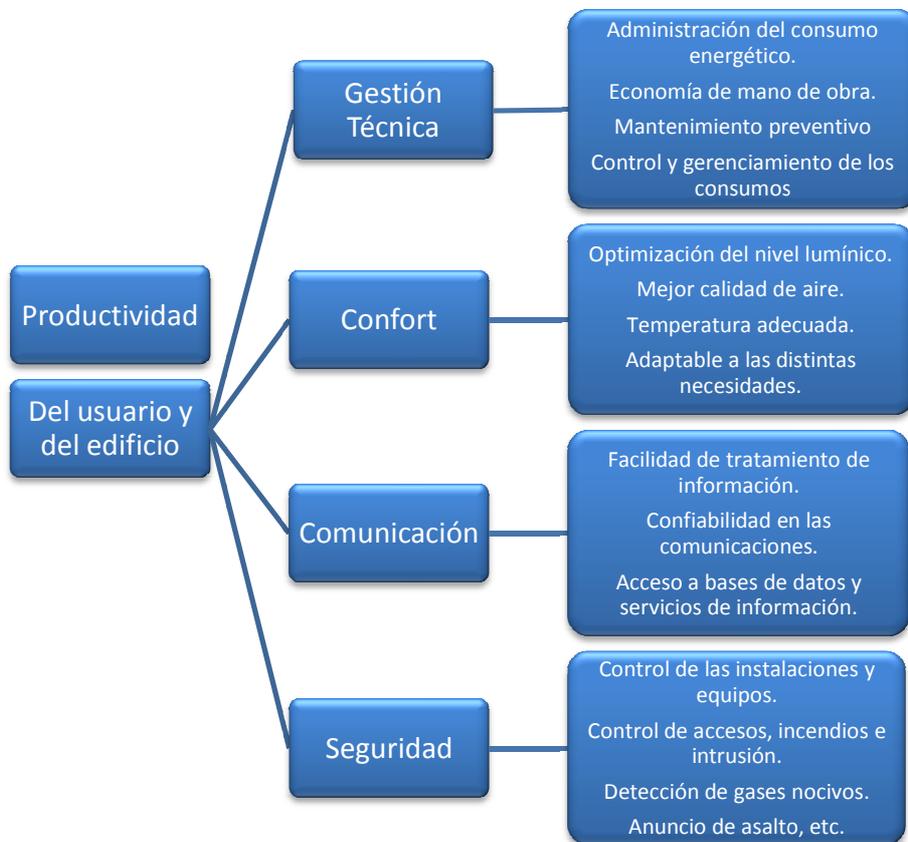


**Figura 5: Diagrama de ventajas**

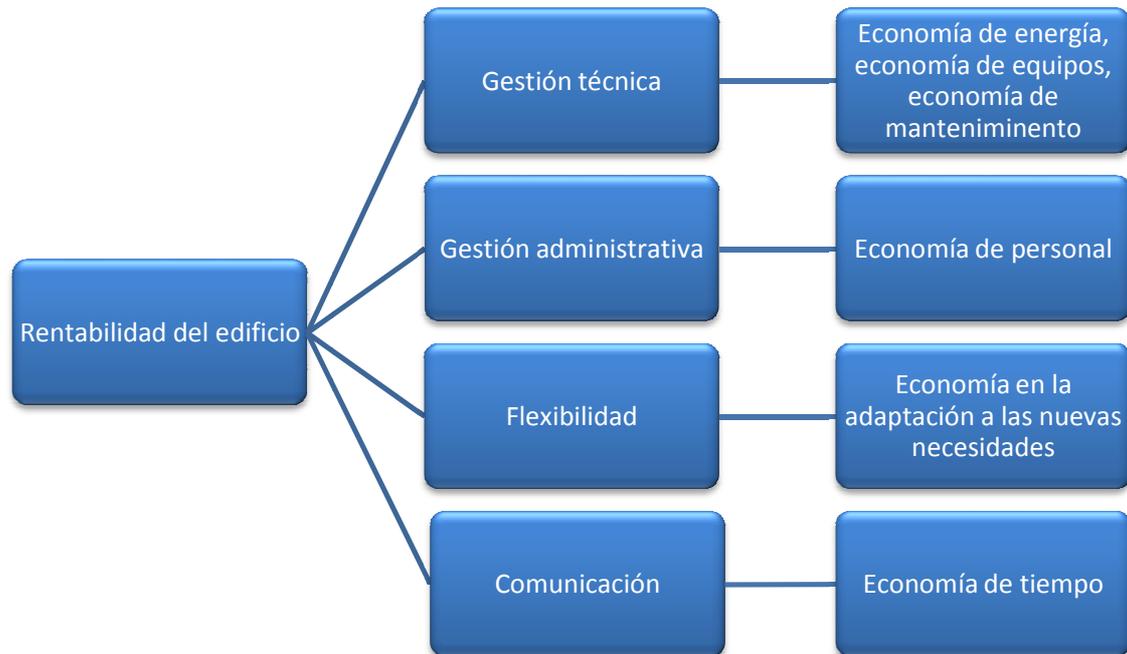
○ Desde el punto de vista económico:

La Productividad, ya sea del edificio como de las personas.

La rentabilidad, comprendido como el recupero de la inversión y los ahorros que se generan en las distintas áreas.



**Figura 6: Diagrama de ventajas**



**Figura 7: Diagrama de ventajas**

### 2.5.2 DESVENTAJAS:

Se podría decir que las desventajas son realmente pocas con respecto a las ventajas pero se pueden mencionar las siguientes:

- El precio aún es demasiado alto.
- Al ser relativamente nueva durante su aplicación se pueden experimentar fallos en los sistemas, etc.
- Se puede producir el aislamiento del usuario.
- Se puede dar un entorpecimiento del usuario, dependiendo del grado de automatización del sistema.

## **CAPÍTULO III**

### **ESTUDIO DE LOS SERVIDORES WEB EMBEBIDOS Y MICROCONTROLADORES.**

#### **3.1 MICROCONTROLADORES**

Los microcontroladores están conquistando el mundo. Están presentes en el trabajo, en la casa y en la vida en general. Se pueden encontrar controlando el funcionamiento de los ratones y teclados de los computadores, en los teléfonos, en los hornos microondas y los televisores del hogar. Pero la invasión acaba de comenzar y el nacimiento del siglo XXI será testigo de la conquista masiva de estos diminutos computadores, que gobernarán la mayor parte de los aparatos que fabricarán y usarán los humanos.

##### **3.1.1 CONTROLADOR Y MICROCONTROLADOR:**

Recibe el nombre de controlador el dispositivo que se emplea para el gobierno de uno o varios procesos. Por ejemplo, el controlador que regula el funcionamiento de un horno dispone de un sensor que mide constantemente su temperatura interna y, cuando traspasa los límites prefijados, genera las señales adecuadas que accionan los efectores que intentan llevar el valor de la temperatura dentro del rango estipulado.

Aunque el concepto de controlador ha permanecido invariable a través del tiempo, su implementación física ha variado frecuentemente. Hace tres décadas, los controladores se construían exclusivamente con componentes de lógica discreta, posteriormente se emplearon los microprocesadores, que se rodeaban con chips de memoria y E/S sobre una tarjeta de circuito impreso. En la actualidad, todos los elementos del controlador se han podido incluir en un chip, el cual recibe el nombre de microcontrolador. Realmente

consiste en un sencillo pero completo computador contenido en el corazón (chip) de un circuito integrado.

Un microcontrolador es un circuito integrado de alta escala de integración que incorpora la mayor parte de los elementos que configuran un controlador.

Un microcontrolador dispone normalmente de los siguientes componentes:

- Procesador o UCP (Unidad Central de Proceso).
- Memoria RAM para Contener los datos.
- Memoria para el programa tipo ROM/PROM/EPROM.
- Líneas de E/S para comunicarse con el exterior.
- Diversos módulos para el control de periféricos (temporizadores, Puertas Serie y Paralelo, CAD: Conversores Analógico/Digital, CDA: Conversores Digital/Analógico, etc.).
- Generador de impulsos de reloj que sincronizan el funcionamiento de todo el sistema.

Los productos que para su regulación incorporan un microcontrolador disponen de las siguientes ventajas:

- Aumento de prestaciones: un mayor control sobre un determinado elemento representa una mejora considerable en el mismo.
- Aumento de la fiabilidad: al reemplazar el microcontrolador por un elevado número de elementos disminuye el riesgo de averías y se precisan menos ajustes.
- Reducción del tamaño en el producto acabado: La integración del microcontrolador en un chip disminuye el volumen, la mano de obra y los stocks.
- Mayor flexibilidad: las características de control están programadas por lo que su modificación sólo necesita cambios en el programa de instrucciones.

El microcontrolador es en definitiva un circuito integrado que incluye todos los componentes de un computador. Debido a su reducido tamaño es posible montar el controlador en el propio dispositivo al que gobierna. En este caso el controlador recibe el nombre de controlador empotrado (embedded controller).

### **3.1.2 APLICACIONES DE LOS MICROCONTROLADORES:**

Cada vez existen más productos que incorporan un microcontrolador con el fin de aumentar sustancialmente sus prestaciones, reducir su tamaño y coste, mejorar su fiabilidad y disminuir el consumo.

Algunos fabricantes de microcontroladores superan el millón de unidades de un modelo determinado producidas en una semana. Este dato puede dar una idea de la masiva utilización de estos componentes.

Los microcontroladores están siendo empleados en multitud de sistemas presentes en nuestra vida diaria, como pueden ser juguetes, horno microondas, frigoríficos, televisores, computadoras, impresoras, módems, el sistema de arranque de un vehículo, etc. Y otras aplicaciones con las que seguramente no se está familiarizado como instrumentación electrónica, control de sistemas en una nave espacial, etc. Una aplicación típica podría emplear varios microcontroladores para controlar pequeñas partes del sistema. Estos pequeños controladores podrían comunicarse entre ellos y con un procesador central, probablemente más potente, para compartir la información y coordinar sus acciones, como, de hecho, ocurre ya habitualmente en cualquier PC.

### **3.1.3 RECURSOS COMUNES A TODOS LOS MICROCONTROLADORES:**

Al estar todos los microcontroladores integrados en un chip, su estructura fundamental y sus características básicas son muy parecidas. Todos deben disponer de los bloques esenciales: Procesador, memoria de datos y de instrucciones, líneas de E/S, oscilador de

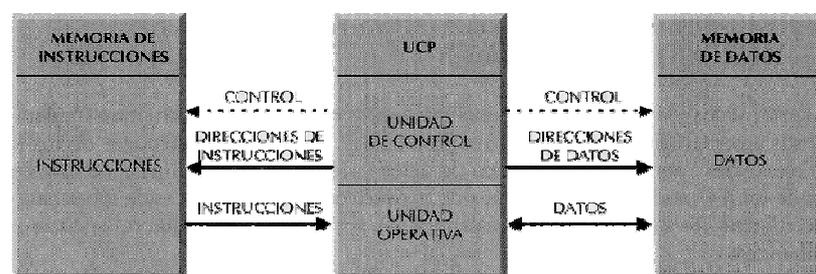
reloj y módulos controladores de periféricos. Sin embargo, cada fabricante intenta enfatizar los recursos más idóneos para las aplicaciones a las que se destinan preferentemente.

En este apartado se hace un recorrido de todos los recursos que se hallan en todos los microcontroladores describiendo las diversas alternativas y opciones que pueden encontrarse según el modelo seleccionado.

- **Arquitectura básica:**

Aunque inicialmente todos los microcontroladores adoptaron la arquitectura clásica de von Neumann, en el momento presente se impone la arquitectura Harvard. La arquitectura de von Neumann se caracteriza por disponer de una sola memoria principal donde se almacenan datos e instrucciones de forma indistinta. A dicha memoria se accede a través de un sistema de buses único (direcciones, datos y control).

La arquitectura Harvard (Figura 8) dispone de dos memorias independientes, una que contiene sólo instrucciones y otra que contiene sólo datos. Ambas disponen de sus respectivos sistemas de buses de acceso y es posible realizar operaciones de acceso (lectura o escritura) simultáneamente en ambas memorias.



**Figura 8: La arquitectura Harvard dispone de dos memorias independientes para datos y para instrucciones, permitiendo accesos simultáneos.**

- **El procesador o UCP**

Es el elemento más importante del microcontrolador y determina sus principales características, tanto a nivel hardware como software.

Se encarga de direccionar la memoria de instrucciones, recibir el código de la instrucción en curso, su decodificación y la ejecución de la operación que implica la instrucción, así como la búsqueda de los operandos y el almacenamiento del resultado.

Existen tres orientaciones en cuanto a la arquitectura y funcionalidad de los procesadores actuales.

- CISC: Un gran número de procesadores usados en los microcontroladores están basados en la filosofía CISC (Computadores de Juego de Instrucciones Complejo). Disponen de más de 80 instrucciones máquina en su repertorio, algunas de las cuales son muy sofisticadas y potentes, requiriendo muchos ciclos para su ejecución.

Una ventaja de los procesadores CISC es que ofrecen al programador instrucciones complejas que actúan como macros.

- RISC: Tanto la industria de los computadores comerciales como la de los microcontroladores están decantándose hacia la filosofía RISC (Computadores de Juego de Instrucciones Reducido). En estos procesadores el repertorio de instrucciones máquina es muy reducido y las instrucciones son simples y, generalmente, se ejecutan en un ciclo.

La sencillez y rapidez de las instrucciones permiten optimizar el hardware y el software del procesador.

- SISC: En los microcontroladores destinados a aplicaciones muy concretas, el juego de instrucciones, además de ser reducido, es "específico", o sea, las

instrucciones se adaptan a las necesidades de la aplicación prevista. Esta filosofía se ha bautizado con el nombre de SISC (Computadores de Juego de Instrucciones Específico).

- **Memoria**

En los microcontroladores la memoria de instrucciones y datos está integrada en el propio chip. Una parte debe ser no volátil, tipo ROM, y se destina a contener el programa de instrucciones que gobierna la aplicación. Otra parte de memoria será tipo RAM, volátil, y se destina a guardar las variables y los datos.

Hay dos peculiaridades que diferencian a los microcontroladores de los computadores personales:

- No existen sistemas de almacenamiento masivo como disco duro o disquetes.
- Como el microcontrolador sólo se destina a una tarea en la memoria ROM, sólo hay que almacenar un único programa de trabajo.

La RAM en estos dispositivos es de poca capacidad pues sólo debe contener las variables y los cambios de información que se produzcan en el transcurso del programa. Por otra parte, como sólo existe un programa activo, no se requiere guardar una copia del mismo en la RAM pues se ejecuta directamente desde la ROM.

Los usuarios de computadores personales están habituados a manejar Megabytes de memoria, pero, los diseñadores con microcontroladores trabajan con capacidades de ROM comprendidas entre 512 bytes y 8 k bytes y de RAM comprendidas entre 20 y 512 bytes.

Según el tipo de memoria ROM que dispongan los microcontroladores, la aplicación y utilización de los mismos es diferente. Se describen a continuación las cinco versiones de memoria no volátil que se pueden encontrar en los microcontroladores del mercado.

1. ROM con máscara: Es una memoria no volátil de sólo lectura cuyo contenido se graba durante la fabricación del chip. El elevado costo del diseño de la máscara sólo hace aconsejable el empleo de los microcontroladores con este tipo de memoria cuando se precisan cantidades superiores a varios miles de unidades.
2. OTP: (One Time Programmable). El microcontrolador contiene una memoria no volátil de sólo lectura "programable una sola vez" por el usuario. Es el usuario quien puede escribir el programa en el chip mediante un sencillo grabador controlado por un programa desde un PC.

La versión OTP es recomendable cuando es muy corto el ciclo de diseño del producto, o bien, en la construcción de prototipos y series muy pequeñas.

Tanto en este tipo de memoria como en la EPROM, se suele usar la encriptación mediante fusibles para proteger el código contenido.

3. EPROM: Los microcontroladores que disponen de memoria EPROM (Erasable Programmable Read Only Memory) pueden borrarse y grabarse muchas veces. La grabación se realiza, como en el caso de los OTP, con un grabador gobernado desde un PC. Si, posteriormente, se desea borrar el contenido, disponen de una ventana de cristal en su superficie por la que se somete a la EPROM a rayos ultravioleta durante varios minutos. Las cápsulas son de material cerámico y son más caros que los microcontroladores con memoria OTP que están hechos con material plástico.
4. EEPROM: Se trata de memorias de sólo lectura, programables y borrables eléctricamente EEPROM (Electrical Erasable Programmable Read Only Memory).

Tanto la programación como el borrado, se realizan eléctricamente desde el propio grabador y bajo el control programado de un PC. Es muy cómoda y rápida la operación de grabado y la de borrado. No disponen de ventana de cristal en la superficie.

Los microcontroladores dotados de memoria EEPROM una vez instalados en el circuito, pueden grabarse y borrarse cuantas veces se quiera sin ser retirados de dicho circuito. Para ello se usan "grabadores en circuito" que confieren una gran flexibilidad y rapidez a la hora de realizar modificaciones en el programa de trabajo.

El número de veces que puede grabarse y borrarse una memoria EEPROM es finito, por lo que no es recomendable una reprogramación continua. Son muy idóneos para la enseñanza y la Ingeniería de diseño.

Se va extendiendo en los fabricantes la tendencia de incluir una pequeña zona de memoria EEPROM en los circuitos programables para guardar y modificar cómodamente una serie de parámetros que adecuan el dispositivo a las condiciones del entorno. Este tipo de memoria es relativamente lenta.

5. FLASH: Se trata de una memoria no volátil, de bajo consumo, que se puede escribir y borrar. Funciona como una ROM y una RAM pero consume menos y es más pequeña. A diferencia de la ROM, la memoria FLASH es programable en el circuito. Es más rápida y de mayor densidad que la EEPROM.

La alternativa FLASH está recomendada frente a la EEPROM cuando se precisa gran cantidad de memoria de programa no volátil. Es más veloz y tolera más ciclos de escritura/borrado.

Las memorias EEPROM y FLASH son muy útiles al permitir que los microcontroladores que las incorporan puedan ser reprogramados "en circuito", es

decir, sin tener que sacar el circuito integrado de la tarjeta. Así, un dispositivo con este tipo de memoria incorporado al control del motor de un automóvil permite que pueda modificarse el programa durante la rutina de mantenimiento periódico, compensando los desgastes y otros factores tales como la compresión, la instalación de nuevas piezas, etc. La reprogramación del microcontrolador puede convertirse en una labor rutinaria dentro de la puesta a punto.

- **Puertas de Entrada y Salida**

La principal utilidad de las patitas que posee la cápsula que contiene un microcontrolador es soportar las líneas de E/S que comunican al computador interno con los periféricos exteriores.

Según los controladores de periféricos que posea cada modelo de microcontrolador, las líneas de E/S se destinan a proporcionar el soporte a las señales de entrada, salida y control.

- **Reloj principal**

Todos los microcontroladores disponen de un circuito oscilador que genera una onda cuadrada de alta frecuencia, que configura los impulsos de reloj usados en la sincronización de todas las operaciones del sistema.

Generalmente, el circuito de reloj está incorporado en el microcontrolador y sólo se necesitan unos pocos componentes exteriores para seleccionar y estabilizar la frecuencia de trabajo. Dichos componentes suelen consistir en un cristal de cuarzo junto a elementos pasivos o bien un resonador cerámico o una red R-C.

Aumentar la frecuencia de reloj supone disminuir el tiempo en que se ejecutan las instrucciones pero lleva aparejado un incremento del consumo de energía.

### **3.1.4 RECURSOS ESPECIALES**

Cada fabricante oferta numerosas versiones de una arquitectura básica de microcontrolador. En algunas amplía las capacidades de las memorias, en otras incorpora nuevos recursos, en otras reduce las prestaciones al mínimo para aplicaciones muy simples, etc. La labor del diseñador es encontrar el modelo mínimo que satisfaga todos los requerimientos de su aplicación. De esta forma, minimizará el costo, el hardware y el software.

Los principales recursos específicos que incorporan los microcontroladores son:

- Temporizadores o "Timers".
- Perro guardián o "Watchdog".
- Protección ante fallo de alimentación o "Brownout".
- Estado de reposo o de bajo consumo.
- Conversor A/D.
- Conversor D/A.
- Comparador analógico.
- Modulador de anchura de impulsos o PWM.
- Puertas de E/S digitales.
- Puertas de comunicación.

#### **3.1.4.1 Temporizadores o "Timers"**

Se emplean para controlar periodos de tiempo (temporizadores) y para llevar la cuenta de acontecimientos que suceden en el exterior (contadores).

Para la medida de tiempos se carga un registro con el valor adecuado y a continuación dicho valor se va incrementando o decrementando al ritmo de los impulsos de reloj o

algún múltiplo hasta que se desborde y llegue a 0, momento en el que se produce un aviso. Cuando se desean contar acontecimientos que se materializan por cambios de nivel o flancos en alguna de las patitas del microcontrolador, el mencionado registro se va incrementando o decrementando al ritmo de dichos impulsos.

#### **3.1.4.2 Perro guardián o "Watchdog"**

Cuando el computador personal se bloquea por un fallo del software u otra causa, se pulsa el botón del reset y se reinicializa el sistema. Pero un microcontrolador funciona sin el control de un supervisor y de forma continuada las 24 horas del día. El Perro guardián consiste en un temporizador que, cuando se desborda y pasa por 0, provoca un reset automáticamente en el sistema.

Se debe diseñar el programa de trabajo que controla la tarea de forma que refresque o inicialice al Perro guardián antes de que provoque el reset. Si falla el programa o se bloquea, no se refrescará al Perro guardián y, al completar su temporización, "ladrará y ladrará" hasta provocar el reset.

#### **3.1.4.3 Protección ante fallo de alimentación o "Brownout"**

Se trata de un circuito que resetea al microcontrolador cuando el voltaje de alimentación (VDD) es inferior a un voltaje mínimo ("brownout"). Mientras el voltaje de alimentación sea inferior al de brownout el dispositivo se mantiene reseteado, comenzando a funcionar normalmente cuando sobrepasa dicho valor.

#### **3.1.4.4 Estado de reposo ó de bajo consumo**

Son abundantes las situaciones reales de trabajo en que el microcontrolador debe esperar, sin hacer nada, a que se produzca algún acontecimiento externo que le ponga de nuevo en funcionamiento. Para ahorrar energía, (factor clave en los aparatos portátiles), los microcontroladores disponen de una instrucción especial (SLEEP en los PIC), que les

pasa al estado de reposo o de bajo consumo, en el cual los requerimientos de potencia son mínimos. En dicho estado se detiene el reloj principal y se "congelan" sus circuitos asociados, quedando sumido en un profundo "sueño" el microcontrolador. Al activarse una interrupción ocasionada por el acontecimiento esperado, el microcontrolador se despierta y reanuda su trabajo.

#### **3.1.4.5 Conversor A/D (CAD)**

Los microcontroladores que incorporan un Conversor A/D (Analógico/Digital) pueden procesar señales analógicas tan abundantes en las aplicaciones. Suelen disponer de un multiplexor que permite aplicar a la entrada del CAD diversas señales analógicas desde las patitas del circuito integrado.

#### **3.1.4.6 Conversor D/A (CDA)**

Transforma los datos digitales obtenidos del procesamiento del computador en su correspondiente señal analógica que saca al exterior por una de las patitas de la cápsula. Existen muchos efectores que trabajan con señales analógicas.

#### **3.1.4.7 Comparador analógico**

Algunos modelos de microcontroladores disponen internamente de un Amplificador Operacional que actúa como comparador entre una señal fija de referencia y otra variable que se aplica por una de las patitas de la cápsula. La salida del comparador proporciona un nivel lógico 1 ó 0 según una señal sea mayor o menor que la otra.

También hay modelos de microcontroladores con un módulo de tensión de referencia que proporciona diversas tensiones de referencia que se pueden aplicar en los comparadores.

#### **3.1.4.8 Modulador de anchura de impulsos o PWM**

Son circuitos que proporcionan en su salida impulsos de anchura variable, que se ofrecen al exterior a través de las patitas del encapsulado.

#### **3.1.4.9 Puertos de E/S digitales**

Todos los microcontroladores destinan algunas de sus patitas a soportar líneas de E/S digitales. Por lo general, estas líneas se agrupan de ocho en ocho formando Puertos.

Las líneas digitales de los Puertos pueden configurarse como Entrada o como Salida cargando un 1 ó un 0 en el bit correspondiente de un registro destinado a su configuración.

#### **3.1.4.10 Puertos de comunicación**

Con objeto de dotar al microcontrolador de la posibilidad de comunicarse con otros dispositivos externos, otros buses de microprocesadores, buses de sistemas, buses de redes y poder adaptarlos con otros elementos bajo otras normas y protocolos. Algunos modelos disponen de recursos que permiten directamente esta tarea, entre los que destacan:

- UART, adaptador de comunicación serie asíncrona.
- USART, adaptador de comunicación serie síncrona y asíncrona
- Puerta paralela esclava para poder conectarse con los buses de otros microprocesadores
- USB (Universal Serial Bus), que es un moderno bus serie para los PC
- Bus I2C, que es un interfaz serie de dos hilos desarrollado por Philips
- CAN (Controller Area Network), para permitir la adaptación con redes de conexionado multiplexado desarrollado conjuntamente por Bosch e Intel para el cableado de dispositivos en automóviles

## **3.2 SISTEMAS EMBEBIDOS**

Podemos definir al sistema embebido como un subsistema electrónico de procesamiento, programado para realizar una o pocas funciones que cumplan con un objetivo específico. Generalmente es parte integral de un sistema heterogéneo mayor, que puede incluir partes mecánicas, eléctricas y/o electromecánicas. Por el contrario, un sistema de procesamiento de propósito general, tal cual una computadora personal, puede realizar diferentes tareas dependiendo de la programación.

En la actualidad, los sistemas embebidos forman parte de la vida cotidiana de todos. La mayoría de los consumidores, a los que la palabra computadora les trae la imagen de una PC, no tienen información de la tremenda cantidad de procesadores o microcomputadoras que forman parte importante de su vida diaria. Se despiertan con la alarma del reloj digital, preparan el desayuno en un microondas, conducen automóviles asistidos por computadoras de abordo, realizan estudios médicos utilizando instrumentos biomédicos como los tomógrafos, monitores cardíacos y ecógrafos. El amplio rango de aplicaciones abarca desde entretenimiento y confort hasta vigilancia, seguridad, salud y protección al medio ambiente.

Otros ejemplos de una larga lista de sistemas embebidos son la mayoría de los periféricos de una computadora, los teléfonos celulares, los ruteadores, los PDAs, los que permiten operar los cajeros automáticos, discos duros, misiles, aviones y todo tipo de transporte, instrumental de medición y control, etc.

Existen un importante número de aplicaciones de sistemas embebidos para la reducción del consumo de recursos naturales: calderas inteligentes usan la mínima energía para

mantener confortable la temperatura del ambiente; sistemas de riego programables hacen uso del agua en horarios y períodos convenientes. La mayoría de los procesos industriales confían en algún tipo de control computarizado para optimizar el uso de la energía y alcanzar las regulaciones internacionales de protección ambiental.

Un sistema embebido (SE) está integrado por circuitos integrados programables, memoria flash o ROM, el correspondiente circuito impreso y el software embebido o empotrado como parte esencial del mismo sistema, conocido en inglés como firmware o embedded software. El software empotrado se utiliza para controlar los productos electrónicos y usualmente se ejecuta sobre un microprocesador interno, o en un microcontrolador, o en un procesador digital de señal (DSP), o en una compuerta programable en campo (FPGA), o en un controlador lógico programable (PLC) y a veces en una PC de propósitos generales adaptada para fines específicos.

Este software difiere del software convencional de una computadora de propósitos generales en una serie de características que justifica considerarlo como un nuevo campo de investigación y desarrollo dentro de las tecnologías de la información. Tiene entre otras las siguientes características:

- Tiene una interfaz directa con el hardware del dispositivo y es el intermediario entre el software de alto nivel y las funciones del hardware. Su lenguaje de programación, en la mayoría de los casos es de bajo y mediano nivel.
- Debido a que un sistema embebido está dedicado a una tarea específica, el diseño puede ser optimizado reduciendo los costos, el tamaño del producto y el consumo de potencia, a la vez de incrementar la confiabilidad y eficiencia.

- Los sistemas informáticos incluidos en productos electrónicos que controlan por ejemplo fábricas, tráfico aéreo y la distribución eléctrica se denominan sistemas de tiempo real. Los sistemas de tiempo real se diferencian de los sistemas informáticos de aplicación general en que deben cumplir con requisitos de tiempo que no sólo implica velocidad de respuesta, sino garantía de acción en el instante requerido de acuerdo a lo especificado. El software consiste en un programa que realiza tareas específicas el cual puede incluir un sistema operativo en tiempo real (RTOS). La naturaleza dedicada en tiempo real del sistema conduce a un mayor grado de dependencia y a una mayor integración con el hardware. Son una combinación de hardware y software en un mismo paquete. Sobre el principio de esta interrelación software-hardware es que se basa, en forma creciente, la producción de equipos electrónicos de nueva tecnología.
- Para alcanzar la meta de desarrollar sistemas insertados eficientes, es necesario emplear sistemas de arquitectura apropiados, hardware de interfaces y dispositivos periféricos, sensores e implementar robustos programas de software para su control. Todos ellos son utilizados en equipos y sistemas electrónicos que requieren el codiseño integrado de HW/SW.
- Generalmente se ejecuta en un hardware limitado tanto en velocidad como en cantidad de memoria.
- En numerosos casos requieren capacidad de auto-prueba, tanto del software como del hardware.

- Típicamente, el software debe ser extremadamente confiable, muy eficiente y compacto, y muy preciso en su respuesta al no siempre predecible instante de la transmisión de la información de ingresos y salidas (Interfaces I/O.) Su tolerancia a fallas es muy baja, porque una vez en manos del usuario en la mayoría de los casos es muy difícil o imposible de realizar cambios.

## **A. Componentes de un Sistema Embebido**

En la parte central se encuentra el microprocesador, microcontrolador, DSP, etc. Es decir la CPU o unidad que aporta inteligencia al sistema. Según el sistema puede incluir memoria interna o externa, un micro con arquitectura específica según requisitos.

La comunicación adquiere gran importancia en los sistemas integrados. Lo normal es que el sistema pueda comunicarse mediante interfaces estándar de cable o inalámbricas. Así un Sistema Embebido normalmente incorporará puertos de comunicaciones del tipo RS232, RS485, SPI, I<sup>2</sup>C, CAN, USB, IP, WiFi, GSM, GPRS, DSRC, etc.

El subsistema de presentación tipo suele ser una pantalla gráfica, táctil, LCD, alfanumérico, etc.

Se denominan actuadores a los posibles elementos electrónicos que el sistema se encarga de controlar. Puede ser un motor eléctrico, un conmutador tipo relé etc. El más habitual puede ser una salida de señal PWM para control de la velocidad en motores de corriente continua.

El módulo de E/S analógicas y digitales suele emplearse para digitalizar señales analógicas procedentes de sensores, activar diodos LED, reconocer el estado abierto cerrado de un conmutador o pulsador, etc.

El módulo de reloj es el encargado de generar las diferentes señales de reloj a partir de un único oscilador principal. El tipo de oscilador es importante por varios aspectos: por la frecuencia necesaria, por la estabilidad necesaria y por el consumo de corriente requerido. El oscilador con mejores características en cuanto a estabilidad y coste son los basados en resonador de cristal de cuarzo, mientras que los que requieren menor consumo son los RC. Mediante sistemas PLL se obtienen otras frecuencias con la misma estabilidad que el oscilador patrón.

El módulo de energía (power) se encarga de generar las diferentes tensiones y corrientes necesarias para alimentar los diferentes circuitos del Sistema Embebido. Usualmente se trabaja con un rango de posibles tensiones de entrada que mediante convertidores ac/dc o dc/dc se obtienen las diferentes tensiones necesarias para alimentar los diversos componentes activos del circuito.

Además de los convertidores ac/dc y dc/dc, otros módulos típicos, filtros, circuitos integrados supervisores de alimentación, etc. El consumo de energía puede ser determinante en el desarrollo de algunos Sistemas Embebidos que necesariamente se alimentan con baterías y es imposible su sustitución, con lo que la vida del Sistema Embebido suele ser a base de baterías.

## **B. Aplicaciones de un sistema embebido**

Los lugares donde se pueden encontrar los sistemas embebidos son numerosos y de varias naturalezas. A continuación se exponen varios ejemplos para ilustrar las posibilidades de los mismos:

- En una fábrica, para controlar un proceso de montaje o producción. Una máquina que se encargue de una determinada tarea hoy en día contiene numerosos circuitos electrónicos y eléctricos para el control de motores, hornos, etc. que

deben ser gobernados por un procesador, el cual ofrece un interfaz persona – máquina para ser dirigido por un operario e informarle al mismo de la marcha del proceso.

- Puntos de servicio o venta (POS, Point Of Service). Las cajas donde se paga la compra en un supermercado son cada vez más completas, integrando teclados numéricos, lectores de códigos de barras mediante láser, lectores de tarjetas bancarias de banda magnética o chip, pantalla alfanumérica de cristal líquido, etc. El sistema embebido en este caso requiere numerosos conectores de entrada y salida y unas características robustas para la operación continuada.
- Puntos de información al ciudadano. En oficinas de turismo, grandes almacenes, bibliotecas, etc. existen equipos con una pantalla táctil donde se puede pulsar sobre la misma y elegir la consulta a realizar, obteniendo una respuesta personalizada en un entorno gráfico amigable.
- Decodificadores y set-top boxes para la recepción de televisión. Cada vez existe un mayor número de operadores de televisión que aprovechando las tecnologías vía satélite y de red de cable ofrecen un servicio de televisión de pago diferenciado del convencional. En primer lugar envían la señal en formato digital MPEG-2 con lo que es necesario un procesado para decodificarla y mandarla al televisor. Además viaja cifrada para evitar que la reciban en claro usuarios sin contrato, lo que requiere descifrarla en casa del abonado. También ofrecen un servicio de televisión interactiva o web-TV que necesita de un software específico para mostrar páginas web y con ello un sistema basado en procesador con salida de señal de televisión.
- Sistemas radar de aviones. El procesado de la señal recibida o reflejada del sistema radar embarcado en un avión requiere alta potencia de cálculo además de

ocupar poco espacio, pesar poco y soportar condiciones extremas de funcionamiento (temperatura, presión atmosférica, vibraciones, etc.).

- Equipos de medicina en hospitales y ambulancias UVI – móvil.
- Máquinas de revelado automático de fotos.
- Cajeros automáticos.
- Pasarelas (Gateways) Internet-LAN.
- Y un sin fin de posibilidades aún por descubrir o en estado embrionario como son las neveras inteligentes que controlen su suministro vía Internet, PCs de bolsillo, etc.

### **C. Ventajas de un sistema embebido sobre las soluciones industriales tradicionales**

Los equipos industriales de medida y control tradicionales están basados en un microprocesador con un sistema operativo propietario o específico para la aplicación correspondiente. Dicha aplicación se programa en ensamblador para el microprocesador dado o en lenguaje C, realizando llamadas a las funciones básicas de ese sistema operativo que en ciertos casos ni siquiera llega a existir. Con los modernos sistemas embebidos se puede llegar a integrar el mundo del PC compatible con las aplicaciones industriales.

# CAPITULO IV

## BASIC STAMP 2

### 4.1 INTRODUCCION

El BASIC Stamp II (Figuras 9 y 10) es un pequeño computador que ejecuta programas en lenguaje PBASIC. El BS2-IC tiene 16 pines de (entrada / salida) I/O que pueden ser conectados directamente a dispositivos digitales o de niveles lógicos, tales como botones, diodos LEDs, altavoces, potenciómetros, y registros de desplazamiento. Además, con unos pocos componentes extras, estos pines de I/O pueden ser conectados a dispositivos tales como solenoides, relay, servomotores, motores de paso a paso, y otros dispositivos de alta corriente o tensión.

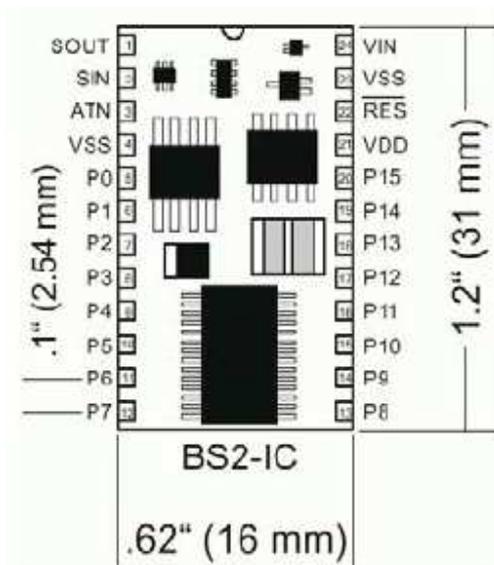


Figura 9: Diagrama esquemático BS2

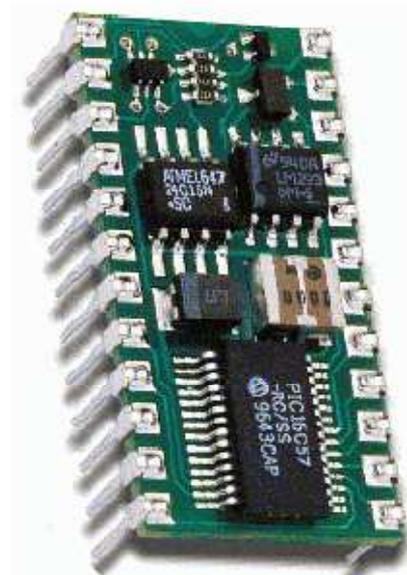


Figura 10: Basic Stamp 2

### **a. FORMATO DE CONVERSIÓN NUMÉRICA DEL BS2**

El editor PBASIC utiliza símbolos para identificar los distintos sistemas numéricos. Los números hexadecimales se representan con el signo de moneda (\$), los números binarios con el símbolo de porcentaje (%), los caracteres ASCII encerrados entre comillas (") y los números decimales de forma directa. Ejemplo:

75 'Decimal

%01001 'Binario

\$65 'Hexadecimal

"A" 'ASCII "

Las 3 instrucciones siguientes contienen el mismo significado:

DIRS = 14

DIRS = \$E

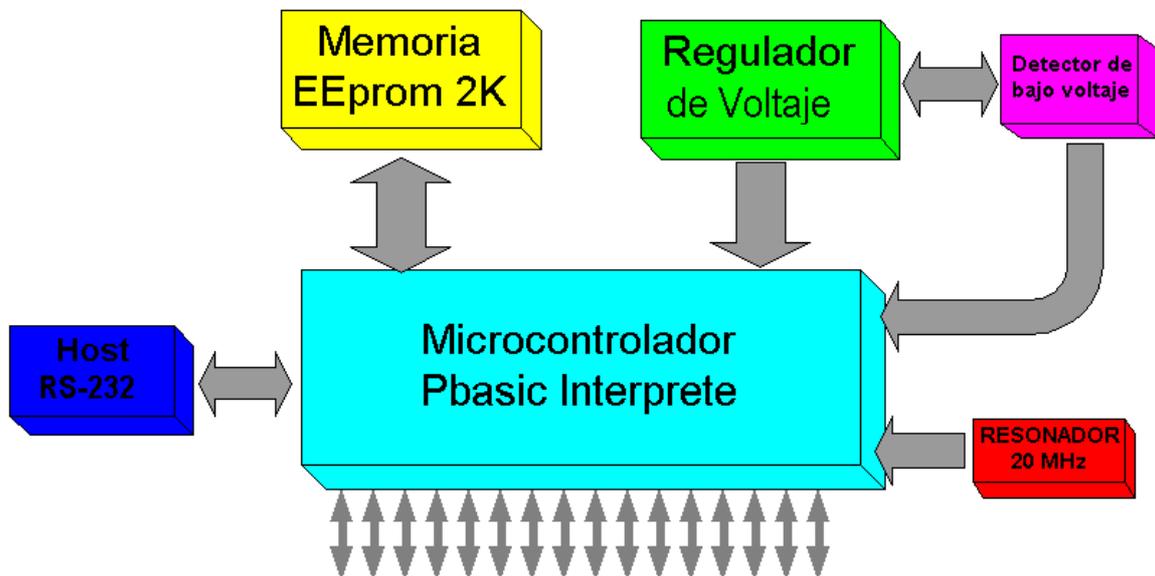
DIRS = %1110

### **b. FUNCIONAMIENTO INTERNO DEL BS2**

El diseño físico (Figura 11) consiste en un regulador de 5+ voltios, un oscilador de 20 MHz, una memoria EEPROM de 2K, un detector de bajo voltaje e chip intérprete PBASIC. Un programa compilado en PBASIC es almacenado en la EEPROM, desde donde el chip intérprete grabado en el microcontrolador lee y escribe las instrucciones.

Este chip intérprete ejecuta una instrucción cada vez, realizando la operación apropiada en los pines de I/O o en la estructura interna del chip intérprete. Debido a que el programa

PBASIC es almacenado en una EEPROM, puede ser reprogramado una cantidad cercana a 10 millones de veces.



*Figura 11: Diagrama en bloque del BS2*

La programación del BS2 se realiza directamente desde un computador personal PC, descargando los programas desde el software editor.

El Basic Stamp II es capaz de almacenar entre 500 y 600 instrucciones de alto nivel (PBASIC) y ejecuta un promedio de 4000 instrucciones / segundo. Para programar el BS2-IC, simplemente se debe conectar un cable serial preparado entre el BS2 y un PC, y ejecutar el software editor para crear y descargar el programa, a través del cable serial.

### **c. VENTAJAS DEL BS2 CON OTROS MICROCONTROLADORES**

La gran ventaja de los BS2 respecto a otros microcontroladores es sin duda que incorporan un chip intérprete de PBASIC, permitiendo ahorrar muchísimo tiempo en el desarrollo de aplicaciones dada su sencillez. El PBASIC es un lenguaje de programación basado en un BASIC estructurado orientado a entrada y salida de señales. La utilización

de sencillas instrucciones de alto nivel, permite programar los Basic Stamps para controlar cualquier aplicación llevada a cabo por un microcontrolador. Las instrucciones de PBASIC permiten controlar las líneas de (entrada / salida), realizar temporizaciones, realizar transmisiones serie asincrónica, utilizar el protocolo SPI, programar pantallas LCD, capturar señales analógicas, emitir sonidos, etc. y todo ello en un sencillo entorno de programación que facilita la creación de estructuras condicionales y repetitivas con instrucciones como IF...THEN o FOR...NEXT y la creación de etiquetas de referencia.

#### **d. APLICACIONES DE LOS BS2**

La única limitante de los Microcontroladores es la imaginación. La facilidad de un puerto abierto de (entrada / salida), la capacidad de evaluación de señales para luego decidir una acción y poder controlar dispositivos externos hacen que el microcontrolador sea el cerebro de los equipos. Estos son algunos ejemplos de áreas de aplicaciones:

- Electrónica Industrial (Automatizaciones)
- Comunicaciones e interface con otros equipos (RS-232)
- Interface con otros Microcontroladores
- Equipos de Mediciones
- Equipos de Diagnósticos
- Equipos de Adquisición de Datos
- Robótica (Servo mecanismos)
- Proyectos musicales
- Proyectos de Física
- Proyectos donde se requiera automatizar procesos artísticos
- Programación de otros microcontroladores
- Interface con otros dispositivos de lógica TTL:



diciendo a U1 que salve las instrucciones compiladas, llamadas fichas de instrucciones hexadecimales, en la memoria EEPROM (U2). Cuando el programa se ejecuta, U1 extrae las fichas de instrucciones hexadecimales de la memoria (U2), los interpreta como instrucciones PBASIC, y ejecuta las instrucciones equivalentes. U1 ejecuta el programa interno a una velocidad de 5 millones de instrucciones por segundo. Algunas instrucciones internas entran en una sola instrucción PBASIC2, así que PBASIC2 ejecuta más lentamente aproximadamente 3000 a 4000 instrucciones por segundo. El PIC16C57 tiene 20 pines en total, 16 están destinados a entrada / salida (I/O); 4 están destinados a la comunicación serial RS-232. En el circuito BS2 16 contactos están disponibles para uso general de los programas. Dos de los otros se pueden también utilizar para la comunicación serial asincrónica. Los dos restantes se utilizan solamente para interconectar con el EEPROM y no se pueden utilizar.

Los contactos de uso general de I/O, (P0-P15), se pueden interconectar con toda la lógica de +5 voltios moderna, de TTL (lógica del transistor-transistor) con CMOS (semiconductor de óxido metálico complementario). Las características son muy similares a las de los dispositivos de la serie lógica 74HCTxxx.

La dirección de entrada y salida de un contacto dado está enteramente bajo el control del programa. Cuando un contacto es declarado como una entrada de información, tiene muy poco efecto en los circuitos conectados con él, con menos de 1 microamperio (uA) de consumo interno. Hay dos propósitos para poner un pin en modo de entrada de información: (1) leer en modo pasivo el estado (1 o 0) de un circuito externo, o (2) para desconectar las salidas que manejan el pin. Para que el consumo de corriente sea el más bajo posible, las entradas de información deben siempre estar cerca de +5 voltios o cercano a la tierra. Los pines no utilizados deben ser declarados como salida aunque no

estén conectados; esto es para evitar que las entradas estén interpretando el ruido externo como señales lógicas.

Cuando un pin esta en modo de salida, internamente está conectado a la tierra o +5 voltios a través de un interruptor muy eficiente del circuito CMOS. Si se carga ligeramente (< 1mA), el voltaje de la salida estará dentro de algunos milivoltios cercanos de la fuente de alimentación (tierra para 0; +5V para 1). Cada pin puede manejar unos 25 mA. Pero Cada puerto de 8 pines no debe exceder de los 50 mA, con el regulador externo y 40 mA con el regulador interno; los pines de P0 a P7 conforman un puerto de 8 BITS y los pines de P8 a P15 el otro.

#### **c. 2048-byte de memoria borrable eléctricamente (U2)**

U1 se programa permanentemente en la fábrica y no puede ser reprogramada, así que los programas PBASIC2 se deben grabar en otra parte. Ése es el propósito de U2, una memoria EEPROM modelo 24LC16B eléctricamente borrable; la EEPROM es un buen medio para el almacenaje del programa porque conserva datos permanentemente aun sin energía y se puede reprogramar fácilmente.

Las EEPROMs tiene dos limitaciones: (1) toman un tiempo relativamente largo para programarlas (tanto como varios milisegundos), y (2) el límite de reprogramaciones es de (aproximadamente 10 millones). El propósito primario del BS2 es almacenar un programa.

#### **d. Circuito de Reset (U3)**

Cuando se enciende al BS2, le toma una fracción de segundo a la fuente estabilizar y alcanzar el voltaje de operación de 5+ voltios. Durante esta operación el circuito de Reset entra en acción.

La finalidad es detectar el voltaje de operación si es menor de 4.5+ el circuito de Reset mantendrá el Microcontrolador desconectado, cuando alcance un voltaje de unos 5+ voltios el circuito de Reset espera unos 30 mili-segundos para conectar al BS2.

Esta previsión evita posibles fallas del procesador y de la memoria (U1 y U2) que pueden incurrir en equivocaciones o bloqueos involuntarios. El circuito de Reset también es conectado externamente para reiniciar al microcontrolador.

#### **e. Fuente de alimentación (U4)**

El BS2 tiene dos formas de ser polarizado la primera es a través de un voltaje de alimentación no regulado el cual puede variar de (5.5+ a 15+ Voltios). La segunda consiste polarizándolo directamente a través de VDD. Es recomendable la segunda opción, pero se debe tener en cuenta que este voltaje no debe exceder los 5 Voltios. Y se puede realizar a través de un regulador externo como el LM7805.

#### **f. Host RS-232 (Q1, Q2, y Q3)**

Unas de las características más notables del BS2 es su capacidad para comunicarse con otras computadoras a través del puerto serial RS-232, esto se realiza de una manera natural. El puerto de interface host RS232 tiene dos funciones básicas la primera es para reprogramar al BS2, y la segunda para comunicarse externamente con otros dispositivos compatibles de comunicación asincrónica de formato RS-232 estándar.

Pero el puerto RS-232 opera con un voltaje de (+12 V, para indicar un 1 lógico y -12 V, para indicar un 0 lógico). Mientras que el BS2 opera con (+5 V, para indicar 1 lógico y 0 V, para indicar un 0 lógico). El circuito de interface se encarga entonces de las conversiones de voltajes necesarias para su correcta operación.

**g. Descripción de los pines del BS2**

<b>PIN</b>	<b>NOMBRE</b>	<b>DESCRIPCION</b>
1	SOUT	Serial Out: Conectar al puerto serial RX (DB9 pin 2)
2	SIN	Serial In: Conectar al puerto serial TX (DB9 pin 3)
3	ATN	Atención: Conectar al puerto serial DTR (DB9 pin 4)
4	GND	Tierra entre el puerto serial y el BS2
5-20	P0-P15	Puerto de propósitos generales, cada uno puede entregar 25 mA.
21	VDD	Voltaje regulado a +5 VDC
22	RES	Reset, basta con aterrizar y el BS2 reiniciará
23	GND	Tierra del BS2
24	PWR	Voltaje no regulado entre +5.5 y +15 VDC, si VDD es utilizado

**Tabla 1: Descripción de los pines del BS2**



**Figura 13: Ubicación de cada PIN**

## **A. Conjunto de instrucciones del Pbasic:**

El programa PBasic para el Basic Stamp 2 consiste en un conjunto de 36 comandos orientados a entrada y salida de señales y evaluación de variables para luego tomar una decisión, además de un conjunto de funciones matemáticas básicas.

Literalmente se le da órdenes al BS2:

- Ponte en alerta
- Envía un pulso por el Pin
- Mide el tiempo de esa señal por el Pin
- Envía una frecuencia por el Pin
- Cuenta el tren de pulso por el Pin
- Envía un estado lógico de +5 V, por el pin
- Revisa el puerto
- Descansa 2 minutos
- Espera 100 milisegundos
- Envía un dato por el puerto serial

El método de programación es lineal, es decir se ejecuta un comando a la vez, por lo general se programa de una forma en que se repita las instrucciones en un ciclo cerrado.

Los programas de PBasic contienen: variables de memoria, constantes, direccionamiento de puertos, instrucciones y sub-rutinas.

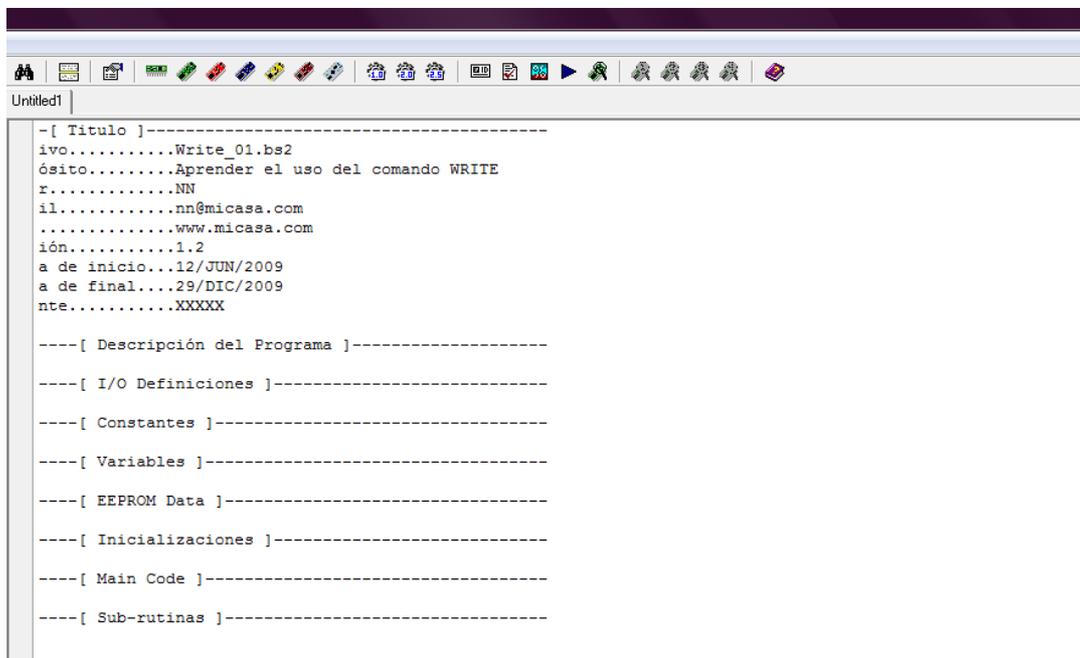
## **B. Estilo de programación**

En la programación se pueden adoptar estilos diferentes, dados los conocimientos básicos de cada instrucción, se plantea el problema el cual se debe solucionar con las funciones disponibles. La combinación de funciones es realmente ilimitada. El mismo

problema se puede solucionar de diversas formas, dependiendo del criterio lógico aprendido.

### C. Código Modelo

Como se mencionó anteriormente, aunque haya estilos diferentes de programación, es conveniente seguir un patrón de ordenamiento, el siguiente ejemplo (Figura 14) muestra un código modelo:



```
Untitled1
-[ Titulo ]-----
ivo.....Write_01.bs2
ósito.....Aprender el uso del comando WRITE
r.....NN
il.....nn@micasa.com
.....www.micasa.com
ión.....1.2
a de inicio...12/JUN/2009
a de final...29/DIC/2009
nte.....XXXXX

----[ Descripción del Programa ]-----
----[ I/O Definiciones ]-----
----[ Constantes ]-----
----[ Variables ]-----
----[ EEPROM Data ]-----
----[ Inicializaciones ]-----
----[ Main Code ]-----
----[ Sub-rutinas ]-----
```

**Figura 14: Patrón para programación**

La idea de este estilo es documentar todo el contenido y distinguir todos los procedimientos. Por lo general en un código PBASIC se debe llevar el siguiente orden:

1. Definir las constantes
2. Definir las variables de Entrada y Salida
3. Definir las variables de Programa

4. Inicializar el puerto
5. Direccionar las entradas y salidas
6. Iniciar los circuitos periféricos, si existen
7. Rutina principal (Main)
8. Rutinas Secundarias
9. Sub-Rutinas (Rutinas que se repiten)

Cuanto más explícito se escriba el código menos tiempo se perderá cuando se tenga que modificarlo. Los comentarios en el Pbasic no ocupan espacio en la memoria del BS2, simplemente el editor ignora los comentarios a la hora de descargar el código objeto al BS2.

### **4.3 ORGANIZACIÓN DE MEMORIA DEL BS2**

El BS2 tiene dos tipos de memoria; RAM para las variables del programa, y EEPROM para almacenar los programas en sí. La memoria EEPROM puede ser utilizada para almacenar datos de la misma forma que lo hace una computadora personal PC.

Una importante diferencia entre la memoria RAM y EEPROM:

- RAM pierde el contenido cuando el BS2 no tiene energía, cuando retorna la energía o cuando se reinicializa el BS2 el contenido completo de RAM se inicializa con 0.
- EEPROM retiene el contenido sin energía o con energía, mientras no se sobre escriba con otro programa o con la sentencia WRITE.

**a) Memoria RAM del BS2**

El BS2 tiene 32 BYTES de memoria RAM, 6 BYTES están reservados para los registros de entradas, salidas y direccionamiento del puerto para el control de (entradas / salidas) I/O. Los 26 BYTES restantes están destinados a variables de uso general.

**b) Jerarquías del Puerto P0-P15 (Registros: Dirs, Ins & Out)**

DIRS															
DIRH								DIRL							
DIRD				DIRC				DIRB				DIRA			
DIR 15	DIR 14	DIR 13	DIR 12	DIR 11	DIR 10	DIR 9	DIR 8	DIR 7	DIR 6	DIR 5	DIR 4	DIR 3	DIR 2	DIR 1	DIR 0
P15	P14	P13	P12	P11	P10	P9	P8	P7	P6	P5	P4	P3	P2	P1	P0

**Tabla 2: Registro DIRS**

OUTS															
OUTH								OUTL							
OUTD				OUTC				OUTB				OUTA			
OUT 15	OUT 14	OUT 13	OUT 12	OUT 11	OUT 10	OUT 9	OUT 8	OUT 7	OUT 6	OUT 5	OUT 4	OUT 3	OUT 2	OUT 1	OUT 0
P15	P14	P13	P12	P11	P10	P9	P8	P7	P6	P5	P4	P3	P2	P1	P0

**Tabla 3: Registro OUTS**

<b>INS</b>															
<b>INH</b>								<b>INL</b>							
<b>IND</b>				<b>INC</b>				<b>INB</b>				<b>INA</b>			
IN15	IN14	IN13	IN12	IN11	IN10	IN9	IN8	IN7	IN6	IN5	IN4	IN3	IN2	IN1	IN0
P15	P14	P13	P12	P11	P10	P9	P8	P7	P6	P5	P4	P3	P2	P1	P0

**Tabla 4: Registro INS**

Como se puede apreciar (Tablas 2, 3 y 4) los registros de direccionamiento DIRS, de entrada INS y de salida OUTS, contienen una jerarquía, el registro OUTS contiene el puerto completo del BS2, abarcado desde P0-P15. Pero la flexibilidad jerárquica consiste en que se pueden dividir y sub-dividir los registros según la necesidad. Si por ejemplo se quiere controlar el Pin 7, se debe elegir OUT7.

**c) Direccionamiento del puerto P0-P15**

Desde el principio de éste capítulo se han mencionado los términos (entrada / salida), E/S, Input/Output y I/O. Los microcontroladores por lo general contienen un puerto direccionable. Esto quiere decir que se puede elegir que pines serán salidas y cuales serán entradas.

Las personas que han trabajado con PLC (controles lógicos programables), saben que los PLC tienen definidas sus entradas y sus salidas. Por lo general más entradas que salidas.

En el caso de los microcontroladores esto es muy flexible hay aplicaciones donde todo podría ser salidas, otras donde solo serian entradas, y otras donde exista la combinación

de ambas. La última es la más común. Pero tal vez surja la pregunta de cómo un pin puede ser una entrada y luego puede ser una salida.

Esto es posible gracias a un circuito que aísla la entrada, el direccionamiento es un interruptor lógico que acciona el pin para fijarlo en modo de entrada o modo de salida. Cuando se enciende el BS2 todo el puerto se convierte en entrada automáticamente, hasta que el usuario no le indique que quiere cambiar el estado a modo de salida el puerto permanece como entrada.

Las entradas de los BS2 tienen una impedancia bastante elevada y el consumo de corriente que les toman a los dispositivos externos conectados a ellos es de menos 1  $\mu\text{A}$ , mientras que el voltaje máximo en una entrada no puede exceder los +5 Voltios. Por lo general se puede polarizar las entradas directamente desde +5 Voltios o directamente a tierra.

En modo de salida si se deben tomar todas las precauciones, cada salida es capaz de suministrar hasta 25 mA. Pero que sucede si un pin determinado que funciona en momento como entrada y está puesto a tierra luego se direcciona como salida? El resultado es un corto circuito en este pin y un daño irreparable en el microcontrolador.

Por lo general nunca se debe manejar directamente hacia las entradas voltajes directos, se debe hacer a través de resistencias en serie por el orden de 10 kOHM, esto evitara posibles daños. En el caso de que una salida se convierta en entrada no tiene efecto alguno en los dispositivos que maneja. Simplemente dejara de fluir la corriente eléctrica.

En las siguientes figuras se pueden visualizar diagramas de cómo funciona internamente el direccionamiento de un pin.

En la figura 15 el pin está en modo de entrada y en la figura 16 el mismo pin está en modo de salida.

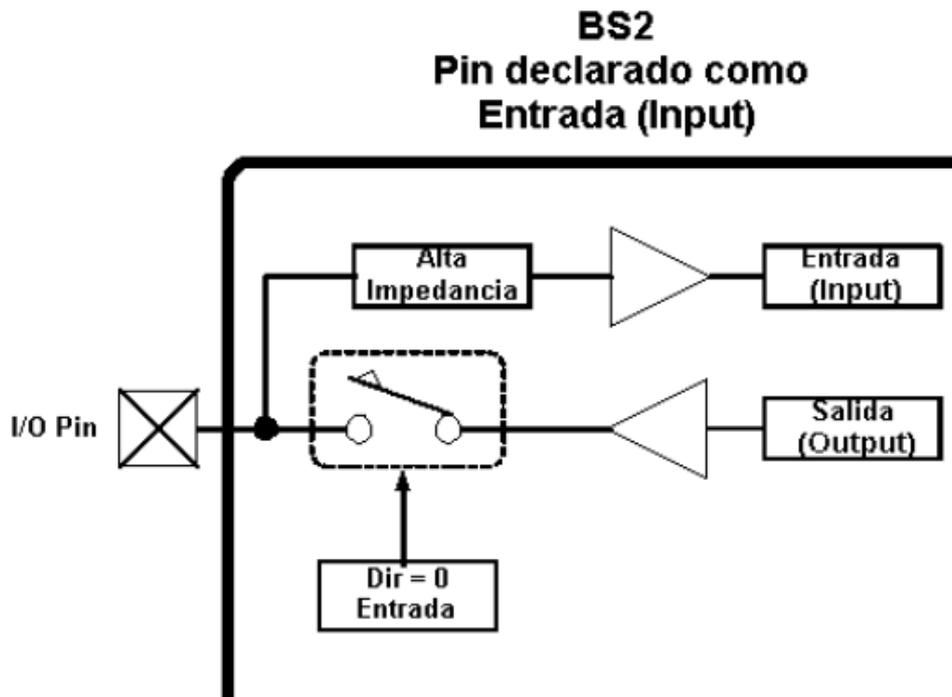


Figura 15: Diagrama en bloque del funcionamiento del direccionamiento

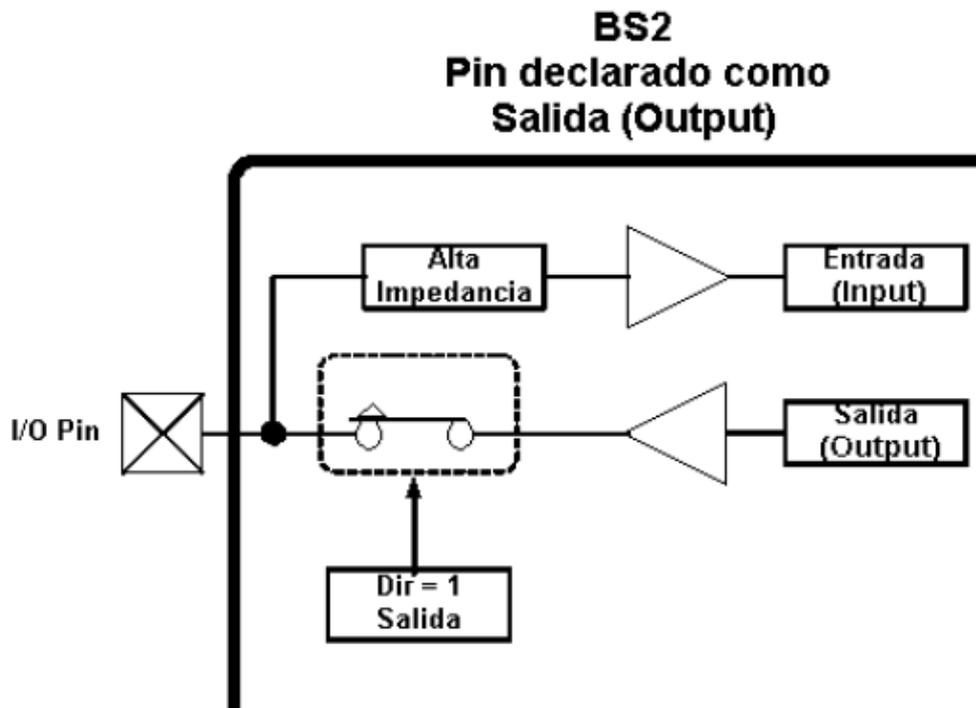


Figura 16: Diagrama en bloque del funcionamiento del direccionamiento

En la segunda figura se puede apreciar que cuando el pin esta en modo de salida, el circuito de entrada (input), se mantiene leyendo el estado de la salida, esto no causa mayor efecto se debe recordar que la entrada tiene una alta impedancia. El registro de entrada INS, es capaz en todo momento de leer el estado de cualquiera de sus pines desde P0-P15, sin importar que estén declarados como salida. Este registro puede leer la situación de cada pin.

Una vez que se direcciona un pin o un puerto este permanecerá indefinidamente en ese estado o hasta que se le indique otra dirección.

Para direccionar un pin como salida bastara con:

- DIR0 = 1 Direcciona el Pin 0 como salida

Para direccionar un pin como entrada bastara con:

- DIR0 = 0 Direcciona el Pin 0 como entrada

Un 1 direcciona un pin como salida, mientras que un 0 direcciona un pin como entrada, al principio esto puede resultar extraño, pero luego resultara natural. El direccionamiento se coloca por lo general al principio del programa. Si se quiere direccionar el puerto completo como salida el formato será (DIRS = %1111111111111111), el registro DIRS contiene el puerto completo.

En la tabla 5 se tiene que (DIRD = %0000), (DIRC = %1111), (DIRB = %1101) Y (DIRA = %0001).

DIRD				DIRC				DIRB				DIRA			
0	0	0	0	1	1	1	1	1	1	0	1	0	0	0	1
P15	P14	P13	P12	P11	P10	P9	P8	P7	P6	P5	P4	P3	P2	P1	P0

**Tabla 5: Direccionamiento de pines de salida**

Esto quiere decir que el puerto D, está definido como entrada, el puerto C, está definido como salida, el puerto B, contiene 3 salidas y una entrada y el puerto A, contiene 3 entrada y una salida. Nótese en la tabla anterior el orden de los pines de (P15 – P0). Este es el orden que siempre se debe llevar para mantener la secuencia. Del BIT más significativo y el menos significativo.

Esta definición también sería equivalente a (DIRS = %0000111111010001) o equivalente a (DIRH = %00001111) y (DIRL = %11010001).

Este formato es equivalente para los registros INS y OUTS.

#### **d) Variables de nombres fijos del BS2**

Las variables son donde se guardan los datos en forma temporal. Una variable es un símbolo que contiene un cierto valor. Ese valor puede ser cambiado bajo el control del programa y por lo tanto, el valor de las variables puede cambiar, pero su nombre no.

PBASIC puede utilizar variables con nombres de fábrica o variables con nombres definidos por el usuario. En cualquier caso el número de variable de 26 BYTE no varía. Las variables fijas tienen su orden de jerarquía (W0 es una variable tipo WORD de 16 BITS, que contiene a su vez a dos variables tipo BYTE de 8 BITS: B0 y B1). Por ejemplo si la variable W0 contiene el valor binario (%0011101011101001), entonces B0 contiene la parte baja de 8 BITS y B1 la parte alta de los 8 BITS.

W0 = %0011101011101001

B0 = %11101001

B1 = %00111010

Las variables predefinidas de fabrica no necesitan ser declaradas PBASIC las reconoce. Pero puede ser algo confuso sobre todo cuando se tiene un programa muy extenso.

Afortunadamente PBASIC da la libertad de que el usuario defina sus propias variables con el nombre más apropiado siempre relacionándola a la acción a ejecutar. En otras palabras se puede personalizar los nombres ejemplo: conteo\_general = 56, en vez de B1 = 56, es mucho más fácil relacionar un nombre que un nombre fijo como B1.

#### **e) Limite de longitud en los nombres de variables**

En PBASIC, los nombres de las variables pueden tener una longitud de hasta 32 caracteres. La longitud del nombre no tiene ninguna influencia en la velocidad de ejecución del programa. Por ejemplo, la instrucción: x = 38, tendrá la misma velocidad de ejecución que: éste\_es\_un\_nombre\_muy\_largo = 38.

De cualquier manera, en lugar de usar las variables predefinidas una recomendación sería utilizar un nombre específico para cada variable de acuerdo a algo relacionado con la aplicación utilizada o usando nombres con significado para el usuario.

#### **f) Declaración de Variables del BS2**

La declaración de variables consiste en fijarle un nombre de menos de 32 caracteres y un tamaño en BITS. Las declaraciones de variables hay que realizarlas al principio del programa o antes de utilizarlas. Para declarar variables se utiliza el comando VAR. La sintaxis es la siguiente:

nombre\_variable **VAR** tamaño

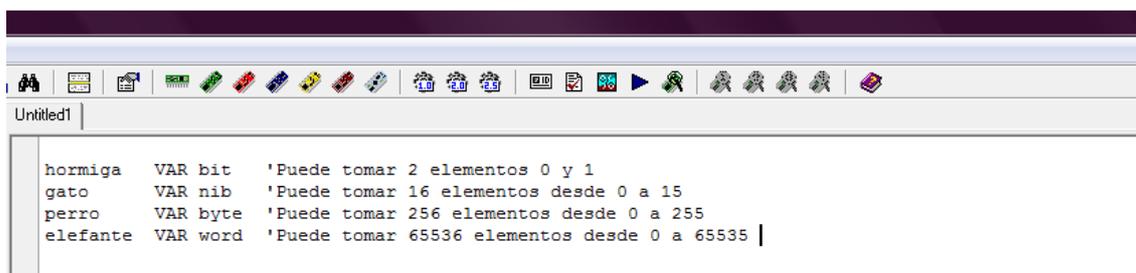
Donde:

- *Nombre\_variable* es el nombre que el usuario le asignara a la variable que no debe ser mayor de 32 caracteres, puede contener una secuencia de letras combinadas con números, también acepta el guión largo “\_”. En PBASIC, los nombre\_variable no son sensibles a mayúsculas y minúsculas.
- *Tamaño* establece el número de BITS reservados. PBASIC da 4 tipos de tamaño:

TIPO			ELEMENTOS	VALORES
BIT	1 Bit	2 <sup>1</sup>	2	(0-1)
NIB	4 Bits	2 <sup>4</sup>	16	(0-15)
BYTE	8 Bits	2 <sup>8</sup>	256	(0-255)
WORD	16 Bits	2 <sup>16</sup>	65536	(0-65535)

**Tabla 6: Número de bits reservados**

El espacio para cada variable es automáticamente destinado en la memoria del BasicStamp. El tamaño de las variables a utilizar depende de la cantidad de variaciones que se necesiten, a continuación se muestra un ejemplo (Figura 17) de algunos casos utilizando definiciones propias con la sentencia VAR:



```

hormiga  VAR bit    'Puede tomar 2 elementos 0 y 1
gato     VAR nib    'Puede tomar 16 elementos desde 0 a 15
perro    VAR byte   'Puede tomar 256 elementos desde 0 a 255
elefante VAR word   'Puede tomar 65536 elementos desde 0 a 65535 |

```

**Figura 17: Ejemplos de tamaños de variables**

Es buena práctica en principio para cada sub-rutina, evento o fórmula matemática utilizar o definir una variable para cada caso. Esto evitaría algún conflicto o error inesperado. Luego en la optimización de la aplicación se podrá notar cómo se pueden compartir muchas variables las cuales se utilizan en un evento y luego quedan libres sin efecto. Si

por ejemplo se necesita leer una entrada del microcontrolador se requiere de una variable tipo bit, pues la entrada solo tiene dos valores posible 0 o 1 lógico.

Si es necesario realizar un conteo del 1 al 10 es suficiente con una de tipo nib, pues esta puede contener 16 elementos. En este caso se podría utilizar una de tipo byte, pero se la estaría sub-utilizando.

Si se desea almacenar un conteo de 10,000 se necesita una variable tipo word, que puede almacenar hasta 65,536 elementos, la tipo byte en este caso sería menos que insuficiente, pues solo puede contener 256 elementos.

Cuando una variable excede el límite de su tamaño la variable retorna a su origen es decir a cero. Por ejemplo si una variable tipo byte realiza un conteo de 258 elementos el resultado sería 2, pues la variable cuando llega a 255 en el próximo conteo de 256 se desborda a cero, luego a uno y después a dos.

#### **g) Variables de grupo ARRAYS (ARREGLOS)**

Los arreglos de variables pueden ser creados en una manera similar a las variables:

**nombre\_variable VAR tamaño(n)**

Donde:

- *nombre\_variable* y *tamaño* es el mismo de las declaraciones de variables. El nuevo elemento es (n), y le dice PBASIC cuanto espacio reservar para la misma variable del tamaño especificado. Un ejemplo de creación de arreglo es el siguiente:

```
automovil var byte (10) ' Crea 10 variables tipo byte
```

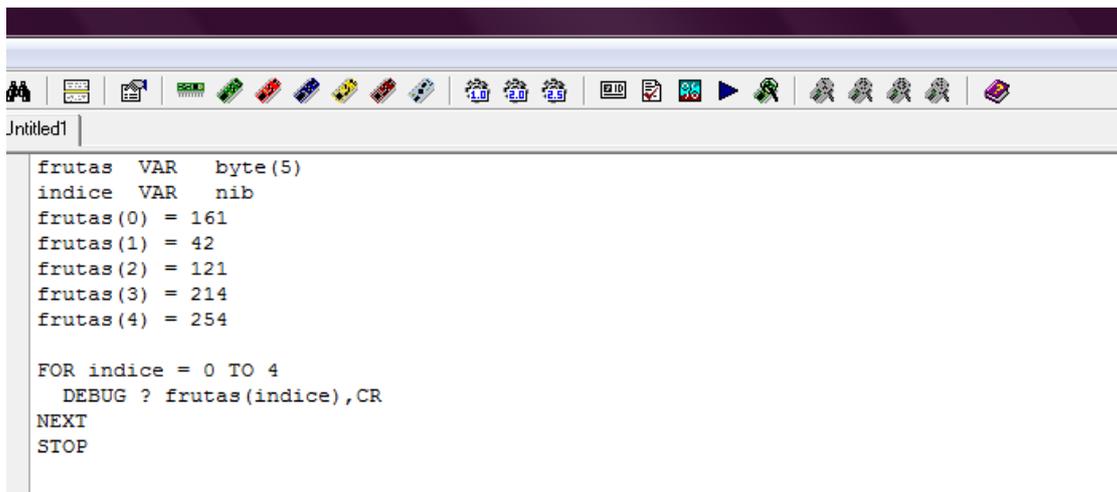
La primera ubicación dentro del arreglo es el elemento cero. En el arreglo *automovil* los elementos están numerados *automovil(0)* a *automovil(9)* conteniendo 10 elementos en

total. Dada la forma en que los arreglos están localizados en memoria hay límites de tamaño para cada tipo (Tabla 7).

Tamaño	Número máximo de elementos
BIT	208
NIB	52
BYTE	26
WORD	13

**Tabla 7: Límites de tamaño para arreglos**

Los arreglos son muy convenientes para recolección de datos, en vista de que el número de elementos (n) puede ser sustituido por otra variable. Ejemplo:



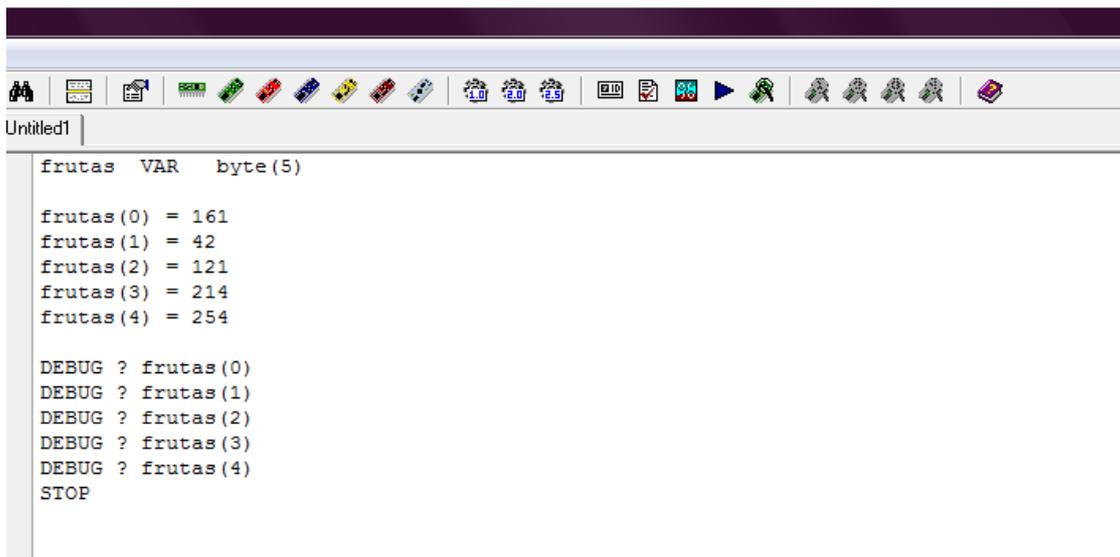
```
frutas VAR byte(5)
indice VAR nib
frutas(0) = 161
frutas(1) = 42
frutas(2) = 121
frutas(3) = 214
frutas(4) = 254

FOR indice = 0 TO 4
  DEBUG ? frutas(indice),CR
NEXT
STOP
```

**Figura 18: Ejemplo de uso de arreglos**

Se visualizan de forma dinámica las siguientes 5 variables almacenadas previamente.

Nótese que el (número de elementos) fue sustituido por una variable llamada índice, la cual puede tomar cualquier valor entre 0 a 15. Otra forma de realizar el mismo ejemplo (Figura 19) pero menos eficiente es:



```
frutas VAR byte (5)

frutas(0) = 161
frutas(1) = 42
frutas(2) = 121
frutas(3) = 214
frutas(4) = 254

DEBUG ? frutas(0)
DEBUG ? frutas(1)
DEBUG ? frutas(2)
DEBUG ? frutas(3)
DEBUG ? frutas(4)
STOP
```

**Figura 19: Otra forma de realizar el ejemplo anterior**

Se puede ver como en este ejemplo hubo que repetir la sentencia DEBUG 5 veces en vez de una como en el ejemplo anterior.

#### **h) ALIAS (Modificadores) de variables**

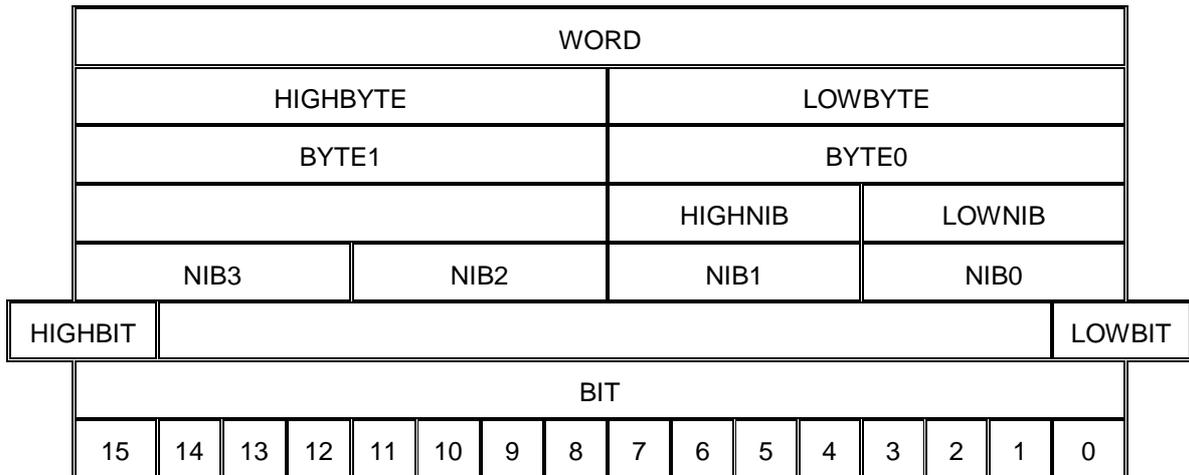
VAR también puede ser usado para crear un alias para otra variable. Esto es muy útil para acceder al interior de una variable.

dog var byte ´ dog es una variable tipo byte

fido var dog ´ fido es otro nombre de dog

En este ejemplo, fido es el alias de la variable dog. Cualquier valor almacenado en dog puede ser mostrado por fido y viceversa. Ambos nombres se refieren a lo mismo.

Con los ALIAS podemos acceder en cualquier momento al interior de una variable sin causar ninguna alteración a la variable original. En la tabla 8 se puede ver la jerarquía de los modificadores o alias de las variables.



**Tabla 8: Jerarquía de los modificadores**

## 4.4 ESTRUCTURA DE PROGRAMACIÓN EN BS2

### a. Declaración de Constantes del BS2

Las constantes pueden ser creadas de manera similar a las variables. Puede ser más conveniente utilizar un nombre de constante en lugar de un número.

Son creadas usando la palabra clave CON. Si el número necesita ser cambiado, únicamente habría que cambiarlo en un parte del programa donde se define la constante.

No pueden guardarse datos variables dentro de una constante. Esto es evidente. Más adelante se podrá apreciar la importancia de definir constantes en vez de fijar un número.

La sintaxis es la siguiente:

*nombre\_constante* **CON** *valor\_numerico*

Donde:

- *nombre\_constante* es el nombre se le asignara a la variable, no debe ser mayor de 32 caracteres, puede contener una secuencia de letras combinadas con números, también

acepta el guión largo “\_”. En PBASIC, los nombre\_variable no son sensibles a mayúsculas y minúsculas.

- *valor\_numerico* es un valor de (0-65535). PBASIC permite definir constantes numéricas en tres bases: decimal, binaria y hexadecimal. Los valores binarios son definidos usando el prefijo “%” y valores hexadecimales usando el prefijo “\$”. Los valores decimales se toman por defecto y no requieren prefijo. Ejemplo:

100 ´ valor decimal 100

%100 ´ valor binario para el decimal 4

\$100 ´ valor hexadecimal para el decimal 256.

“A” ´ ASCII equivalente a decimal (65).

Algunos ejemplos son:

bateria CON 12

continentes CON 5

libro CON \$E7

encendido CON %1101

detener CON “s”

Es posible calcular expresiones a través de constantes previamente definidas:

temperatura con 37

grados\_c con (temperatura\*5)/9

grados\_f con (grados\_c –32)

## b. Etiquetas de direccionamiento (labels)

Para marcar una referencia o dirección dentro del programa se puede referenciar con los comandos GOTO ó GOSUB. PBASIC no permite número de línea y no requiere que cada línea sea etiquetada. Cualquier línea PBASIC puede comenzar con una etiqueta de línea que es simplemente un identificador finalizando por dos puntos (:). Las etiquetas no deben ser mayores de 32 caracteres, pueden contener una secuencia de letras combinadas con números también acepta el guión largo “\_”. Las etiquetas no son sensibles a mayúsculas y minúsculas. En otras palabras las etiquetas:

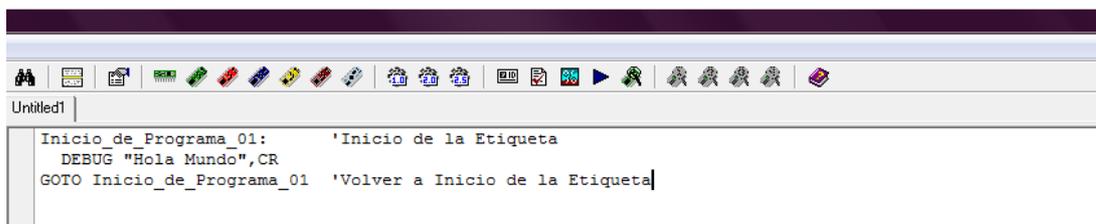
Inicio\_de\_Programa\_01:

inicio\_de\_programa\_01:

INICIO\_DE\_PROGRAMA\_01:

iNICIO\_DE\_pROGRAMA\_01:

Tienen el mismo significado o valor para el editor Pbasic.



```
Untitled1 |
Inicio_de_Programa_01:      'Inicio de la Etiqueta
  DEBUG "Hola Mundo",CR
GOTO Inicio_de_Programa_01  'Volver a Inicio de la Etiqueta|
```

**Figura 20: Ejemplo de programa en PBASIC**

## c. Comentarios

Un comentario de PBASIC comienza con el apóstrofe ('). Todos los demás caracteres de esa línea se ignoran. Los comentarios aunque no son obligatorios y a veces pueden aparentar ser innecesarios son de vital importancia, pues la experiencia muestra que al

escribir programas y luego revisarlos después de un tiempo puede ser difícil recordar el por qué se escribieron algunas rutinas. Al momento de realizar un programa todo puede aparentar muy claro. Pero quizás después de un tiempo éste puede aparentar sin sentido. Como regla específica se debe comentar cada línea de programación para gastar unos segundos en el momento y evitar gastar varias horas después. Realmente los comentarios no ocupan espacio en memoria pues el compilador lo ignora como función, así que no se debe evitar el hecho de documentar explícitamente los programas.

#### **d. Declaraciones múltiples**

Para permitir programas más compactos y agrupamientos lógicos de comandos relacionados, PBASIC soporta el uso de (:) para separar comandos ubicados en la misma línea. Los siguientes dos ejemplos son equivalentes.

```
W2 = W0
```

```
W0 = W1
```

```
W1 = W2
```

Es lo mismo que:

```
W2 = W0 : W0 = W1 : W1 = W2
```

En los dos casos, el tamaño del código generado es el mismo.

Otro ejemplo:

Loop:

```
HIGH 0
```

```
PAUSE 500
```

```
LOW
```

PAUSE 500

GOTO Loop

Es lo mismo que:

Loop: HIGH 0 : PAUSE 500 : LOW 0 : PAUSE 500 : GOTO Loop

Aunque ambos códigos realizan la misma función, este último puede resultar confuso.

## 4.5 REFERENCIA DE COMANDOS

### 4.5.1 Clasificación de comandos:

El lenguaje Pbasic está conformado por 37 comandos, 24 funciones matemáticas, instrucciones para definiciones de variables, constantes y etiquetas de referencia. La combinación de estos comandos con las referencias de direcciones (etiquetas), conformaran un programa en Pbasic. La complejidad del mismo dependerá de lo que se quiera realizar. La gran mayoría de instrucciones de Pbasic está orientada al procesamiento de señales de entrada y salida de uso industrial, las otras están destinadas a la evaluación de datos y cálculos matemáticos.

- **Bifurcaciones**

**IF...THEN:** Evaluación para tomar una decisión según la condición sea Falso o Verdadera.

**BRANCH:** GOTO computado (equiv. a ON..GOTO).

**GOTO:** Salta a una posición especificada dentro del programa, a través de una dirección de etiqueta.

**GOSUB:** Llama a una subrutina PBASIC en la dirección de etiqueta especificada.

- **Ciclos repetitivos controlados**

**FOR...NEXT:** Bucle controlado, ejecuta declaraciones en forma repetitiva.

- **Acceso de Datos a la EEprom**

**DATA:** Almacena datos en la EEPROM del BS2.

**READ:** Lee un BYTE de la EEPROM del BS2.

**WRITE:** Graba un BYTE en EEPROM del BS2.

- **Búsqueda de Datos y Tabla de Datos**

**LOOKUP:** Obtiene un valor constante de una tabla.

**LOOKDOWN:** Busca un valor en una tabla de constantes.

**RANDOM:** Genera numero aleatorio (0-65535).

- **Señales Digitales**

**INPUT:** Convierte un pin en entrada.

**OUTPUT:** Convierte un pin en salida.

**REVERSE:** Convierte un pin de salida en entrada o uno de entrada en salida.

**LOW:** Hace bajo la salida de un pin.

**HIGH:** Hace alto la salida del pin.

**TOGGLE:** Cambia el estado de un pin si es alto lo convierte en bajo, si es bajo lo convierte alto.

**PULSIN:** Mide el ancho de pulso en un pin.

**PULSOUT:** Genera pulso en un pin.

**BUTTON:** Entrada de pulsadores momentáneos, Antirebote y auto-repetición de entrada en el pin especificado.

**COUNT:** Cuenta el numero de pulsos en un pin en un tiempo determinado.

**XOUT:** Salida X-10 Formato de control de electrodomésticos y alarmas a través de la red eléctrica 110 AC / 60Hz.

- **Comunicación Asincrónica**

**SERIN:** Entrada serial asincrónica (RS-232).

**SEROUT:** Salida serial asincrónica (RS-232).

- **Comunicación Sincrónica**

**SHIFTIN:** Entrada serial sincrónica (SPI).

**SHIFTOUT:** Salida serial sincrónica (SPI).

- **Señales Análogas**

**PWM:** Salida modulada en ancho de pulso a un pin.

**RCTIME:** Mide capacitores o resistores en función del tiempo de carga conformado por un circuito (RC).

- **Funciones de Tiempo**

**PAUSE:** Hace una pause de (0-65535) milisegundos.

- **Funciones de Sonido**

**FREQOUT:** Produce una o dos 2 frecuencias en un pin especificado.

**DTMFOUT:** Produce tonos DTMF de formato telefónico en un pin específico.

- **Control de Energía**

**NAP:** Apaga el procesador por un corto periodo de tiempo.

**SLEEP:** Descansa el procesador por un periodo de tiempo.

**END:** Detiene la ejecución e ingresa en modo de baja potencia

- **Depuración de Programa**

**DEBUG:** Salida de Datos por el puerto de programación.

# CAPITULO V

## Parallax Internet Netburner Kit

### 5.1 INTRODUCCION

El módulo denominado PINK (Parallax Internet Netburner Kit) constituye un servidor web embebido (incrustado en el circuito del módulo) capaz de alojar páginas web propias, desde las cuales se puede mostrar datos para ser gestionados por el circuito BS, gracias al soporte de 100 variables y registros especiales; y además da la posibilidad de interactuar con estos datos.

### 5.2 CONFIGURACIÓN DEL MÓDULO

La configuración del módulo es posible gracias a una interfaz basada en web, con lo que la puesta en marcha resulta fácil. Así como que cada dispositivo en una red Ethernet requiere una dirección IP, el módulo PINK también. Así pues, el primer paso consiste en especificar la IP con la que este servidor web deberá mostrarse a Internet. Esto se puede realizar tanto manualmente (introduciendo la IP, la máscara de red, la puerta de enlace y el servidor DNS), como automáticamente (en modo DHCP: Dynamic Host Configuration Protocol). Esta operación se realiza mediante el software que acompaña al módulo en un CD.

Una vez configurado, es posible acceder a la "home" del servidor tecleando simplemente en el navegador `http://xx.xx.xx.xx` , donde "xx.xx.xx.xx" es la dirección IP que se le ha asignado. La página principal que se carga por defecto es "index.html"; si no se ha cargado previamente alguna página, se carga la que viene por defecto en el módulo.

Por supuesto, para evitar accesos no autorizados, el módulo PINK aporta protección mediante contraseñas. Desde la página web de configuración

([http://xx.xx.xx.xx/nb\\_factory.htm](http://xx.xx.xx.xx/nb_factory.htm) , se las puede crear tanto para el acceso vía FTP como para el acceso a las páginas web. Es posible prescindir de esto último o, mejor aún, tal vez lo que se necesite sea especificar una clave sólo a determinadas páginas. En esto, el campo "Web file password filter" permitirá especificar ese grupo de páginas, incluso archivos, según el texto que sea introducido y que filtrará las páginas cuyo nombre contengan dicho texto. Por ejemplo, si se especifica en el campo del filtro el texto "pass", los archivos con nombre "indexpass.htm", "passindex.htm", "passtime.jpg" solicitarán las claves de acceso, mientras que el resto de los archivos serán libremente accesibles.

### **5.3 ACCESO VÍA FTP**

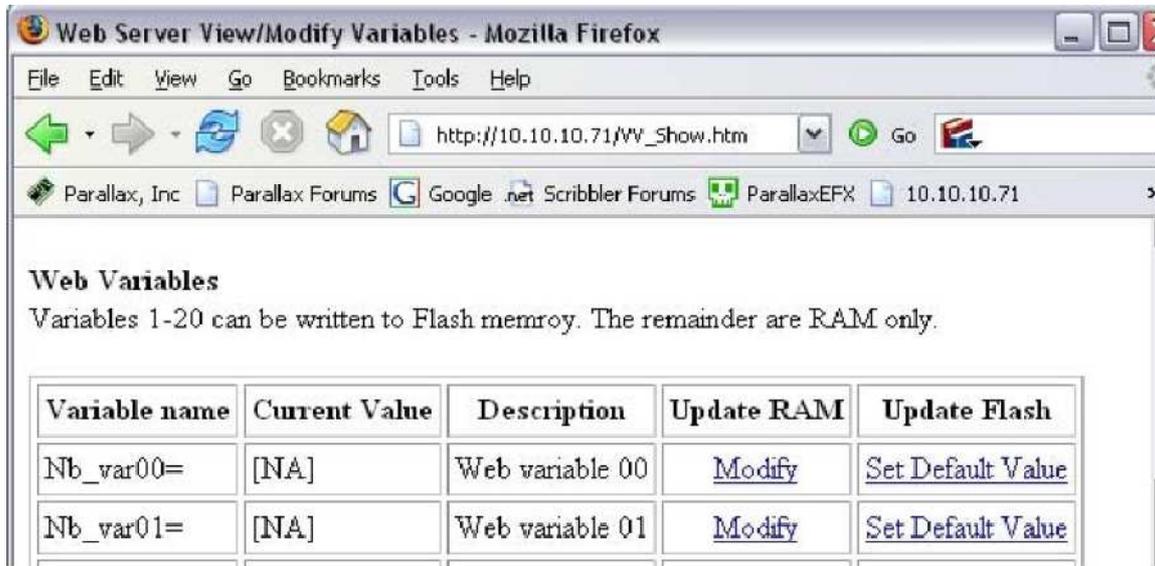
Las páginas y archivos pueden ser cargados en el módulo PINK mediante protocolo FTP (Protocolo de Transferencia de Archivos). Escribiendo <ftp://xx.xx.xx.xx> en un navegador Internet Explorer se mostrará una ventana donde poder mover los archivos mediante "drag-and-drop" (arrastrar y soltar) en la memoria del módulo del servidor.

### **5.4 VARIABLES**

El módulo PINK cuenta con 100 variables de propósito general y 18 son variables del sistema. Cada variable puede almacenar a 64 bytes máximo. Las primeras 20 variables del propósito generales (0 a través de 9) pueden escribirse en la memoria flash como valores predeterminados. Las variables de propósito generales restantes (20 a través de 99) son variables de la memoria RAM y los datos se pierden cuando se apaga.

Adicionalmente a las 20 variables de propósito general que son escritas en la memoria flash, las variables de sistema de e-mail y UDP también pueden ser almacenadas en la memoria flash convirtiéndose también en valores predeterminados. Las variables de

estado son de solo lectura, pero algunas son establecidas por el PINK y determinaran la configuración IP, incluyendo la dirección IP, la máscara de red, la puerta de enlace y el servidor DNS.



**Figura 21: Variables modificables**

Todas las variables pueden ser accedidas y modificadas a través la página web tecleando XXX.XXX.XXX.XXX/VV\_Show.htm (donde XXX.XXX.XXX.XXX es la dirección IP del modulo PINK) (Figura 21). Solo a través de esta página se pueden modificar los valores de dichas variables que serán escritas en la memoria flash al momento de presionar el botón Set Default Value.

Para acceder serialmente a las variables a través del BASIC Stamp, se debe referir al nombre de la variable usando las últimas dos letras que son referidas en la página de modificación de variables del PINK. Para leer el contenido de una variable se deber usar el siguiente formato: **!NBORXX**.

Nótese que XX puede ser reemplazado por las dos letras del nombre de la variable. Por ejemplo, para leer un valor de la variable de propósito general 01 se debe usar esta forma de comando: **!NB0R01**.

Para leer el contenido de una variable de estado sobre la IP, se puede usar el siguiente formato: **!NBORSI**.

Nótese que en los ejemplos anteriores !NB es seguido de el número cero, no de la letra O.

En la página web del modulo PINK están incluidos ejemplos de programas para leer variables del BS2.

Para escribir a las variables se puede usar el siguiente formato: **!NB0WXX:DD**.

Nuevamente XX será reemplazado por las dos letras del nombre de la variable y DD será reemplazado por el dato a ser enviado a la variable. La manera en que se envíe la variable es como se almacenará.

Ejemplo:

```
SEROUT TX, Baud, ["NB0W01:100",CLS] `Se pone el valor de 100 en  
la variable 01
```

## 5.5 UTILIZANDO LAS VARIABLES

Acceder al valor de cualquier variable desde una página web creada por el usuario es tan fácil como usar una referencia HTML como nombre de variable. Por ejemplo, para crear una página que muestre el valor de la variable 01, se puede escribir el siguiente código

HTML:

```
<html>  
The value in variable 01 is: <Nb_var01>  
</html>
```

Después, se guarda esta página con un nombre de archivo con extensión htm o html (por ejemplo: test.html), se abre una sesión FTP y mediante "drag-and-drog" se carga dicha página en el módulo PINK.

Ahora, escribiendo en el navegador la url `http://xx.xx.xx.xx/ test.html` se mostrará la página ejemplo, tal como se ve en la figura 22.



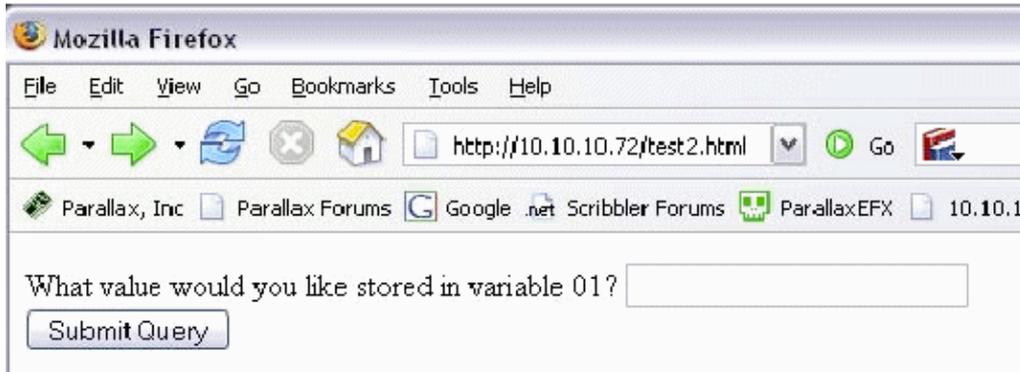
**Figura 22: Página que muestra un valor de la variable**

En este ejemplo, la palabra "hello" está en la variable 01. Notese que si el valor de esta variable es cambiado, la página se actualizará mostrando el cambio, de manera que esto favorecerá el trasiego de datos dinámicos que puedan ser mostrados en la página web cada vez que cambien.

Escribir valores en las variables no es nada complicado usando el método HTML POST (formulario). Por ejemplo: para crear una página que permita escribir o cambiar el valor de la variable 01 (Figura 23), se puede escribir lo siguiente:

```
<html>
<FORM method="post" action="/test.html">
<P>
What value would you like stored in variable 01?
<INPUT name="Nb_var01" type="text" size="24" maxlength="63">
<INPUT type="submit">
</P>
```

</FORM>  
</html>



**Figura 23: Ejemplo para cambiar el valor de una variable**

Se realiza la misma operación de antes para cargar la página en el módulo PINK y, una vez que se visualiza el formulario en el navegador, se puede escribir el valor que se desee, tal como muestra la imagen.

En este momento es cuando el circuito BS entra en escena: ¿cómo actúa el BS para leer o proporcionar valores a dichas variables? Para leer el valor de una variable en el módulo PINK con el BS, el programa en el BS debe enviar serialmente el siguiente comando: !NB0Rxx donde "xx" es el número de la variable que se desea leer (00 a 99). Este comando se puede usar en un programa como el siguiente:

```
' PINK_01.bs2
' {$STAMP BS2}
' {$PBASIC 2.5}
NBVAR VAR Byte
SEROUT 8,396,["!NB0R06"]
SERIN 7,396,[NBVAR]
DEBUG DEC NBVAR
END
```

O el siguiente programa ejemplo, para leer variables tipo "string" (cadena de caracteres):

```
' PINK_02.bs2
' {$STAMP BS2}
```

```
' {$PBASIC 2.5}
NBVAR VAR Byte(16)
SEROUT 8,396,["!NB0R06"]
SERIN 7,396,[STR NBVAR\16\CLS]
DEBUG STR NBVAR
END
```

Escribir valores en las variables también es muy simple; para esto se utiliza el siguiente comando: !NB0Wxx:DD donde "xx" es la variable en la que se desea escribir, y "DD" es el dato que se desea enviar. El comando debe ser seguido por un CLS. Por ejemplo:

```
' PINK_03.bs2
' {$STAMP BS2}
' {$PBASIC 2.5}
SEROUT 8,396,["!NB0W06:25",CLS]
END
```

## 5.6 REGISTROS ESPECIALES

Uno de los aspectos más interesantes entre las capacidades del módulo PINK es que tiene 9 registros para propósitos especiales, entre estos el de poder enviar un email debidamente formateado:

- **Nb\_varET** corresponde al registro email TO:
- **Nb\_varEF** corresponde al registro email FROM:
- **Nb\_varES** corresponde al registro email SUBJECT:
- **Nb\_varEC** corresponde al registro email CONTENT: Este registro lleva el nombre del archivo con el contenido del mensaje. Por supuesto, este archivo deberá haber sido cargado previamente en el módulo PINK.
- **Nb\_varEV** corresponde al registro email del servidor SMTP.
- **Nb\_varST** corresponde al registro de estado del módulo PINK. Se trata de una variable de sólo lectura, cuyos bits son usados para información relacionada con el estado del módulo PINK y la red.

- **Nb\_varSV** guarda el número de la última variable actualizada desde una página web tipo formulario. Para leer el valor de esta variable el comando es el siguiente:  
!NB0SV
- **Nb\_varBI** es usado para guardar la dirección IP destino para mensajes UDP (User Datagram Protocol).
- **Nb\_varBM** es usado para guardar el contenido de un mensaje UDP.

Para enviar un mensaje e-mail desde el módulo PINK, deben ser especificadas las variables de los registros: Nb\_varET, Nb\_varEF, Nb\_varES, Nb\_varEC y Nb\_varEV. El siguiente programa puede servir de ejemplo:

```
' PINK_05.bs2
' {$STAMP BS2}
' {$PBASIC 2.5}
SEROUT 8,396,["!NB0WET:null@parallax.com",CLS]
SEROUT 8,396,["!NB0WEF:PINKmodule@parallax.com",CLS]
SEROUT 8,396,["!NB0WES:This is a test message from PINK.",CLS]
SEROUT 8,396,["!NB0WEC:message.txt",CLS]
SEROUT 8,396,["!NB0WEV:your.SMTP.server.address.here",CLS]
SEROUT 8,396,["!NB0SM"]
END
```

Nótese que la instrucción !NB0SM le dice al módulo PINK que envíe el mensaje.

El modulo PINK puede enviar y recibir mensajes bajo el protocolo abierto UDP. Los mensajes viajan encapsulados, de manera que lo que lee el receptor es exactamente lo que envía el emisor.

Para enviar un mensaje UDP, las variables de registro Nb\_varBI y Nb\_varBM deben ser inicializadas. Nb\_varBI debe contener la dirección IP destino y Nb\_varBM debe contener el mensaje UDP que se desea enviar. Una vez inicializadas las mencionadas variables, se

usa el siguiente comando: !NB0SB y el mensaje será enviado. El siguiente programa puede servir perfectamente para enviar un mensaje UDP:

```
' PINK_06.bs2
' {$STAMP BS2}
' {$PBASIC 2.5}
SEROUT 8,396,["!NB0WBI:10.10.10.71",CLS]
SEROUT 8,396,["!NB0WBM:This is a UDP test message",CLS]
SEROUT 8,396,["!NB0SB"]
END
```

# **CAPITULO VI**

## **DESARROLLO DEL PROYECTO**

### **6.1 FASE DE ANALISIS**

#### **6.1.1 ESPECIFICACIÓN DE REQUERIMIENTOS SOFTWARE Y HARDWARE.**

##### **a. Equipos a utilizar**

###### **Recursos Hardware**

- 1 Computador
- 1 Impresora
- 1 Cable programador de microcontroladores

###### **Recursos Software**

- Windows XP con Service Pack 2
- Windows Server 2003
- Basic Stamp Editor v2.3.9
- Netburner Kit CD

##### **b. Materiales a utilizar**

- Basic Stamp 2
- PINK Module
- Componentes electrónicos pasivos y activos.
- Papel

- Tinta
- Copias
- Memoria Flash USB

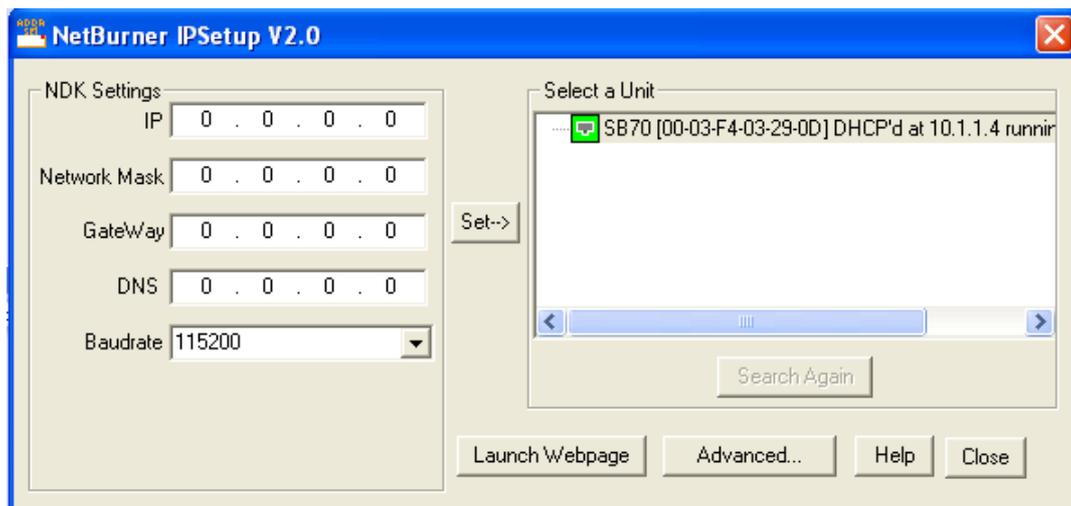
**c. Otros**

- Internet
- Libros
- Revistas



La dirección IP puede ser estática (asignada manualmente) o dinámica (asignada por el servidor DHCP). La configuración de la red dependerá de la opción que se use.

Cuando se corre el programa IPSETUP el módulo PINK aparece en la parte superior derecha de la ventana (Figura 25). Se selecciona entonces el módulo dando click sobre la opción SB70.



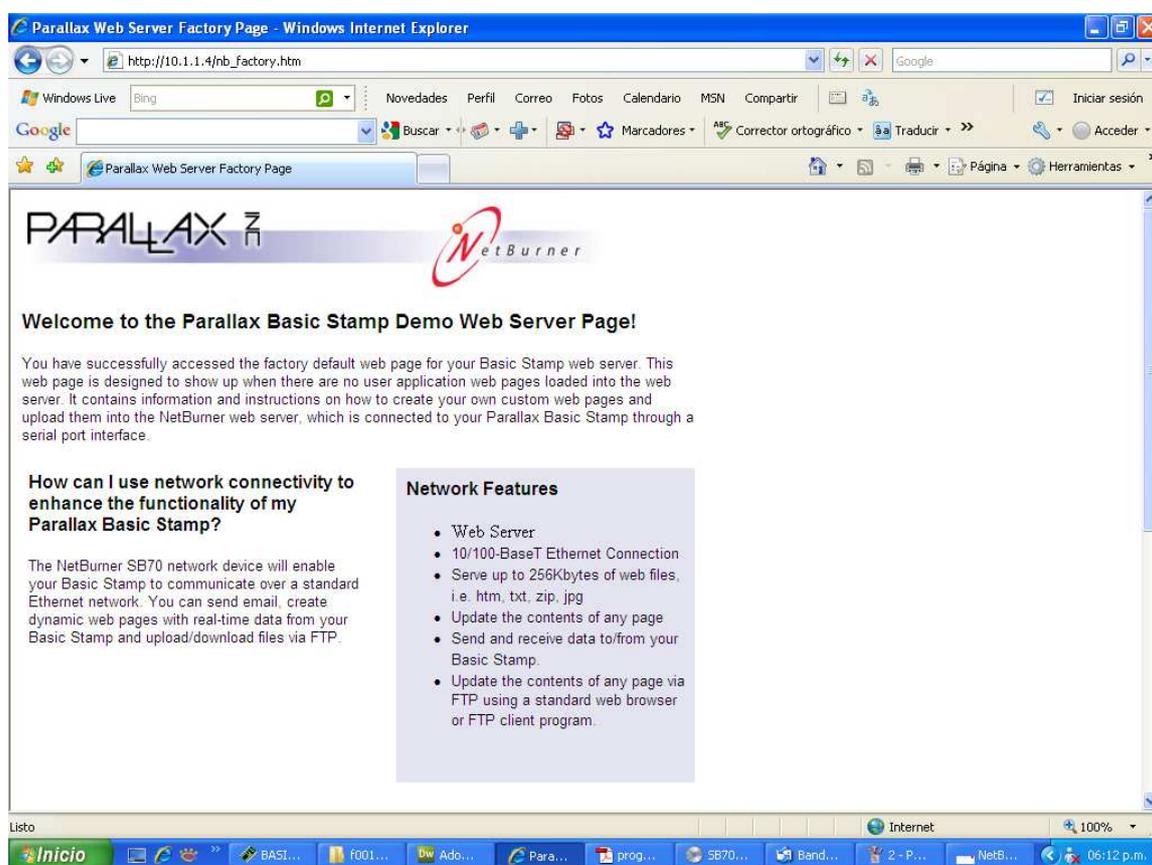
**Figura 25: IPSETUP del modulo PINK**

Para establecer una dirección IP estática se debe digitar la IP, la máscara de red, la puerta de enlace y el servidor DNS dentro de los campos correspondientes. Para establecer una dirección dinámica se deben ubicar todos los campos en 0.0.0.0. Es importante no cambiar el valor del Baud Rate que está ubicado por defecto para evitar que la comunicación con el microcontrolador se vea afectada. En el caso del presente proyecto se estableció que la dirección IP sea dinámica. Luego de establecer la dirección IP se debe dar click sobre el botón Set. Si el módulo PINK no reaparece en la lista, se debe dar click en el botón de Search Again. Debe haber la seguridad de que el dispositivo se encuentra seleccionado para acceder a Launch Webpage que abrirá una

ventana del explorador para las nuevas asignaciones que se deberán realizar en el módulo.

- **Configuración Web**

Cuando el explorador se despliega (Figura 26) se dispondrá de la página que viene de fábrica por defecto. Cuando el módulo no tiene cargadas otras páginas aparecerá automáticamente la siguiente pantalla.

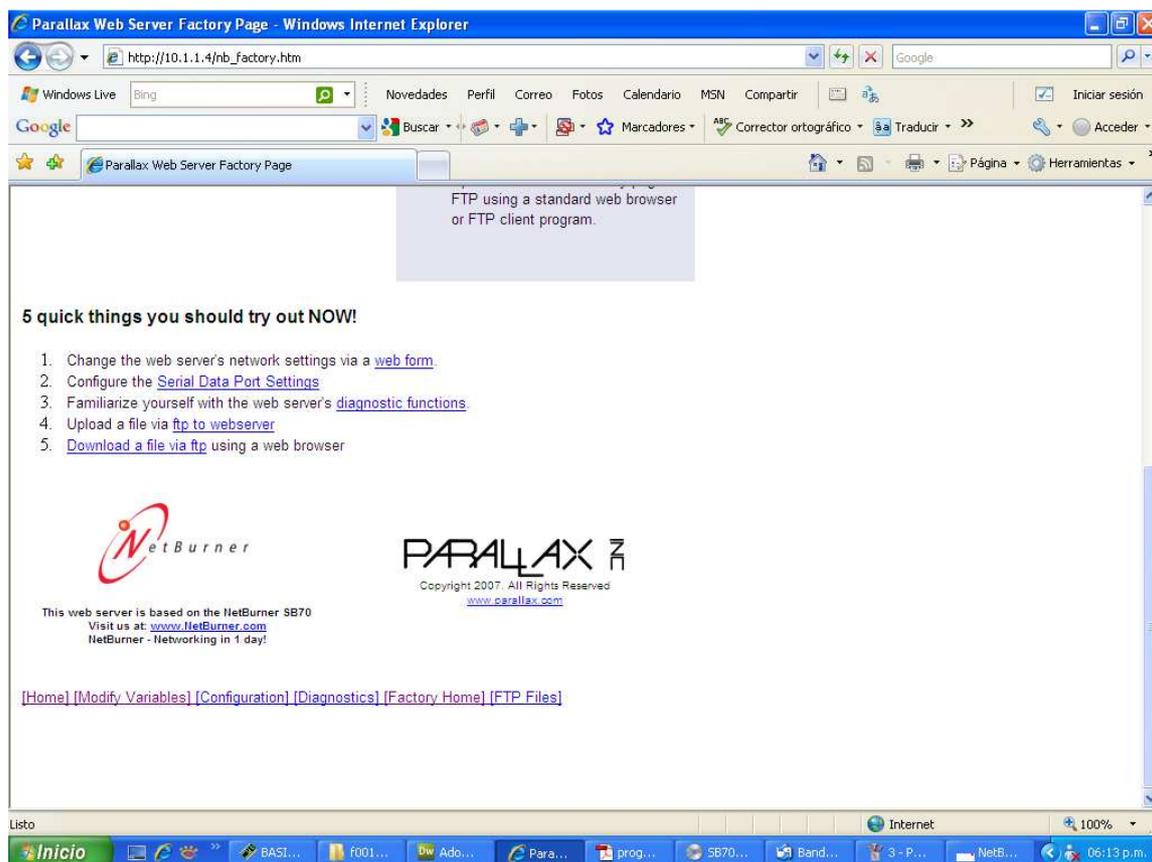


**Figura 26: Página de fábrica**

La dirección de ésta página es XXX.XXX.XXX.XXX/nb\_factory.htm donde XXX.XXX.XXX.XXX es la dirección IP del módulo PINK. Desde esta página se puede

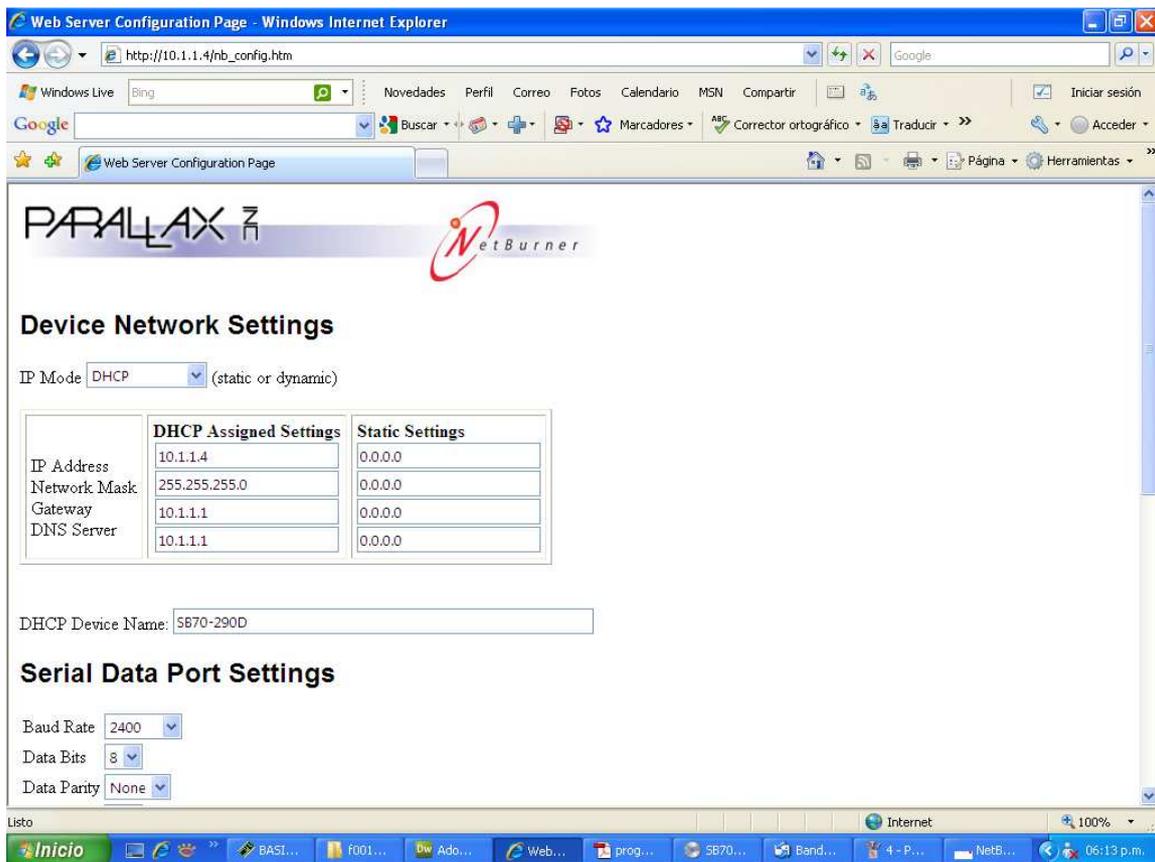
tener acceso a la configuración de red, de puertos, diagnóstico y funciones FTP dando click en los links que proporciona la página.

La página de fábrica lista cinco rápidos (Figura 27) pasos para acceder a las configuraciones.



**Figura 27: Pasos rápidos para la configuración**

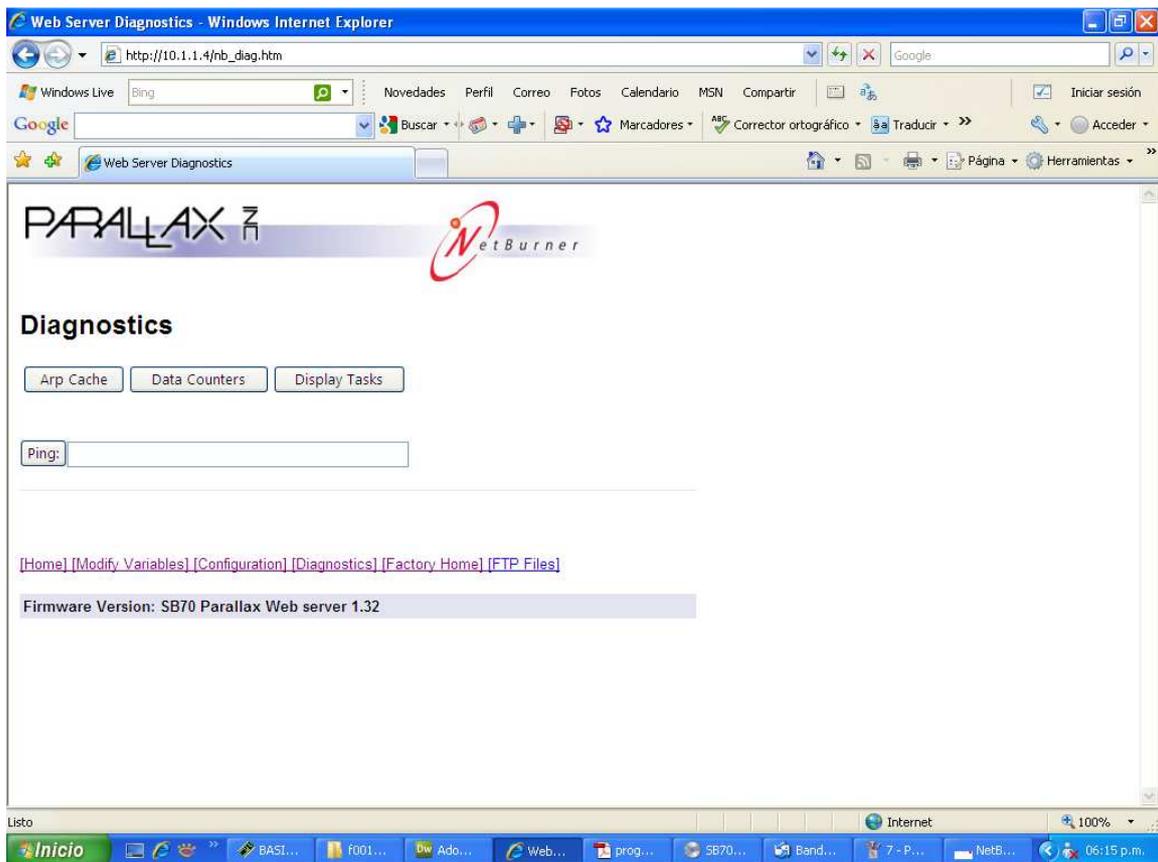
Cuando se ubica en la página de configuración (Figura 28) se debe asegurar que el IP Mode (estático y dinámico) debe ser igual a la opción establecida al inicio en el programa IPSETUP. La información de la dirección se desplegará dependiendo de la opción escogida.



**Figura 28: Página de configuración**

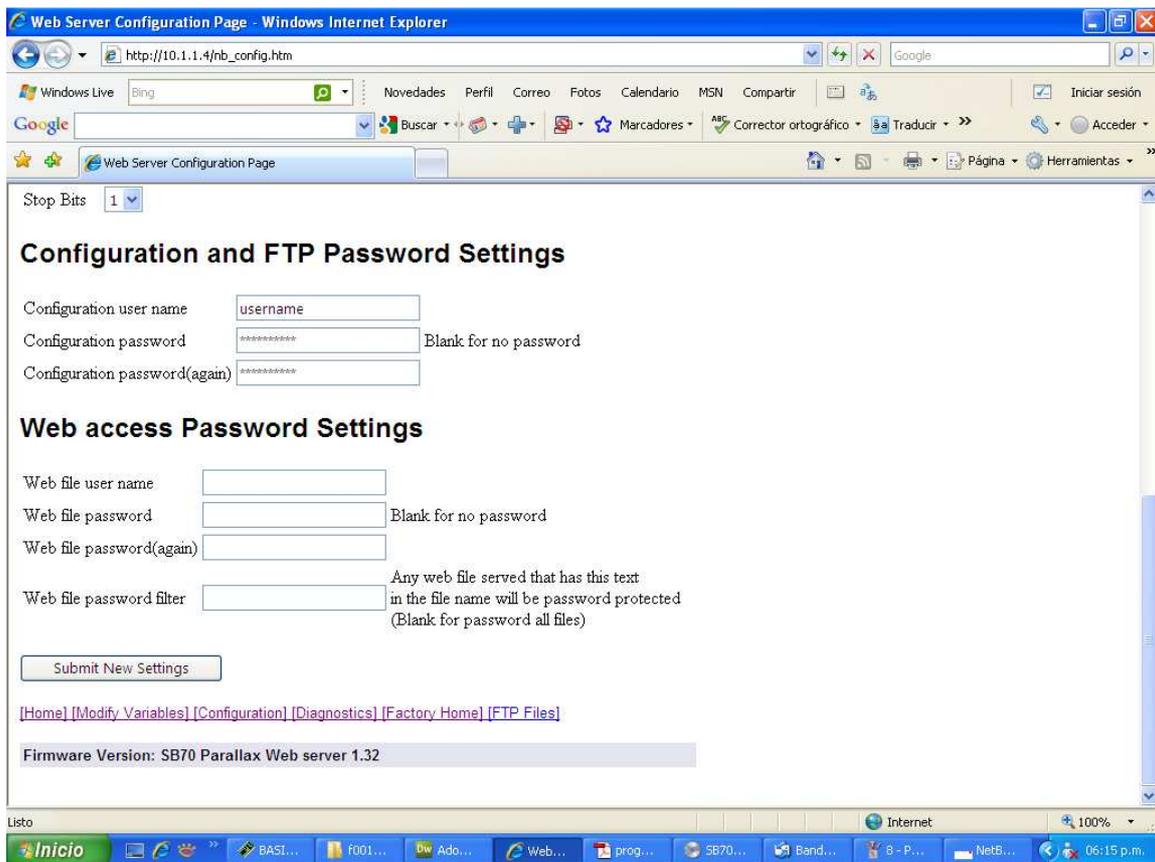
Ahora se debe establecer el Baud Rate para la comunicación serial. Para la mayoría de los BS2 se usa un Baud Rate de 2400. Hay excepciones para los BS2sx, BS2p y BS2px en los que se usa 9600bps. Data Bits debe estar establecido en 8, Data Parity en None y Stop Bits en 1. Se debe dar click en Submit New Settings para guardar los cambios hechos.

En este momento se puede correr el programa de prueba PINKTestV1.0.BS2 que está dentro de la página por defecto (Figura 29) para probar la comunicación serial con el módulo PINK.



**Figura 29: Página de diagnóstico o prueba**

Configurando las contraseñas se previene conexiones FTP anónimas y también protegen las configuraciones. La contraseña se deberá ubicar en dos ocasiones para verificación y prevención de accidentes al momento de digitar la clave (Figura 30).



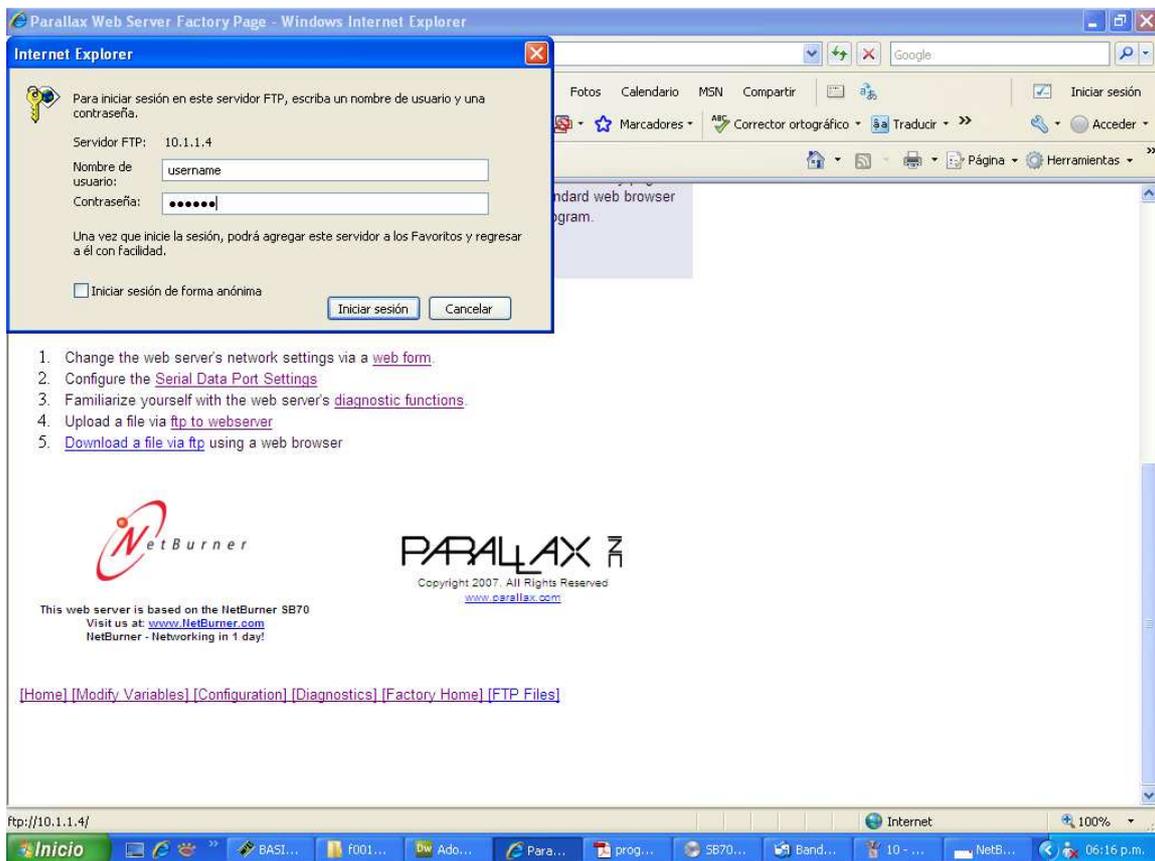
**Figura 30: Página para establecer contraseñas**

Activando la opción de contraseña para acceso a la web, se pueden proteger archivos que contengan la clave filtro. Por ejemplo, si la palabra filtro es “lock”, todos los archivos que contengan esa palabra como “lockest.htm” o “iconlock.jpg” solicitarán el nombre de usuario y la clave de acceso para ser abiertos.

- **Páginas del usuario**

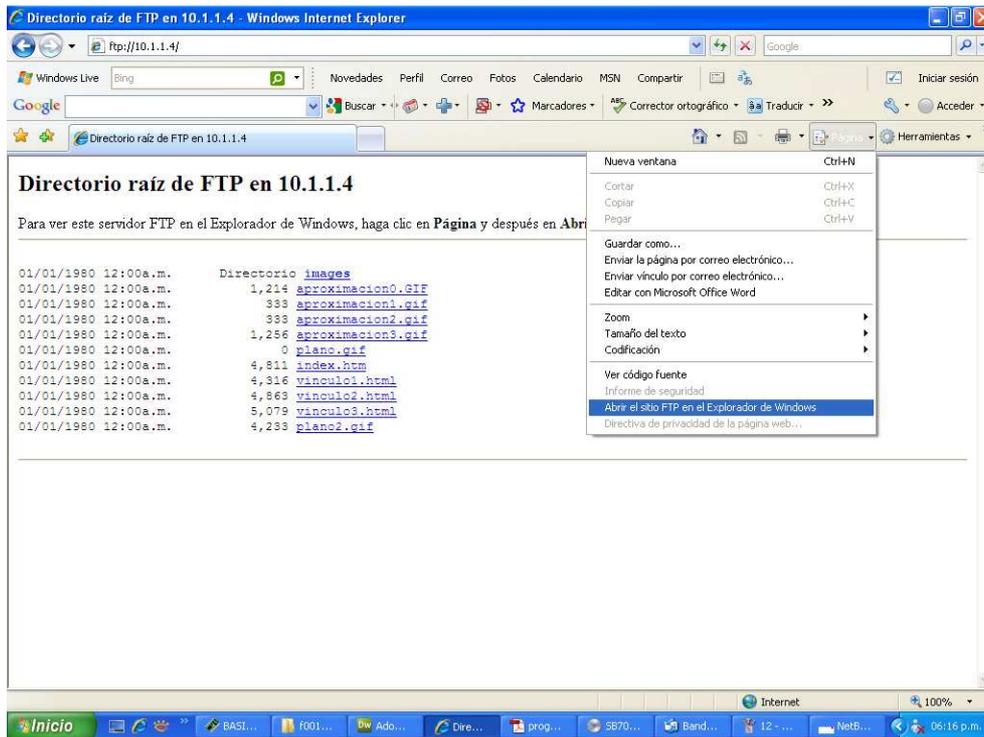
Las páginas del usuario pueden ser cargadas en el módulo PINK vía FTP (Figura 31). Definiendo la dirección ftp://XXX.XXX.XXX.XXX (donde XXX.XXX.XXX.XXX es la dirección IP del PINK) en una ventana del Internet Explorer se abrirá una ventana FTP en

dónde utilizando el conocido drag-and-drop (arrastrar y soltar) (Figura 34) se podrán cargar las páginas y archivos que el usuario desea ubicar en la memoria flash del módulo. También se pueden borrar o renombrar archivos a través de ésta pantalla.

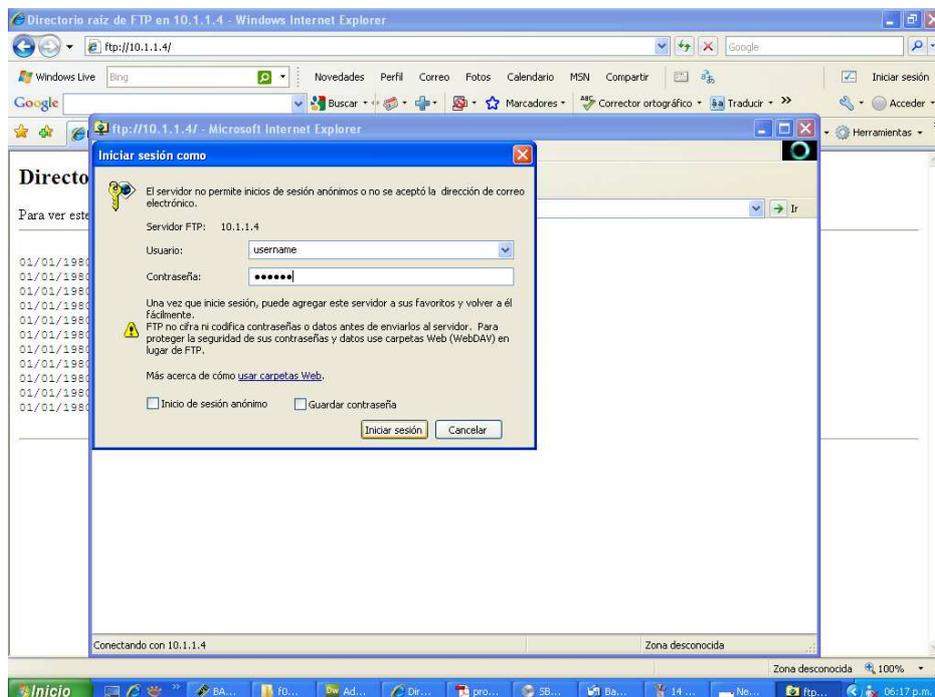


**Figura 31: Página para carga de archivos vía FTP**

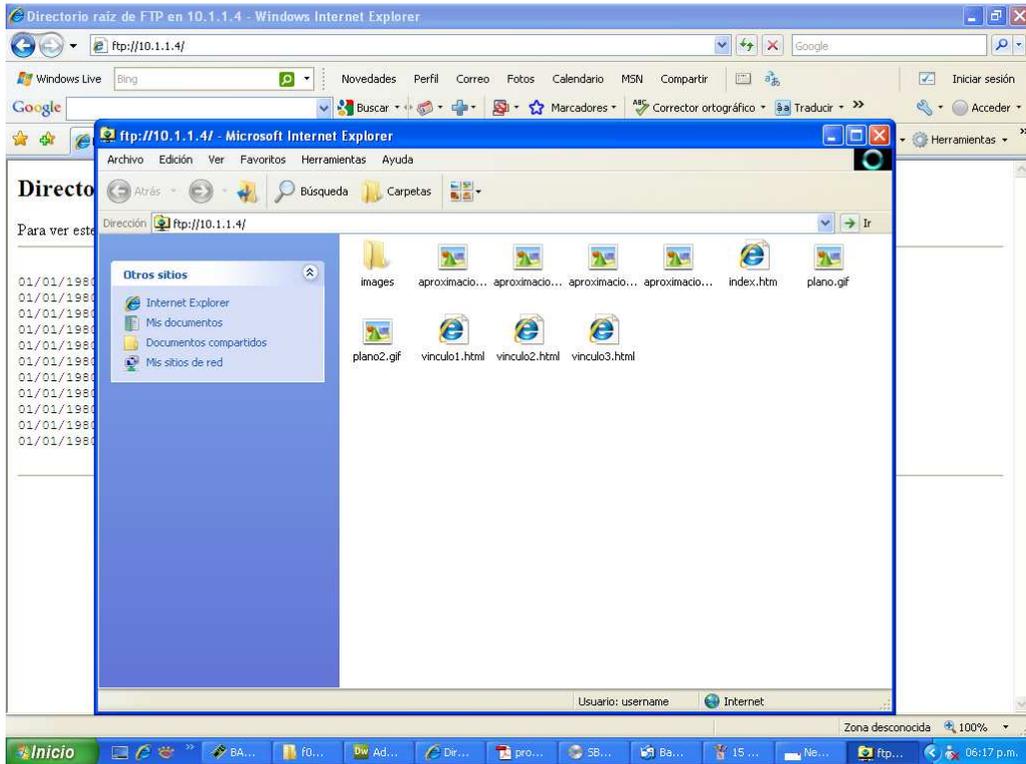
Internet Explorer 7 requiere que la ventana FTP sea abierta por separado, pero brinda las instrucciones necesarias para usarla (Figuras 32 y 33). Algunos exploradores no soportan la opción de arrastrar y soltar por lo que se necesitará un programa FTP cliente.



**Figura 32: Guía para almacenamiento vía FTP**



**Figura 33: Guía para almacenamiento vía FTP**



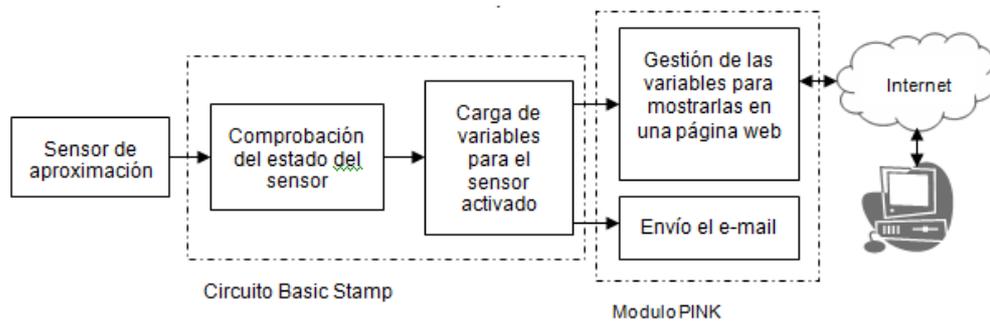
**Figura 34: Arrastrar y soltar para almacenamiento de archivos vía FTP**

Una recomendación es que al momento de cargar las páginas en el módulo la principal debe tener el nombre de index.htm ya que el módulo PINK siempre buscará este archivo para abrirlo y si no lo encuentra se desplegará la página de fábrica.

## 6.2.2 CONFIGURACION DEL BASIC STAMP 2

- **Esquema General**

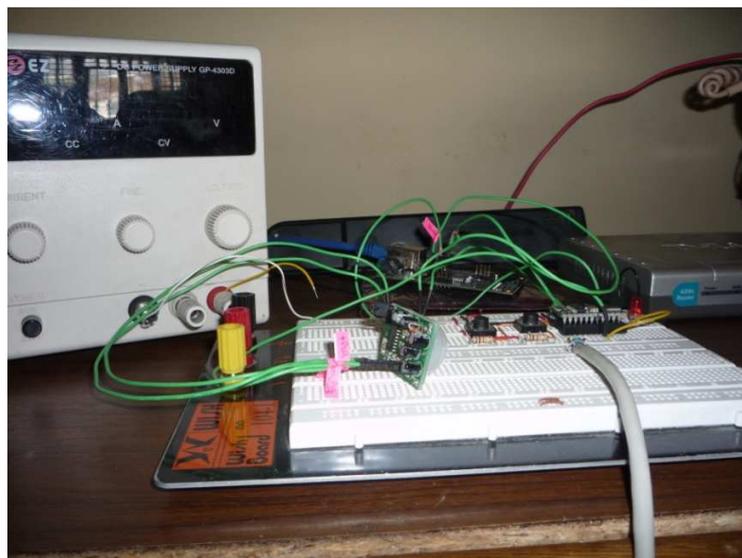
El sistema de alarma conectado a Internet que a continuación se diseña sigue el siguiente esquema:



**Figura 35: Esquema general del sistema de alarma**

○ **El Circuito Basic Stamp 2**

Como se puede ver en la imagen, inicialmente se ha conectado el sensor con cable corto (Figura 36) para realizar las primeras pruebas y ajustes al programa.



**Figura 36: Conexiones iniciales**

○ **El Programa para el BS**

Una vez realizado el conexionado de todos los elementos, se diseñó el programa que en el BS controlará el sensor y enviará datos al módulo PINK.

Al comienzo del programa, se declararon las variables que se van a usar, el diseño del mismo es apropiado para cuando el usuario se ausenta de la vivienda ya que el disparo del sensor provocará el envío de un email notificando la alerta.

La desactivación del sistema se podrá llevar a cabo mediante el botón RESET del circuito, lo que permite devolver el control del programa al inicio.

```
' {$STAMP BS2}
' {$PBASIC 2.5}
NBVAR VAR Byte
n VAR Byte
email VAR Byte
cont3 VAR Word
time VAR Word
SEROUT 8,396,["!NB0W03:0",CLS] 'inicializa
DO WHILE modo = 0
  IF IN14 = 1 THEN 'modo [alerta durante ausencia]
    modo = 1
    FOR N = 1 TO 10 'damos tiempo para abandonar la casa
      FREQOUT 4, 250, 3000 ' beep
      PAUSE 750
    NEXT
  ENDIF
```

Ya que el programa trabaja en modo "alarma", se dispuso de un tiempo suficiente para abandonar la casa; por esta razón, se incluye una instrucción FOR-NEXT (ajustado para contar 10seg. aproximadamente) antes de que el programa comience a comprobar el estado del sensor.

A continuación se tiene la parte del programa que corresponde a la comprobación del sensor:

```
DO
  GOSUB PIR
LOOP
```

Se debe notar que lo que se hace es disponer de una serie de llamadas a la subrutina para la continua comprobación del estado del sensor.

A continuación se escribe la subrutina que corresponde al sensor PIR:

```
PIR:
  IF IN3 = 1 THEN
    cont3 = 1 'OJO alerta aproximación
    GOSUB EMAIL_BEEP 'envía email o beep
  ENDIF
  IF cont3 > 0 THEN
    SELECT cont3
      CASE > 800
        cont3 = 0 ` vuelve a estado alerta
        SEROUT 8,396,["!NB0W03:0",CLS] 'inicializa después
de 120 segundos
        RETURN
      CASE > 400
        SEROUT 8,396,["!NB0W03:3",CLS] 'sensor ha sido
activado hace más de 60 segundos-> mostrar huella
      CASE > 200
        SEROUT 8,396,["!NB0W03:2",CLS] 'sensor ha sido
activado hace más de 30 segundos-> mostrar huella
      CASE > 0
        SEROUT 8,396,["!NB0W03:1",CLS] 'sensor ha sido
activado
    ENDSELECT
    cont3 = cont3 +1 'contador en marcha (aunque no exista
aproximación)
```

```
    PAUSE 10
ENDIF
RETURN
```

Con la variable contador "contx" se controla el paso del tiempo para ir cambiando las diferentes imágenes del sensor activado.

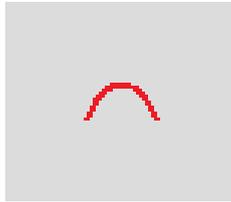
Las siguientes líneas de código son en las que se asignan diferentes valores a las variables que pasarán al módulo PINK: el programa en el BS2 envía serialmente el comando: !NBOWxx donde "xx" es el número de la variable que se está usando para el sensor.

Pero ¿por qué asignar varios valores para un único estado activo del sensor? Supongamos el caso el que el sensor se activa un solo instante, aún cuando se esté observando en ese momento la página donde es posible monitorear el comportamiento del sistema, puede que el usuario no se percate de que el sensor ha estado activo durante un instante. De manera que se ha buscado la manera en que la huella de la actuación del sensor permanezca activa hasta que pueda ser vista por el usuario.

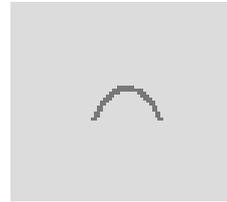
Para lograr ese modo, se ha incluido tres lecturas del gráfico del sensor activado: una que nos indica que en ese momento el sensor está actuando (durante el tiempo de su activación, que en algunos casos podrá ser un solo instante); otra en la que el sensor ha actuado recientemente (durante unos minutos después de su activación); y otra, después de una hora desde su activación. Finalmente, al cabo de esa hora, el gráfico representativo, la huella, "se desvanecerá".

- ✓ Cuando existe intrusión en ese momento. (Figura 37)
  
- ✓ Transcurridos 30 segundos desde que se activó el sensor. (Figura 38)

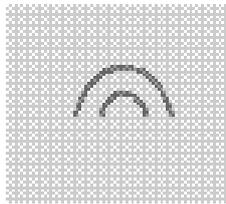
- ✓ Transcurridos 60 segundos desde que se activó el sensor. (Figura 39)



**Figura 37: Activación en ese momento**



**Figura 38: Activación hace 30 segundos**



**Figura 39: Activación hace 60 segundos**

Los tiempos que la "huella" permanece se pueden configurar cambiando los valores que se van comprobando con los SELECT. Con este uso, lo que se consigue es un sistema de alertas más eficaz, gráfico e informativo.

Pero, no sólo se va a pasar al PINK los valores que corresponden al estado de los sensores: veamos ahora cómo se puede "construir" el email que se enviará en modo "alerta", cuando se dispare el sensor de aproximación. Deben ser especificadas las variables de los registros: Nb\_varET (dirección email destinatario), Nb\_varEF (dirección email remitente), Nb\_varES (asunto del mensaje), Nb\_varEC (contenido del mensaje) y Nb\_varEV (servidor SMTP que no requiera autenticación):

EMAIL\_BEEP:

```
IF modo = 1 THEN FREQOUT 4, 250, 3000 ' beep
```

```
IF cont3 > 1 THEN 'envía email sólo en caso de que se detecte  
un intruso en la casa
```

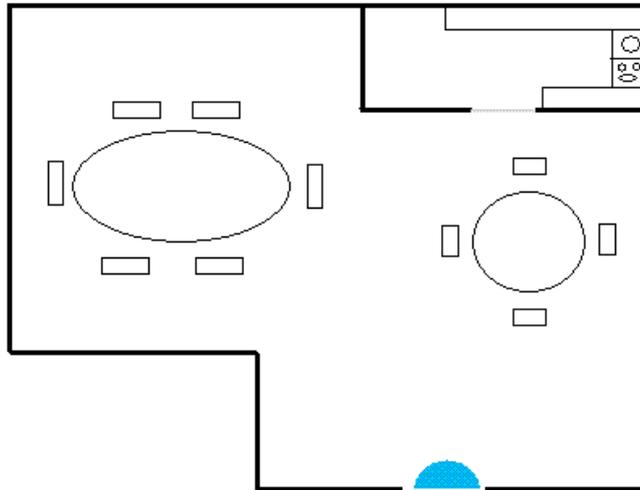
```
IF email = 0 THEN
    SEROUT 8,396,["!NB0WET:irma@tesis.com",CLS]
    SEROUT 8,396,["!NB0WEF:administrador@tesis.com",CLS]
    SEROUT 8,396,["!NB0WES:Mensaje de Alerta",CLS]
    SEROUT 8,396,["!NB0WEC:Alerta, intruso en la casa",CLS]
    SEROUT 8,396,["!NB0WEV:10.1.1.5",CLS]
    SEROUT 8,396,["!NB0SM"]
    email = 1 'email ya enviado
ENDIF
ENDIF
RETURN
```

### **6.2.3 CONSTRUCCIÓN DE LA PÁGINA WEB**

En este punto se creó una página web para poder mostrar los diferentes estados del sensor de aproximación.

Primero se debe preparar con cualquier programa de dibujo una imagen que represente el plano de la vivienda (Figura 40) donde se va a instalar el sistema, con trazos simples y un solo color a fin de que destaque cualquier otra señalización que se haga sobre el plano.

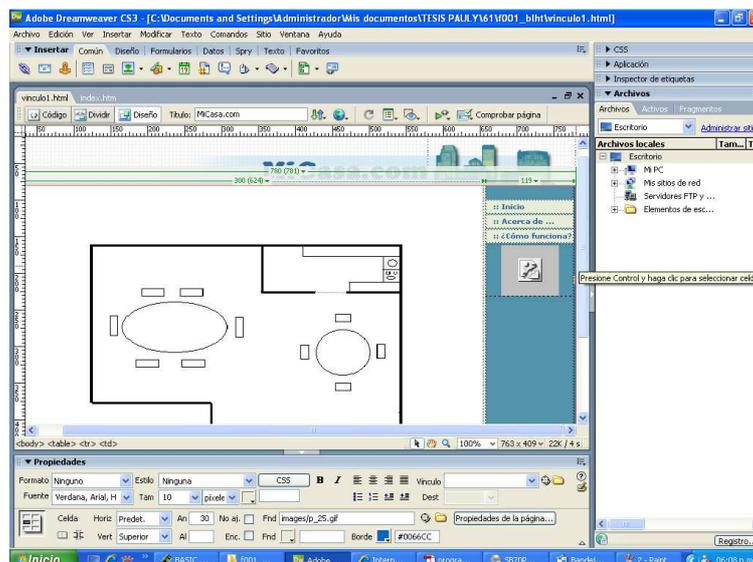
En el caso de este proyecto el gráfico es:



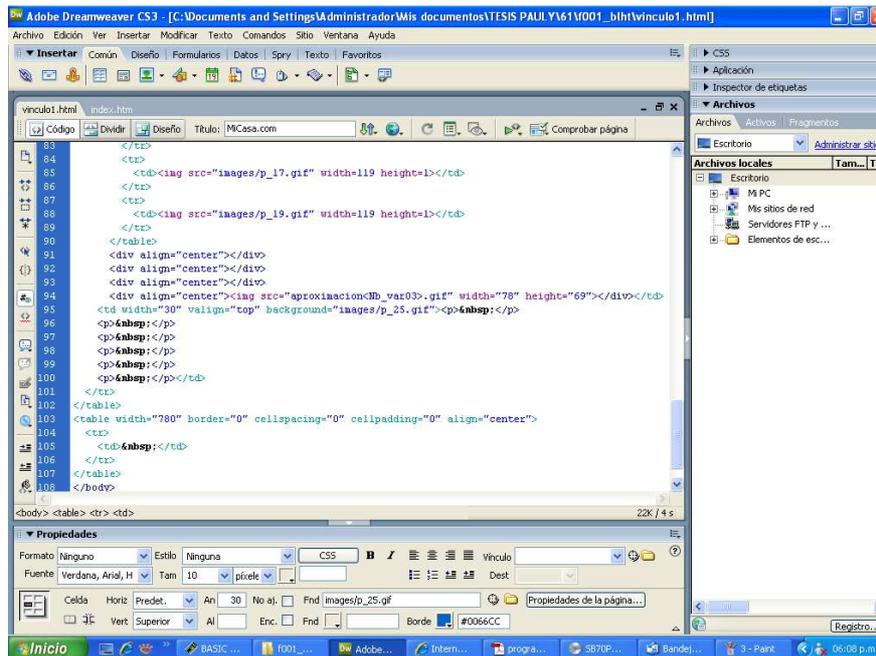
**Figura 40: Plano representativo de la vivienda**

En el plano se ha incluido una referencia en color azul que corresponde a la ubicación física del sensor de aproximación, la imagen se guarda con nombre y formato "plano.gif".

El código HTML para mostrar esa imagen en la página (Figuras 41 y 42), que fue guardada con nombre y formato "micasa.htm", será:



**Figura 41: Inclusión de la imagen del plano en la página web**



**Figura 42: Código para mostrar la imagen del plano en la página web**

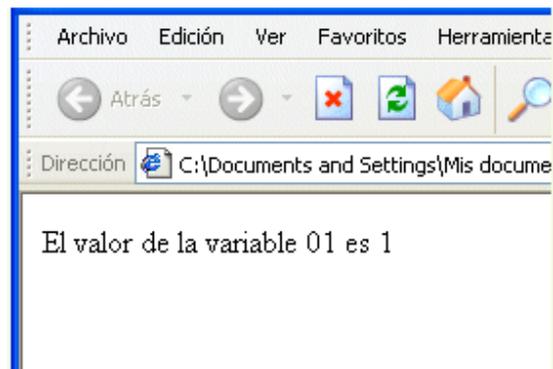
```
<meta http-equiv="refresh" content="20"; URL="micasa.htm">
  <td width="300" valign="top" background="images/p_22.gif"><div
align="center"></div></td>
```

Fijese en la línea `<meta http-equiv="refresh" content="20"; URL="micasa.htm">`, esta etiqueta HTML permitirá que la página se recargue automáticamente cada 5 seg. con el fin de que muestre información actualizada constantemente.

Para que la página web "lea" los valores de las variables que el circuito BS2 envía al módulo PINK es tan sencillo como, por ejemplo, emplear este tipo de código:

*El valor de la variable 01 es <Nb\_var01>*

Si el valor de esa variable tuviera el valor "1" (porque está activado), se mostraría una página como la figura 43.



**Figura 43: Página que muestra el valor de la variable 1**

Pero lo interesante va a ser representar esos valores gráficamente. ¿Qué pasaría si en vez de la línea de código anterior se intenta cargar una imagen con esta otra línea?

*El valor de la variable 01 es *

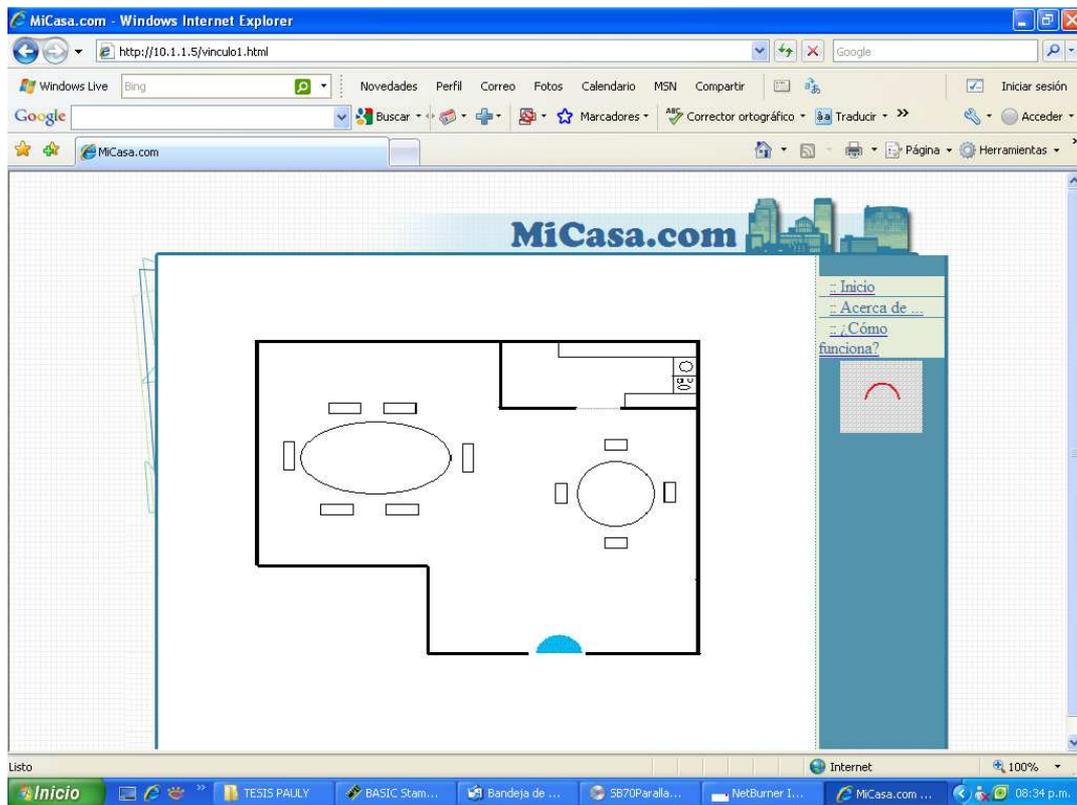
Si el valor de la variable Nb\_var01 es "0", la página que contenga esa línea mostrará la imagen "aproximacion0.gif"; en cambio, si el valor es "1", la página mostrará la imagen "aproximacion1.gif". De esta forma se pueden probar los valores de las variables que entregue nuestro circuito BS al módulo PINK para que en determinados casos se muestre, en la página web, la imagen que se desee.

Así pues los nombres de los archivos de las imágenes del sensor deberán acabar en "1" las que corresponden a su estado activo; acabarán en "2" las que correspondan a su huella más reciente; y acabarán en "3" las que correspondan a la última huella (en el caso de este proyecto: aproximacion1.gif, aproximacion2.gif y aproximacion3.gif)

También se preparó una imagen gif transparente que se guardó con los mismos nombres que las imágenes anteriores, pero acabado en "0": aproximacion0.gif. Esta imagen servirá para cuando el valor de la variable sea "0"; es decir, cuando el sensor esté en reposo.

Ahora ha llegado el momento de colocar las imágenes en la página web.

Así pues queda programada la página web para que se pueda detectar la huella del sensor como se muestra en la figura 44:



**Figura 44: Página web que muestra la activación del sensor**

La carga de la imagen se realiza mediante la etiqueta "<img src>" al igual que se hizo con la imagen del plano. De esta forma se concluye con el diseño de la página web resultando en las figuras que se muestran a continuación.

#### **6.2.4 CONFIGURACION DEL SERVIDOR SMTP**

Un servidor de correo es una aplicación informática cuya función es parecida al Correo postal solo que en este caso los correos (otras veces llamados mensajes) que circulan, lo hacen a través de las Redes de transmisión de datos y a diferencia del correo postal, por

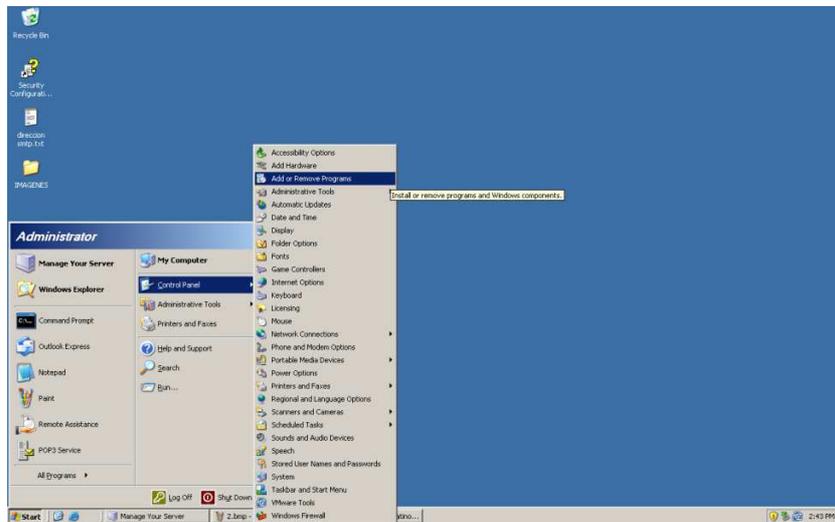
este medio solo se pueden enviar adjuntos de ficheros de cualquier extensión y no bultos o paquetes al viajar la información en formato electrónico.

### ¿Qué es un servidor de correo electrónico Windows Server 2003?

Un servidor de correo electrónico en Windows Server 2003 tiene instalado los servicios de correo electrónico (E-mail) y sirve para proporcionar a los usuarios el envío y la recepción de correo electrónico. Esos servicios incluyen el servicio Protocolo de oficina de correo 3 (POP3) y el servicio Protocolo simple de transferencia de correo (SMTP), que recuperan y transfieren correo electrónico, respectivamente.

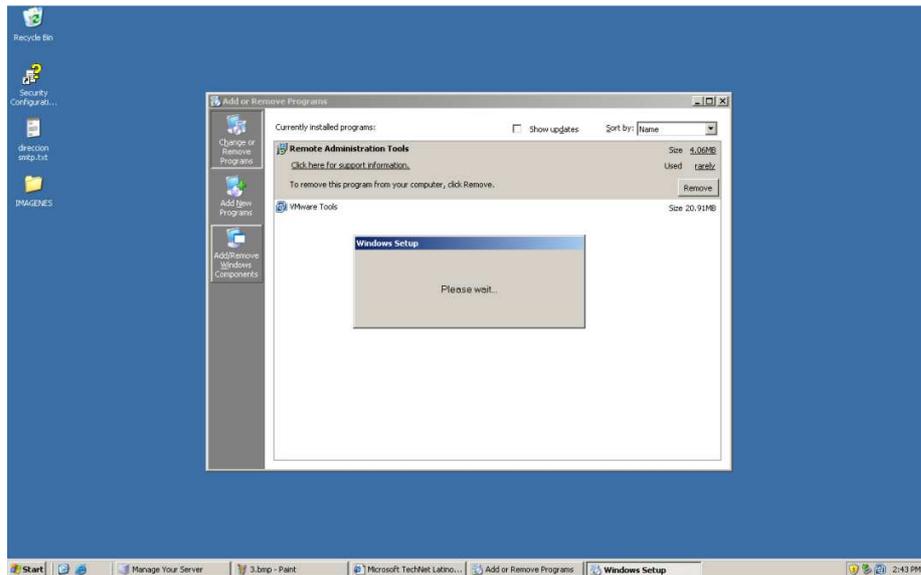
### Instalación de servicios de correo electrónico.

- Hacer clic en Inicio, luego clic en Panel de control y, a continuación, hacer clic en Agregar o quitar programas (Figura 45).



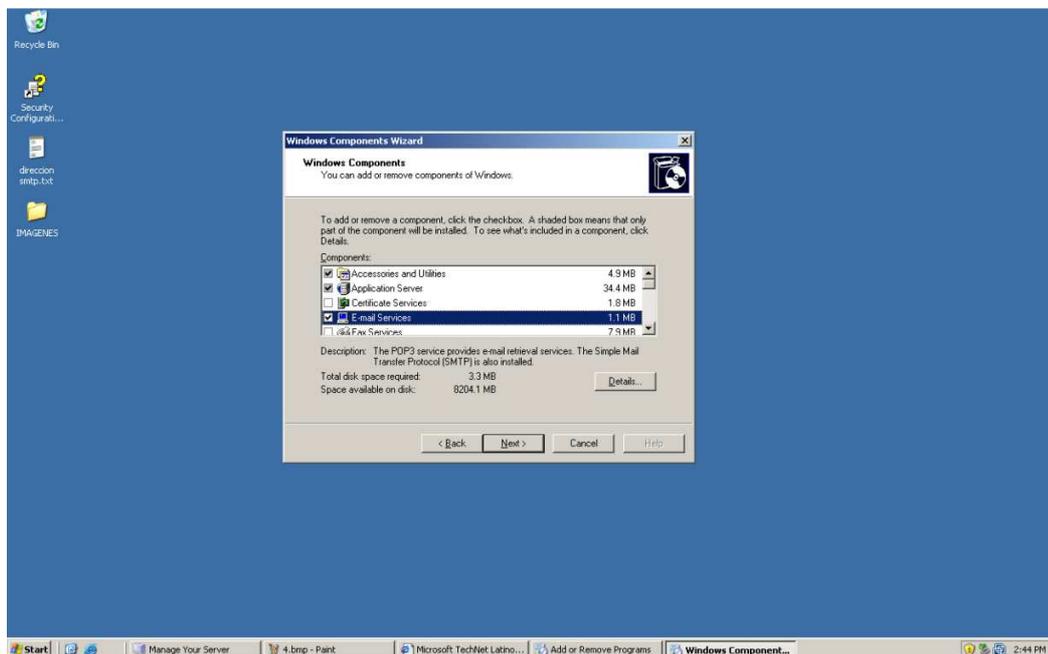
**Figura 45: Pasos para la configuración del servidor SMTP**

- Una vez abierto, ver la solapa izquierda inferior y hacer clic en Agregar o quitar componentes de Windows (Figura 46).



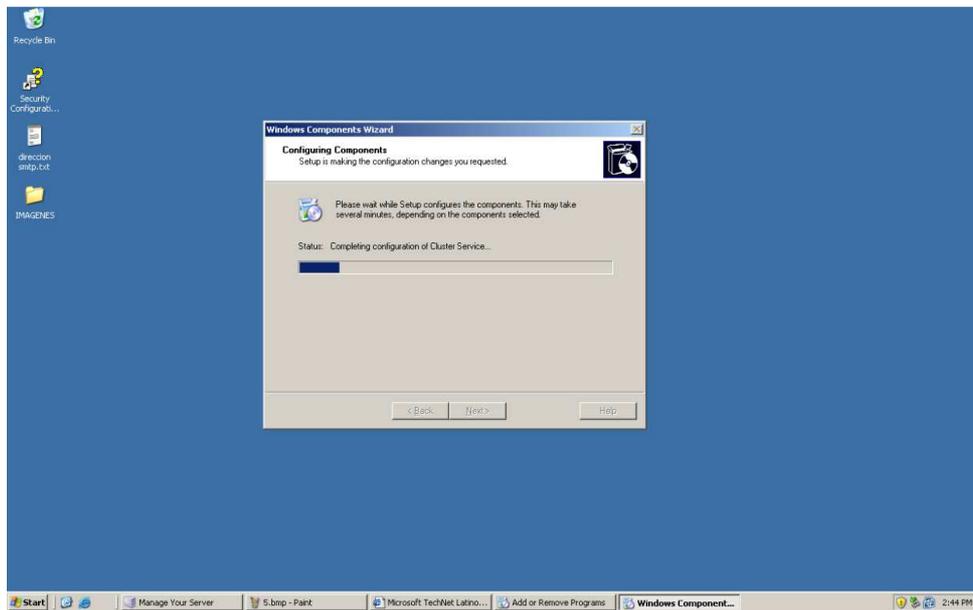
**Figura 46: Pasos para la configuración del servidor SMTP**

- o Activar la casilla de verificación Servicios de correo electrónico, hacer clic en Siguiente (Figura 47).



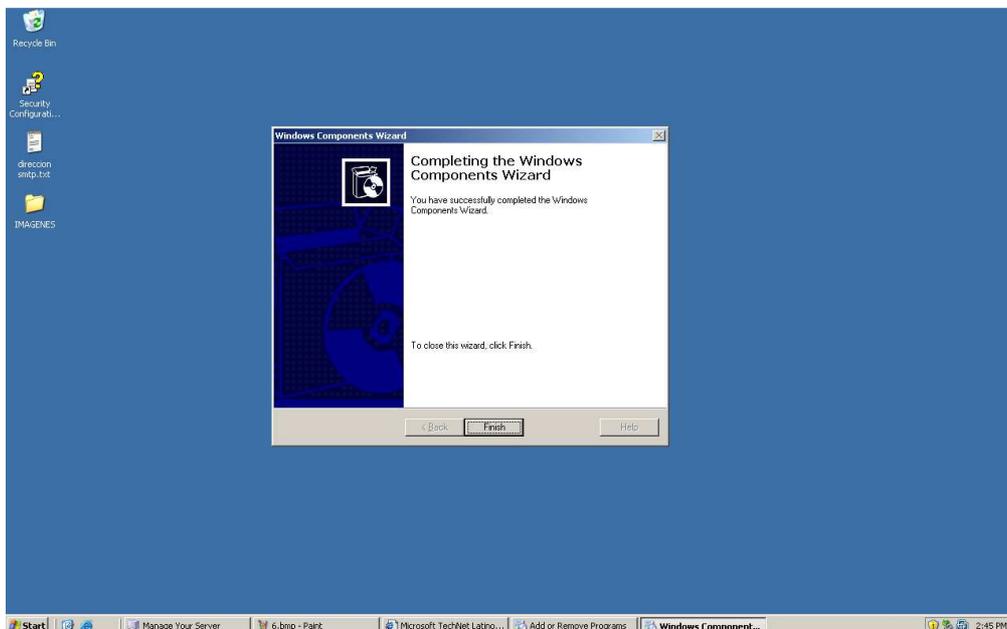
**Figura 47: Pasos para la configuración del servidor SMTP**

- Hacer clic en Siguiente y, a continuación, completar el asistente (Figura 48).



**Figura 48: Pasos para la configuración del servidor SMTP**

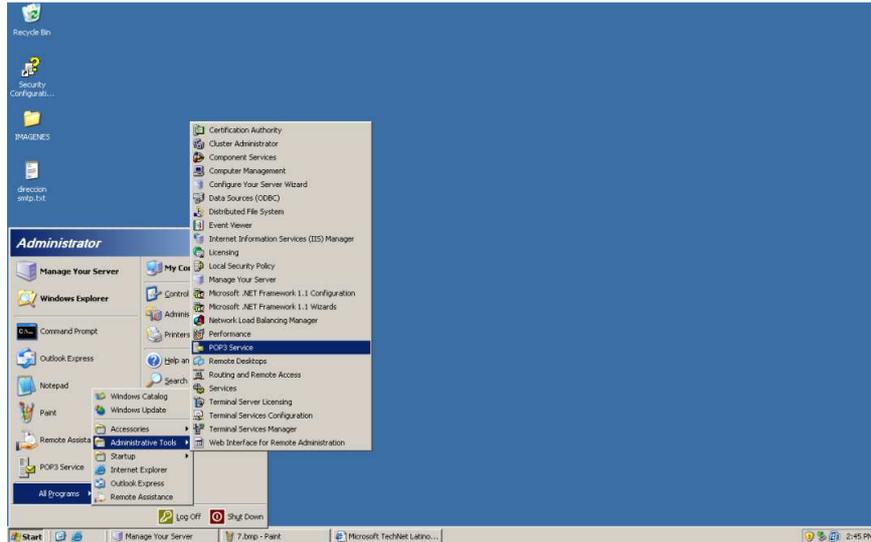
- Finalización del Asistente para componentes de Windows hacer clic en Finalizar, luego cerrar la pantalla Agregar o quitar programas (Figura 49).



**Figura 49: Pasos para la configuración del servidor SMTP**

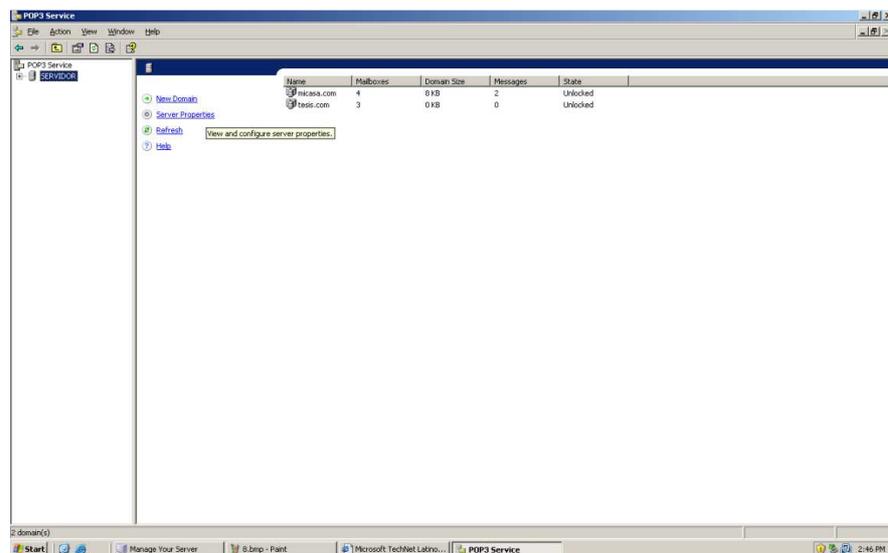
Para configurar los servicios de correo electrónico con cuentas locales de Windows:

- Hacer clic en el botón Inicio, seleccionar Todos los programas, Herramientas administrativas y, después, hacer clic en Servicio POP3 (Figura 50).



**Figura 50: Configuración de servicios de correo electrónico**

- En el árbol de la izquierda bajo Servicio POP3, hacer clic en nombre del servidor, Hacer clic en el vínculo Propiedades del servidor (Figura 51).

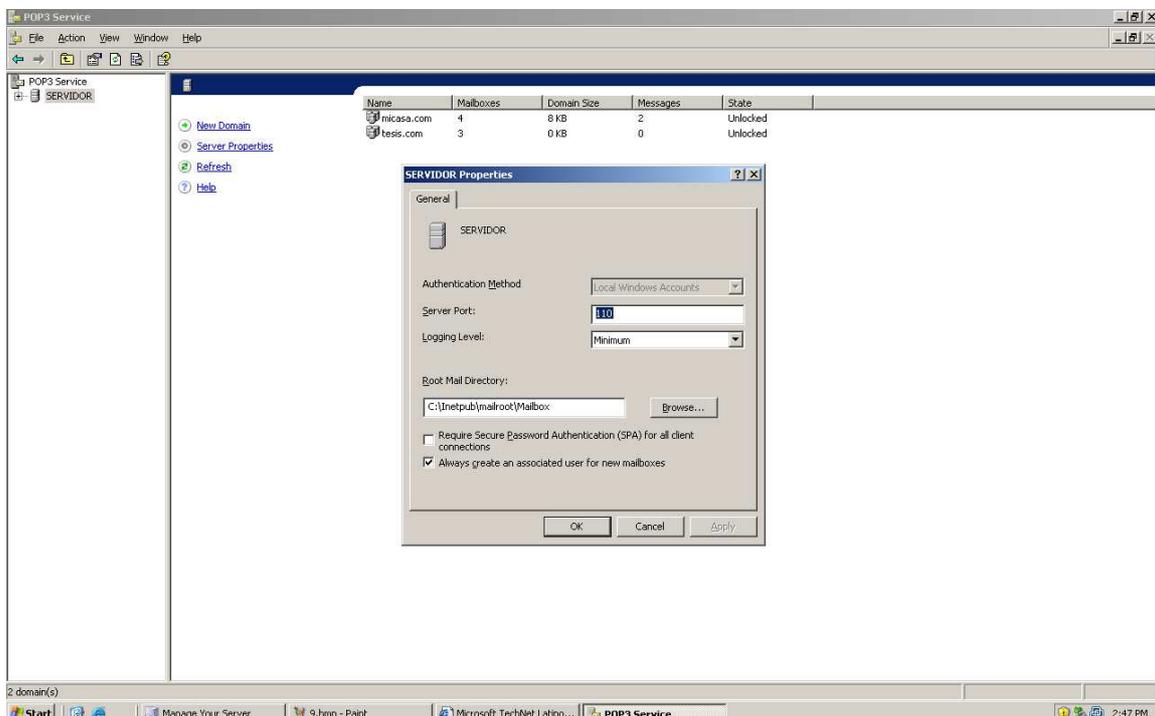


**Figura 51: Configuración de servicios de correo electrónico**

Dependiendo del modo que se haya instalado el servidor Windows Server 2003, y el modo que se quiera autenticar, tendrá las siguientes opciones:

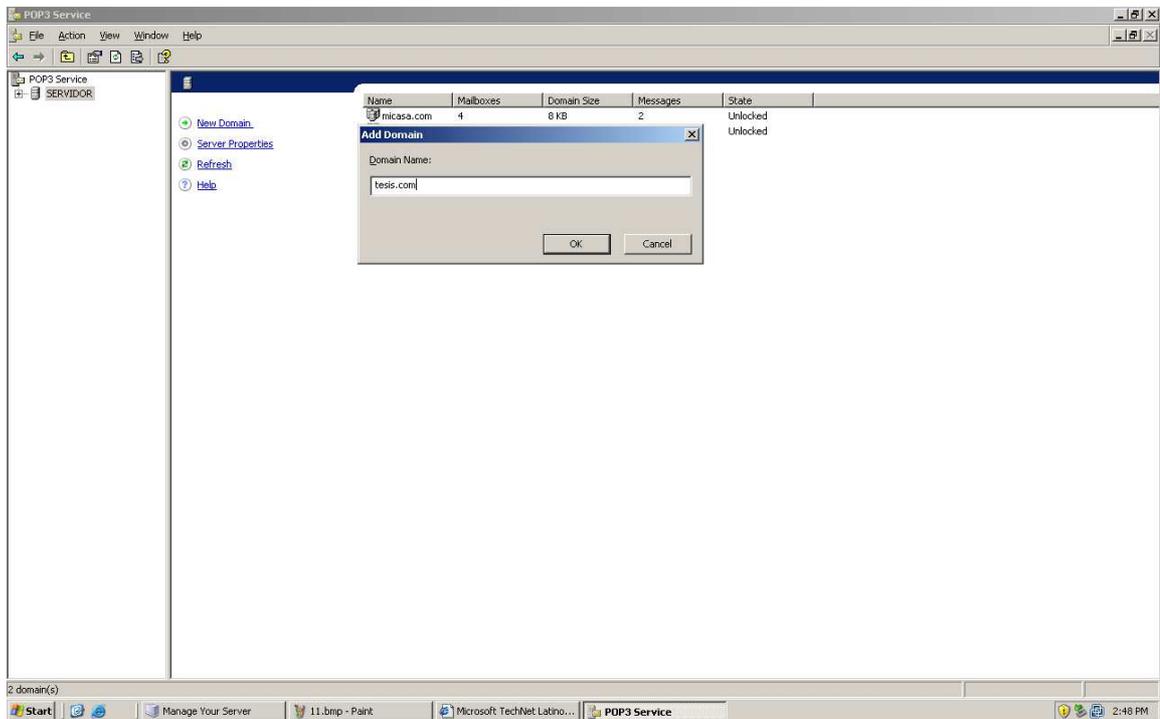
- Cuentas Locales de Windows locales: El servidor de correo no debe ser un servidor miembro de Active Directory y ni debe almacenar cuentas de usuario en el servidor en el que está instalado el servicio POP3.
- Integrado en Active Directory: El servidor de correo debe ser un controlador de dominio o un servidor miembro.
- Archivo de contraseña cifrado: El servidor de correo no tiene que estar utilizando Active Directory o no debe tener cuentas de usuario para el servicio POP3 en el equipo local.

En ese caso seleccionar Cuentas Locales de Windows, hay que elegir de la lista de Método de autenticación, luego hacer clic en Aceptar (Figura 52).



**Figura 52: Configuración de cuentas locales**

- Hacer clic en el vínculo Dominio nuevo, escribir el Nombre de dominio y, a continuación hacer clic en Aceptar (Figura 53).

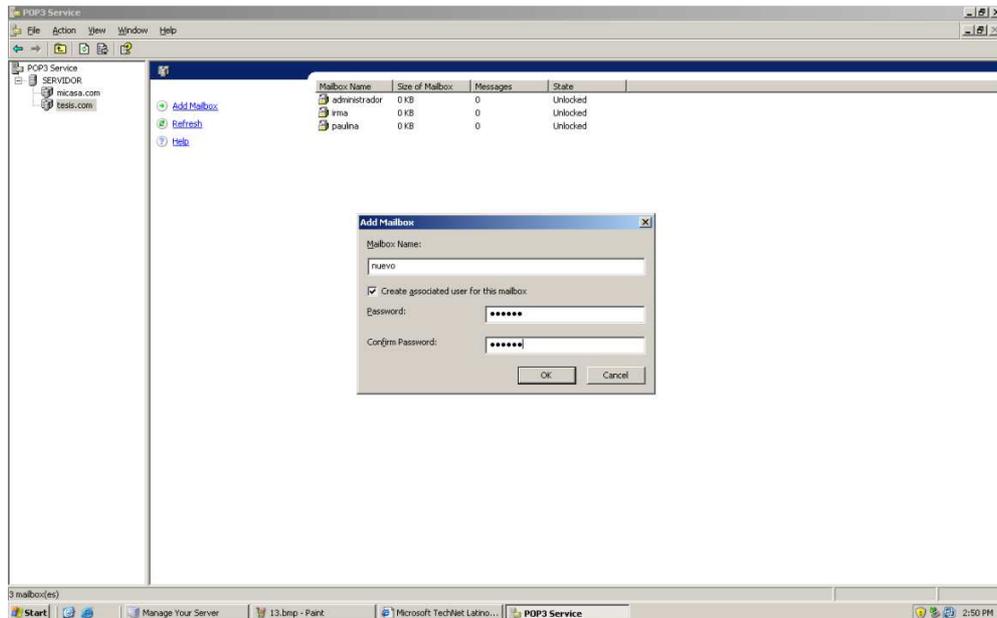


**Figura 53: Configuración de cuentas locales**

En el panel de la derecha, hacer doble clic en nombre del dominio. El árbol del panel de la izquierda se expandirá bajo nombre del servidor y mostrará el dominio. Hacer clic en el vínculo Agregar buzón.

- En la pantalla Agregar buzón, escribir un nombre para Nombre del buzón, que será la dirección de correo electrónico para la que se deberá utilizar algún tipo de nomenclatura. Luego, es necesario dejar activada la casilla de verificación Crear un usuario asociado para este buzón, colocar una contraseña donde dice Contraseña: y

repetirla donde dice Confirmar contraseña: hacer clic en Aceptar y luego otra vez en Aceptar cuando aparezca la pantalla de confirmación de creación de la cuenta Servicio POP3 (Figura 54) .

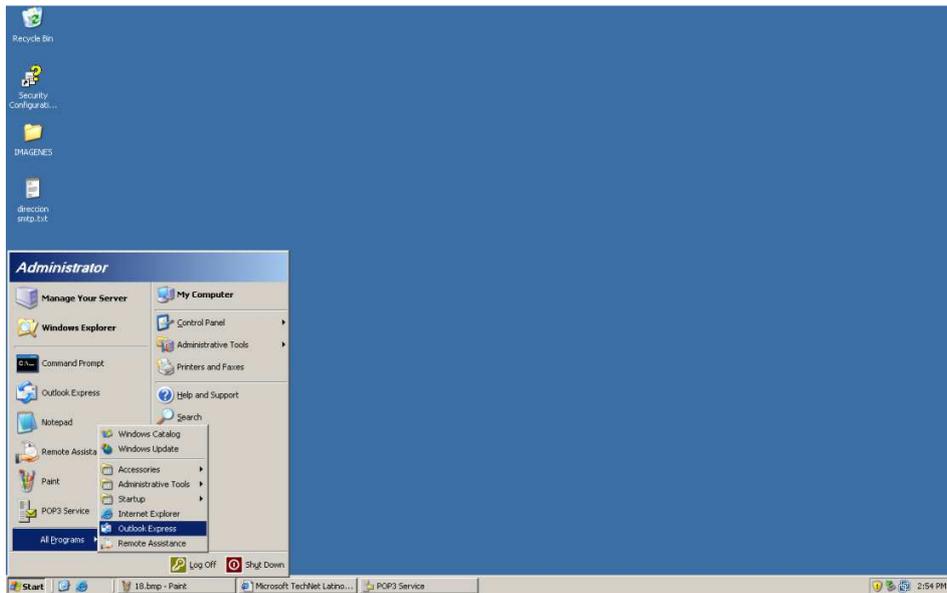


**Figura 54: Agregar nuevo buzón**

- Posteriormente, hacer clic de nuevo en el vínculo Agregar buzón y repetir los pasos necesarios para crear más buzones.

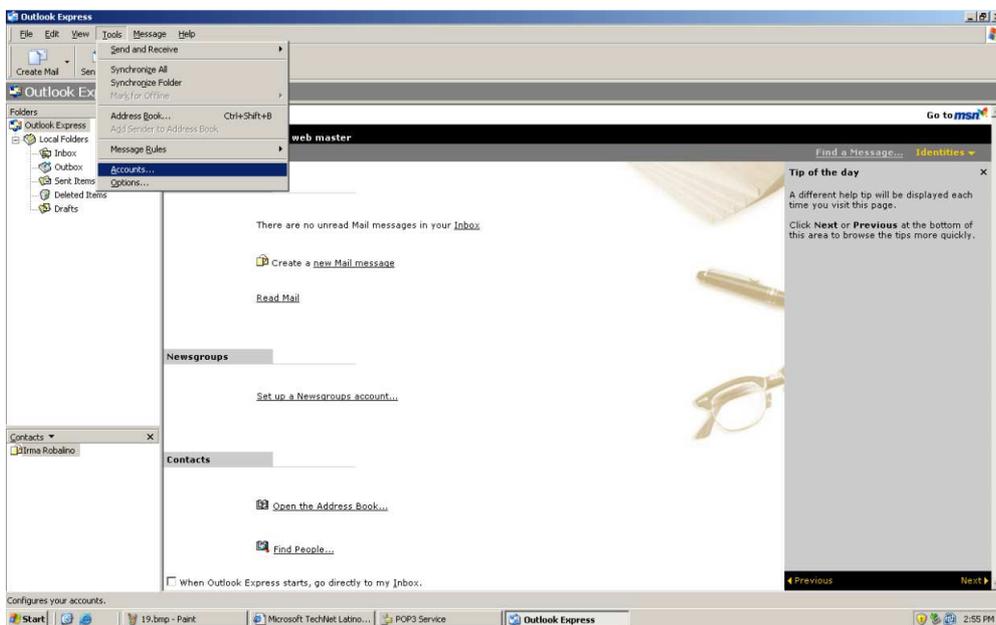
### **Configurar Outlook Express para los servicios de correo electrónico.**

- Hacer clic en Inicio, seleccionar Todos los programas y, a continuación, hacer clic en Outlook Express (Figura 55).

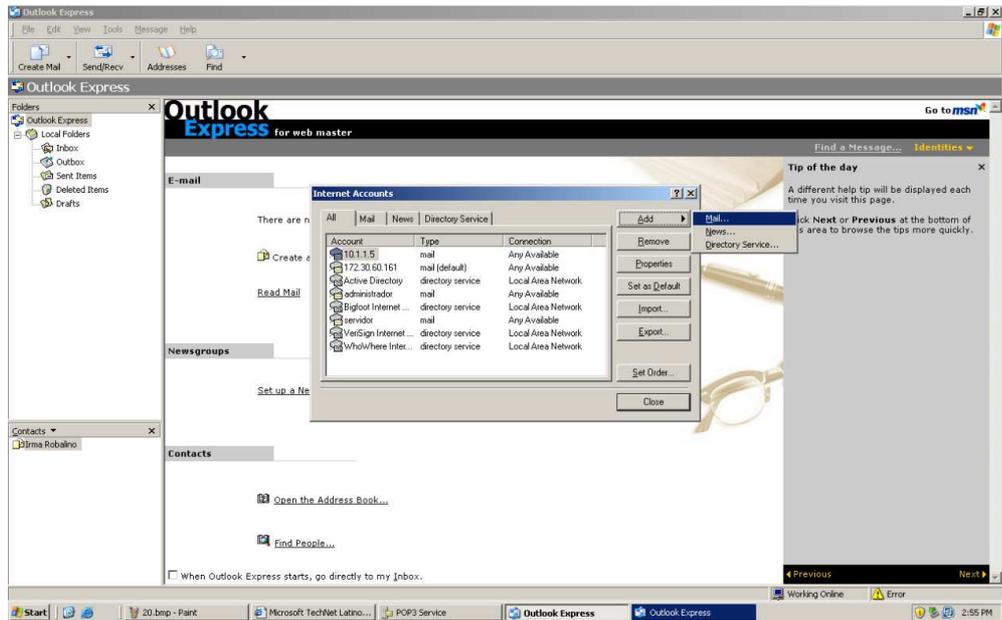


**Figura 55: Configuración de Outlook**

- En el menú Herramientas, hacer clic en Cuentas. Hacer clic en Agregar y, a continuación, en Correo (Figuras 56 y 57).

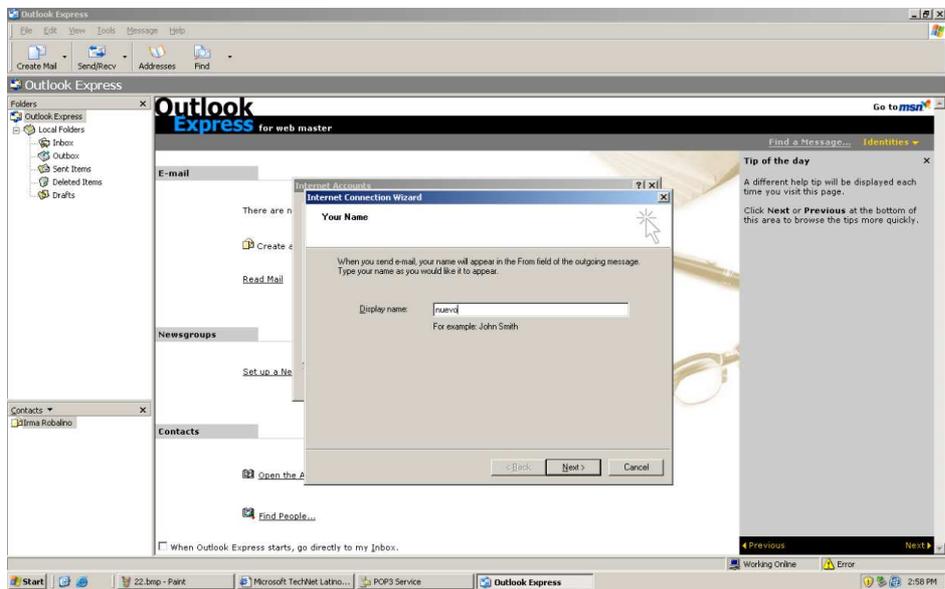


**Figura 56: Configuración de Outlook**



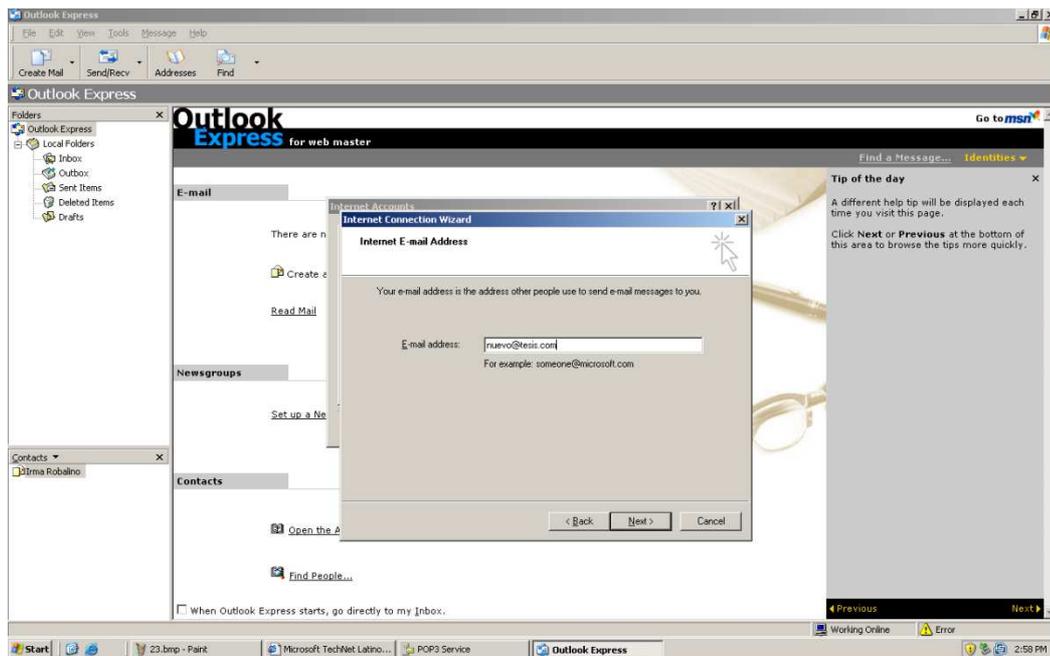
**Figura 57: Configuración de Outlook**

- o Colocar el nombre ( Display Name ) o la forma como se desea que aparezca el mismo en el campo DE en los correos salientes. Hacer clic en Siguiete (Figura 58).



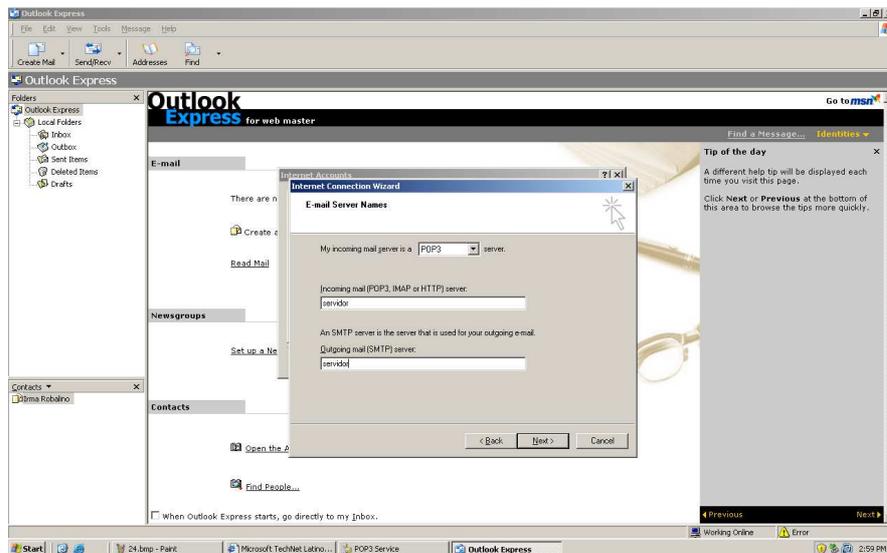
**Figura 58: Configuración de Outlook**

- En Nombre de cuenta, escribir la dirección de correo electrónico que fue creada anteriormente, seguido del dominio; luego se debe hacer clic en Siguiente (Figura 59).



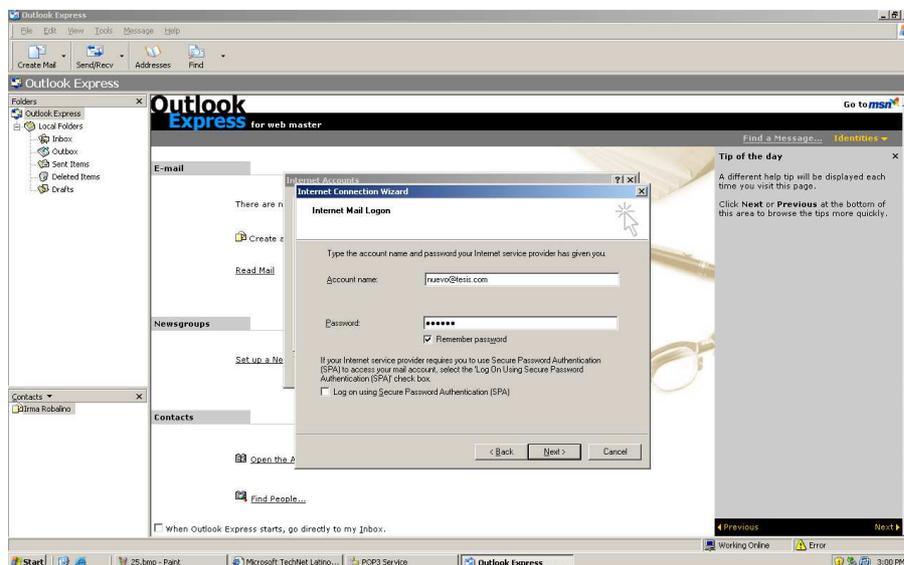
**Figura 59: Configuración de Outlook**

- En esta parte del asistente, es necesario colocar el nombre del servidor de correo, que es el nombre de la máquina donde se instaló el servicio de correo. Colocar el nombre del servidor en la parte de POP3 y de SMTP y hacer clic en Siguiente.



**Figura 60: Configuración de Outlook**

- En esta etapa, colocar el nombre de la cuenta con el dominio para que la validación sea correcta, creada por el usuario anteriormente. Estas cuentas, según este documento, son del tipo Local de Windows. Colocar luego el nombre de la cuenta en Nombre de cuenta y escribir la Contraseña y hacer clic en Siguiente (Figura 61).

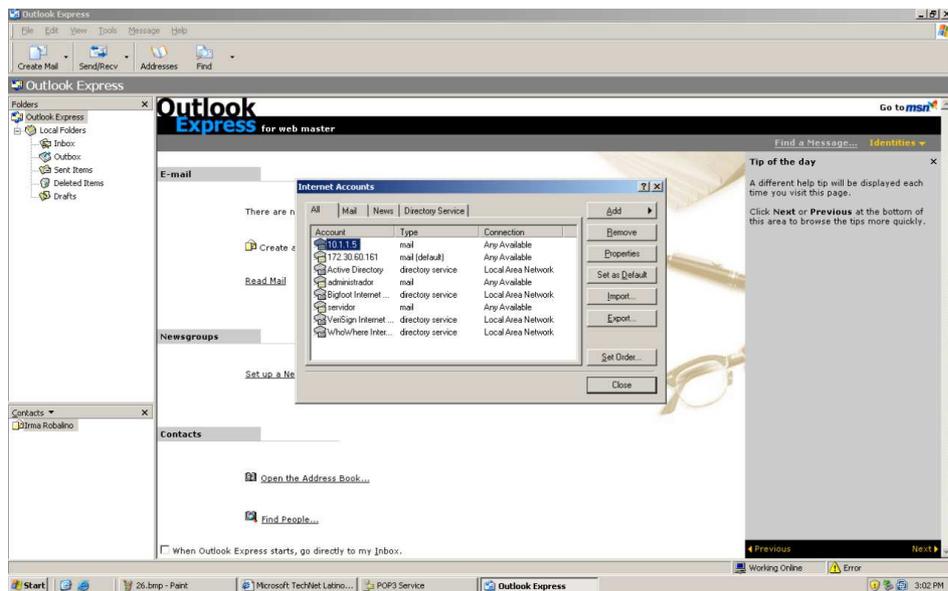


**Figura 61: Configuración de Outlook**

Hacer clic en Finalizar para terminar la configuración básica

### ¿Cómo configurar la cuenta de correo para que se valide al SMTP y así poder enviar correo?

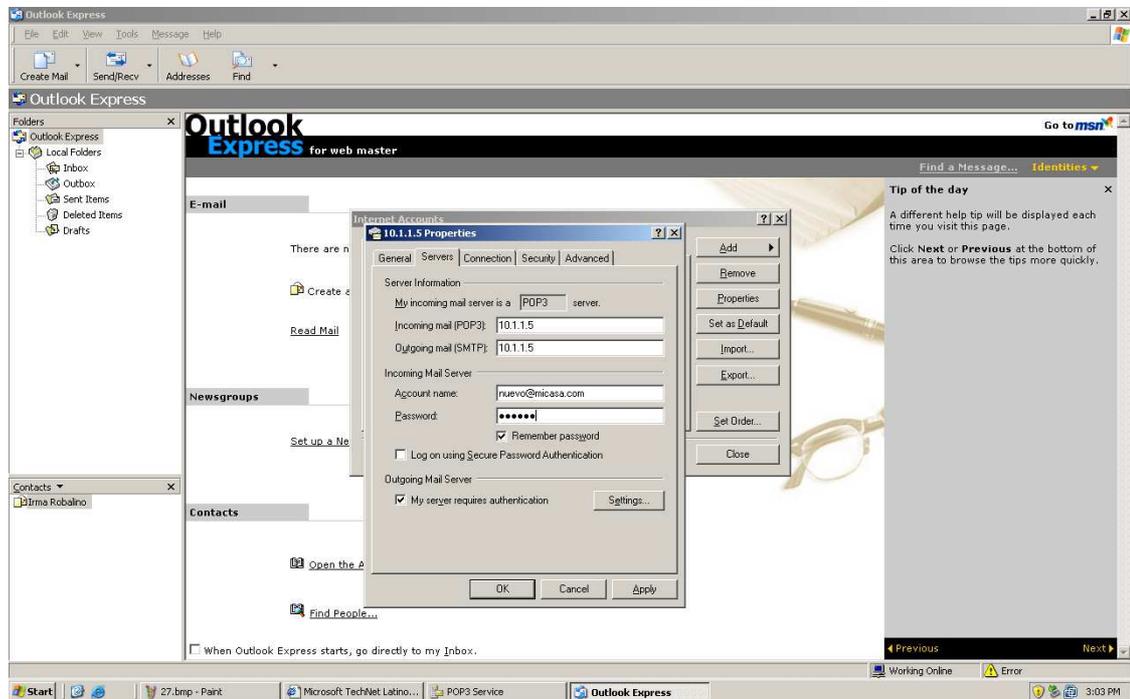
- Hacer clic en Inicio, seleccionar Todos los programas y, a continuación, hacer clic en Outlook Express.
- En el menú Herramientas, hacer clic en Cuentas, seleccionar con un doble clic del mouse la cuenta que se acaba de crear (Figura 62).



**Figura 62: Configuración para envío de correo**

- Para habilitar la validación del Outlook Express para que se puedan enviar correo electrónico en el servidor de correo, seleccionar la solapa Servidores, y hacer clic en

la casilla de verificación que dice Mi servidor requiere autenticación. Luego, hacer clic en Aceptar y posteriormente en Cerrar (Figura 63).



**Figura 63: Configuración para envío de correo**

Una vez realizados todos estos pasos, enviar un correo electrónico para probar su funcionamiento.

## 6.3 FASE DE IMPLEMENTACION

### 6.3.1 CONEXIÓN TÍPICA PARA EL FUNCIONAMIENTO DEL BS2

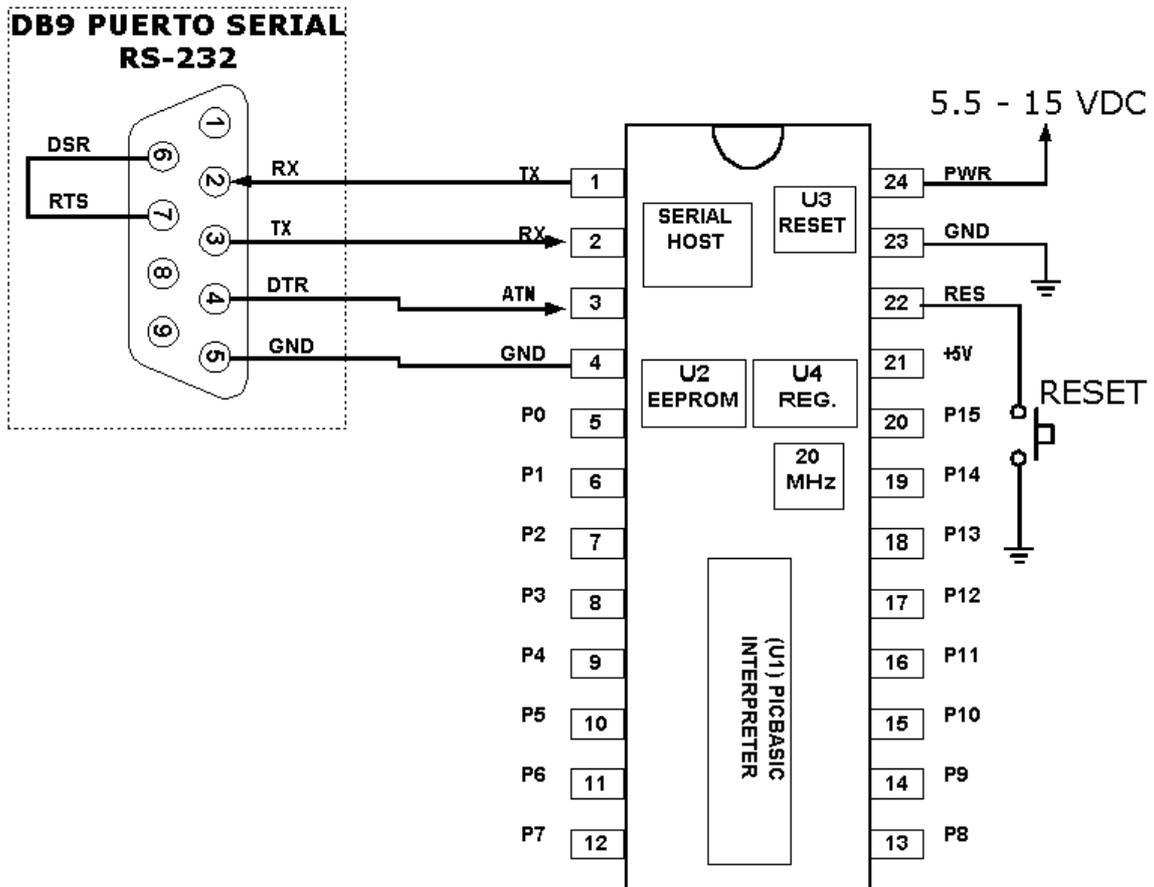


Figura 64: Conexión típica para su funcionamiento

### 6.3.2 MODO DE PROGRAMACIÓN DEL BASIC STAMP 2

#### ✓ Lenguaje de programación PBASIC

El lenguaje de programación PBASIC fue creado específicamente para programar a los BS2, es pariente cercano del lenguaje de programación BASIC, la ventaja que ofrece el

PBASIC con otros lenguajes es su facilidad de aprendizaje. A continuación se explica en detalle cada comando con uno o varios ejemplos.

### ✓ **PBASIC Editor**

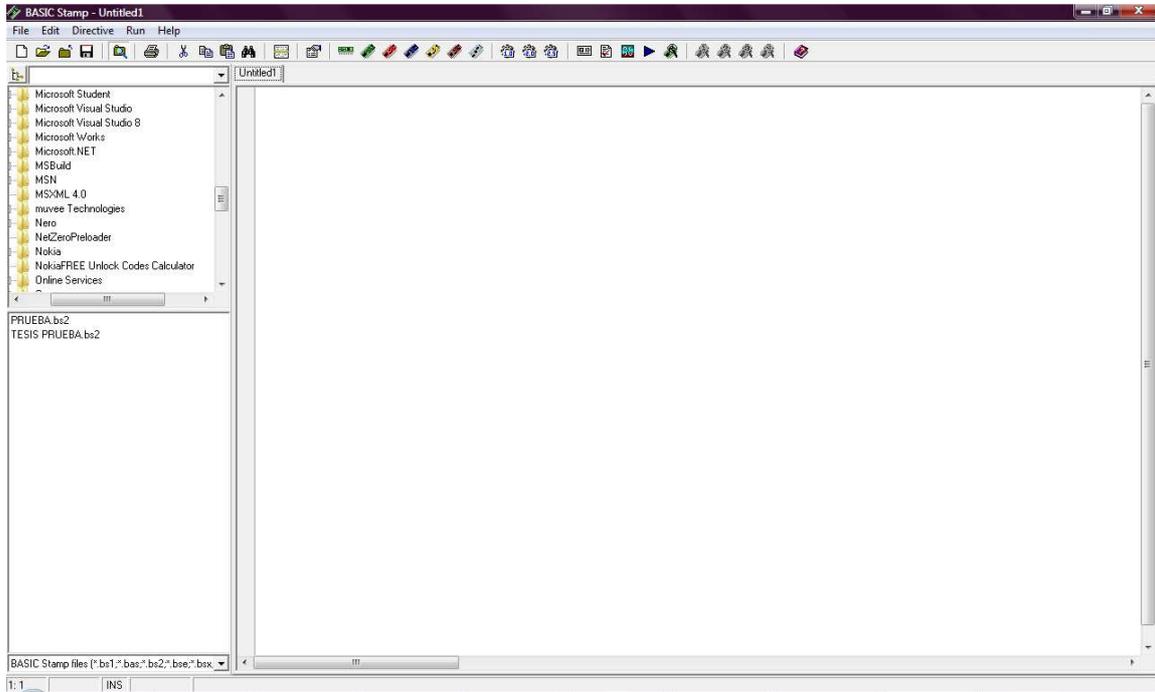
El PBASIC Editor es el programa donde se escribe el conjunto de instrucciones para el Basic Stamp. Es similar en apariencia a cualquier editor de texto del sistema operativo WINDOWS. El editor contiene una serie de herramientas como son: identificadores del Basic Stamp, Corrector ortográfico de sintaxis, Mapa de memoria y Ventana del depurador.

El editor tiene la capacidad para abrir 16 ventanas simultáneamente. La capacidad de cortar, copiar y pegar se mantiene innata. Su entorno es muy sencillo y de fácil comprensión

#### **Los comandos más importantes son:**

- F1 Muestra la ayuda en pantalla
- Ctrl-O Abre un archivo
- Ctrl-S Salva un archivo
- Ctrl-P Imprime el archivo actual
- F9 o Ctrl-R Descarga el programa en el BS2
- F7 o Ctrl-T Corrector de Sintaxis
- F8 o Ctrl-M Muestra el mapa de memoria
- F6 o Ctrl-I Muestra el número de versión de PBASIC
- ESC Cierra la ventana actual

Estos son algunos de los comandos más importantes, aunque es conveniente recordar que [Ctrl-R] descarga el programa al Basic Stamp.

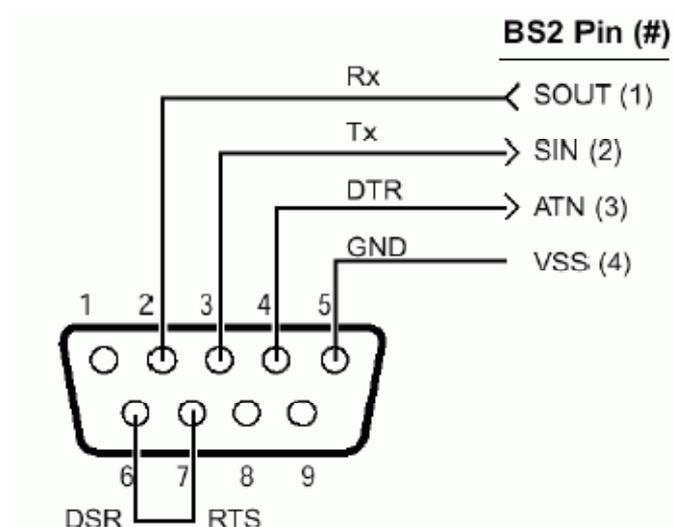


**Figura 65: Pantalla de Editor de Pbasic**

El editor (Figura 65) contiene un grupo de iconos los cuales son fáciles de memorizar. Con solo apuntar con el Mouse el comando será ejecutado. Además posee un menú interactivo.

#### ✓ **Conexión entre la PC y el BS2**

La figura 66 muestra un conector DB9, el cual utiliza 6 pines de los cuales 4 van destinado al BS2 y dos conectados internamente. Preparando este cable (ver Anexo 3) se puede empezar a programar los BS2. En caso de que se utilice un cable serial, debe realizar la conexión de los pines 6 y 7 del cable.

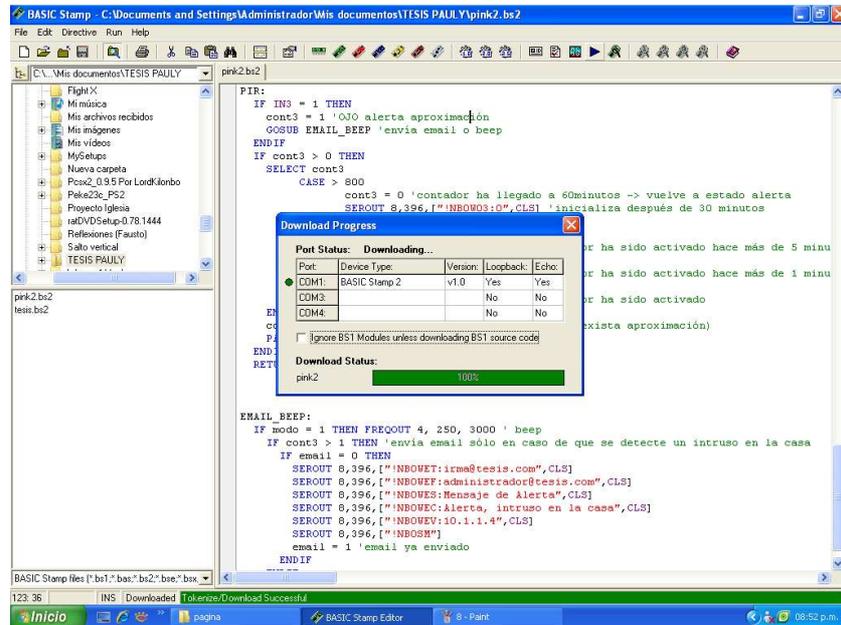


**Figura 66: Conexión del conector DB9**

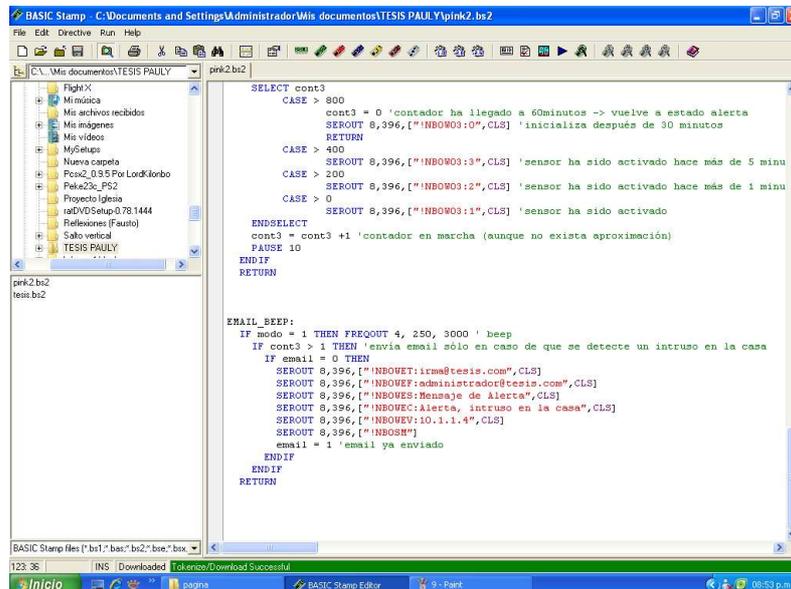
✓ **Procedimiento para descargar el programa al BS2:**

- 1) Con el BS2 previamente energizado y conectado cargar el editor PBASIC.
- 2) Cuando el editor esté listo presionar [Ctrl-I], si todo está bien conectado el editor dará un mensaje de "Found BS2-IC (firmware v1.0)". Esto indica que todo está funcionando correctamente.
- 3) Se puede digitar el programa o cargar uno previamente del disco.
- 4) Para asegurar que el código digitado este bien, presionar el corrector de sintaxis [Ctrl-T], si existe algún problema lo indicara con un mensaje de error. Si todo marcha bien se mostrará un mensaje de "Tokenize Successful"
- 5) Ahora todo está listo para descargar el programa en el BS2, presionar [Ctrl-R], y el programa se descargara permanentemente en la EEprom del BS2. En caso de que no se revise con el corrector de sintaxis, antes de descargarse el programa en el BS2, este lo realiza por su cuenta.

- 6) Apagar el Basic Stamp 2, retirar el cable serial del BS2.
- 7) Encender el Basic Stamp 2 y la aplicación permanecerá hasta que sea modificada nuevamente reprogramando el BS2 por el puerto serial.



**Figura 67: Descarga del programa en el Basic Stamp 2**



**Figura 68: Comprobación de que la descarga se hizo con éxito**

✓ **PROGRAMA DESCARGADO EN EL BS2 (ver Anexo 2)**

### **6.3.3 SENSOR PIR**

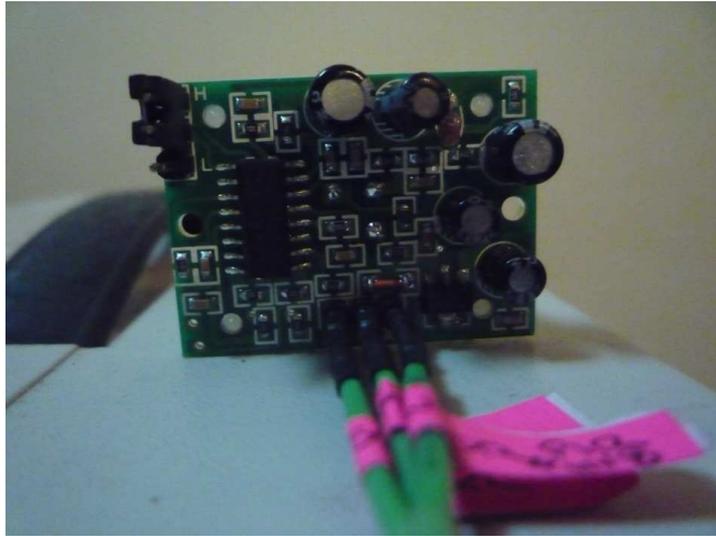
El sensor PIR detector de movimientos por infrarrojos (Figuras 69 y 70) resulta muy adecuado para su empleo en detección de intrusos, gracias a su pequeño tamaño y bajo consumo. El sensor incluye una lente tipo fresnel de plástico que le proporciona un alcance de 5 metros y un ángulo de detección de  $60^\circ$  (ver Anexo 4). La señal de salida es compatible TTL y la alimentación es de 5V con un consumo de tan solo 350  $\mu$ A mientras esta en reposo. Sus reducidas dimensiones de solo 25 x 35 x 18 mm hacen posible su utilización en todo tipo de robots y dispositivos sensores.

Este sensor capta la variación de luz, que después la condiciona y amplifica para proporcionar una tensión de salida de la que obtendremos un nivel lógico "1" para el circuito BS2.

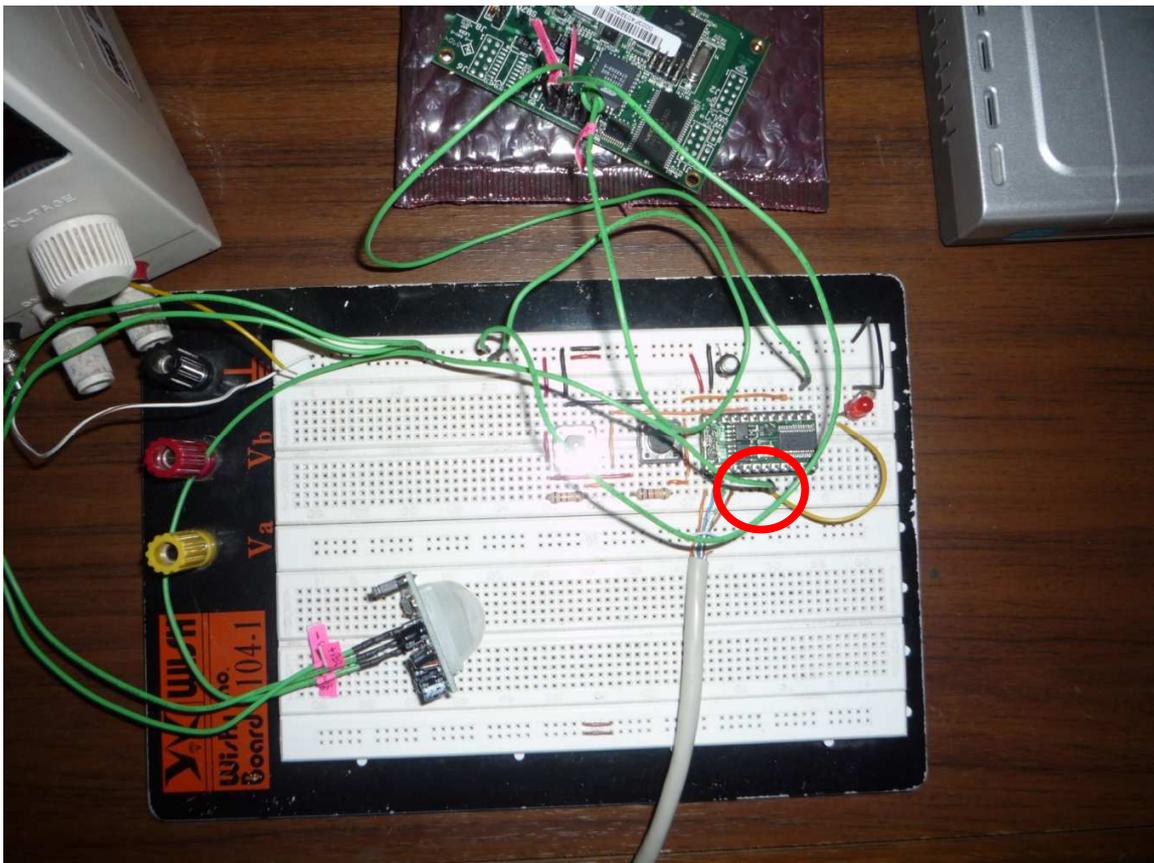
En el presente proyecto el sensor PIR se encuentra conectado al BS2 a través del Pin 3 (Figura 71), es a través de este pin que envía la señal dependiendo de su estado, es decir dependiendo si ha detectado la presencia o no de intrusos en la vivienda.



**Figura 69: Sensor PIR**

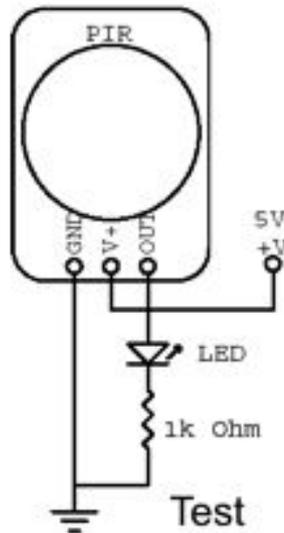


**Figura 70: Sensor PIR**



**Figura 71: Conexión del sensor PIR al BS2**

Se adicionó además un led que permitirá conocer visualmente el momento de la activación del sensor siguiendo el diseño de la figura 72:



**Figura 72: Led adicionado**

Se tomó en cuenta además que al energizarse el sensor PIR requiere de un tiempo de preparación para comenzar a operar de forma adecuada. Esto se debe a que tiene que ocurrir la adaptación a las condiciones propias de operación del ambiente donde fue instalado. Durante este período el sensor “aprende” a reconocer el estado de reposo o no movimiento del ambiente. La duración de esta calibración puede estar entre 10 y 60 segundos.

#### **6.3.4 MODULO PINK**

Este módulo denominado PINK (**Parallax Internet Netburner Kit**) (Figura 73) constituye un servidor web embebido (incrustado en el circuito del módulo) capaz de alojar las páginas

web del usuario, desde las cuales se pueda mostrar datos que pueda gestionar el circuito BS2, como es el caso en este proyecto, donde se interactuará con una de las 100 variables que soporta. Además, posibilitará el envío de mensajes email de acuerdo con la gestión del BS2 y los sensores.

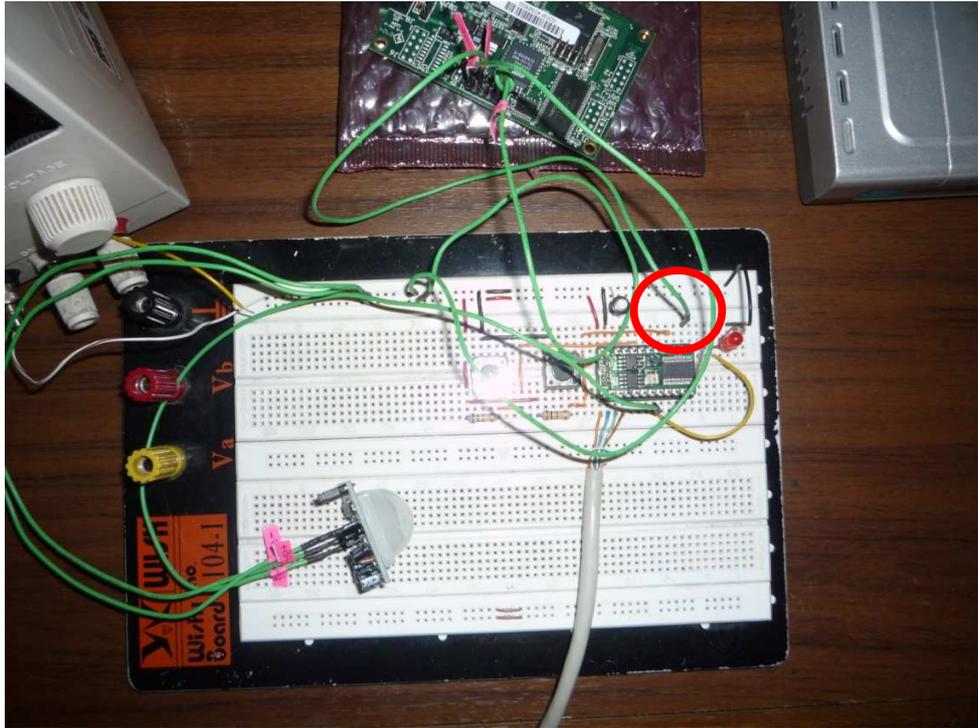


**Figura 73: Modulo PINK**

En el proyecto desarrollado se utilizó el pin P8 (Figuras 74 y 75) para comunicar el circuito BS2 con el módulo PINK (en modo envío), para que de esta manera pueda pasar serialmente valores de variables y el PINK las haga servir en una página web.



**Figura 74: Pin en el modulo que recibe la información**



**Figura 75: Conexión del modulo PINK con el BS2**

# CAPITULO VII

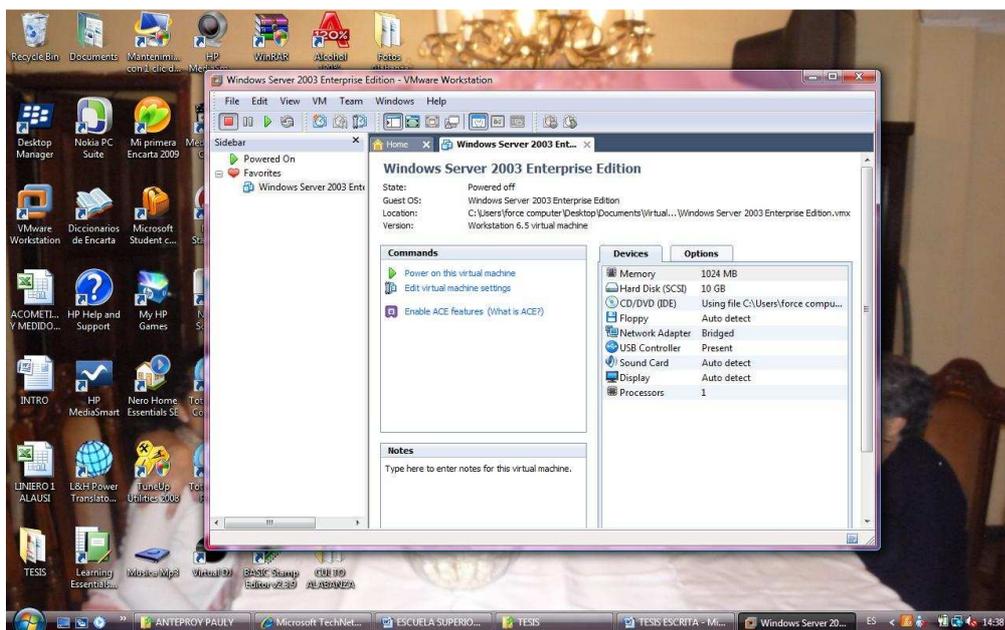
## PRUEBAS Y RESULTADOS

Luego de la programación, diseño e implementación del proyecto de tesis se realizaron las pruebas correspondientes para determinar el correcto funcionamiento o no del mismo.

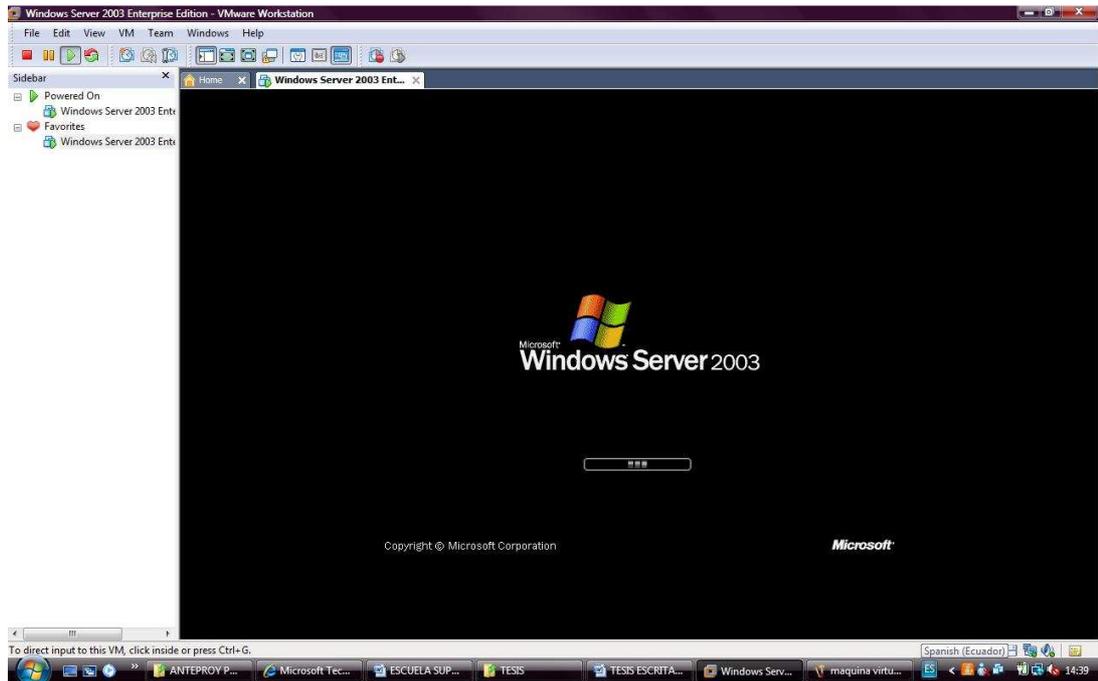
### 7.1 Prueba de servidor SMTP:

La primera prueba que se realizó fue la de confirmación de que el servidor SMTP configurado anteriormente se encuentre cumpliendo con su objetivo que es el de envío y recepción de correos entre dos máquinas dentro de una intranet.

- Es así que se inició encendiendo la máquina virtual (Figuras 76 y 77), donde se encuentra instalado el sistema operativo Windows 2003 que permitió la configuración del servidor SMTP.

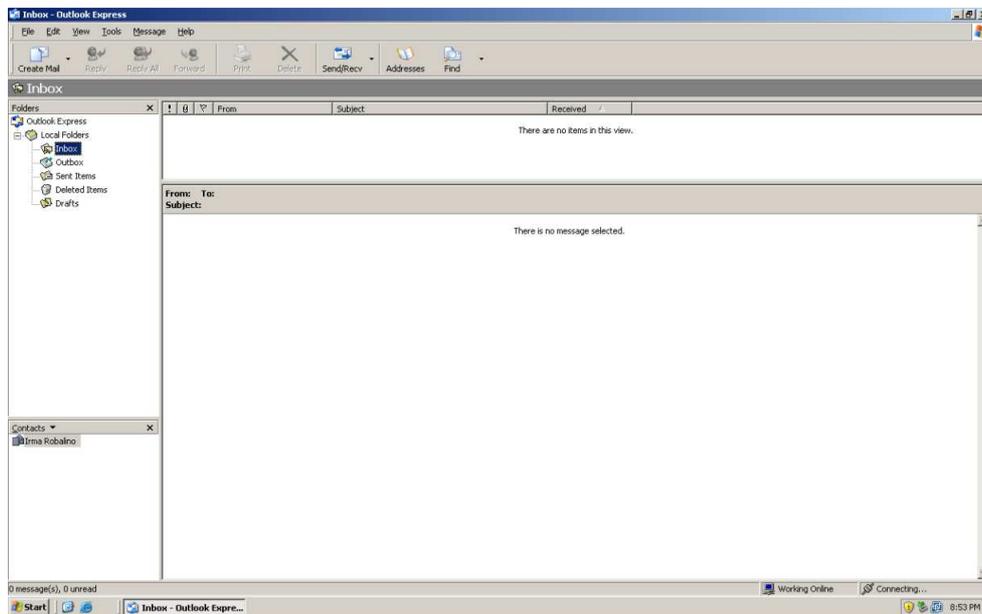


**Figura 76: Encendido de la máquina virtual**

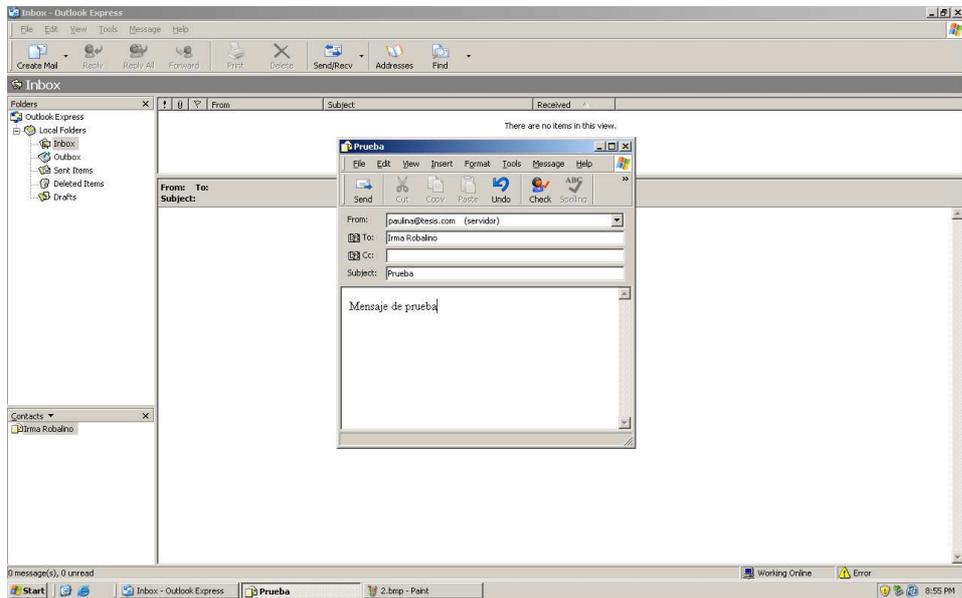


**Figura 77: Encendido de la máquina virtual**

- Luego se envió un mensaje de prueba desde el servidor [paulina@tesis.com](mailto:paulina@tesis.com) hasta el cliente [irma@tesis.com](mailto:irma@tesis.com) (Figuras 78 y 79).

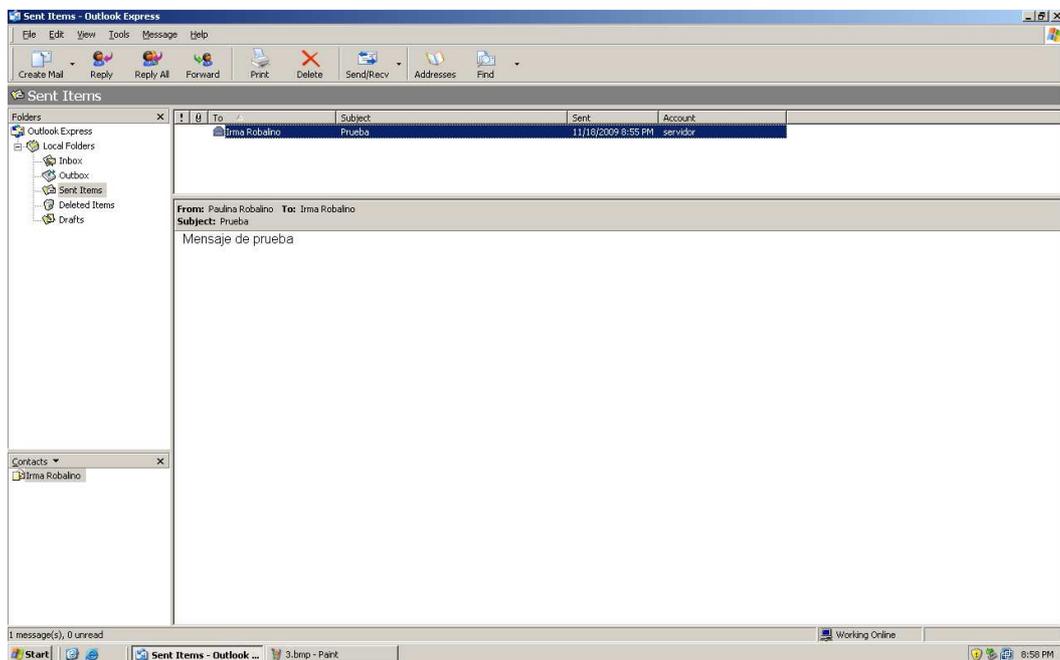


**Figura 78: Envío de mensaje de prueba**

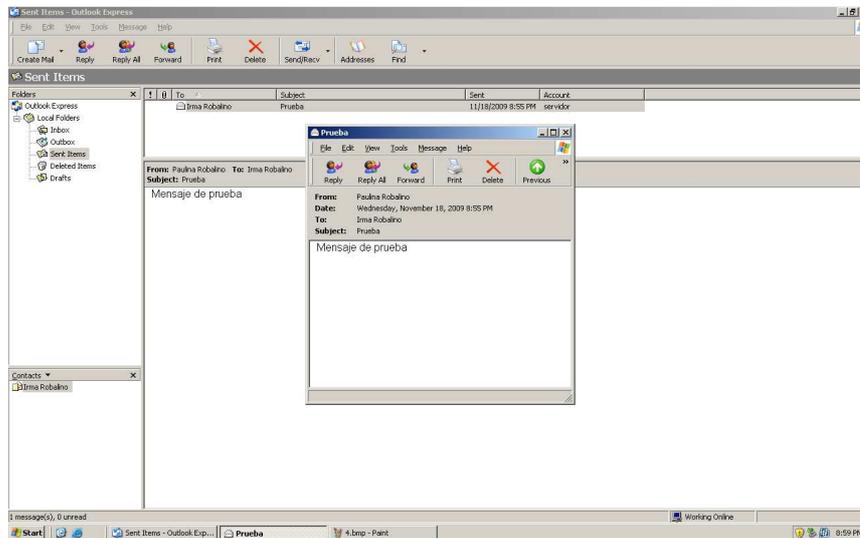


**Figura 79: Envío de mensaje de prueba**

El mensaje fue enviado correctamente (Figuras 80 y 81)

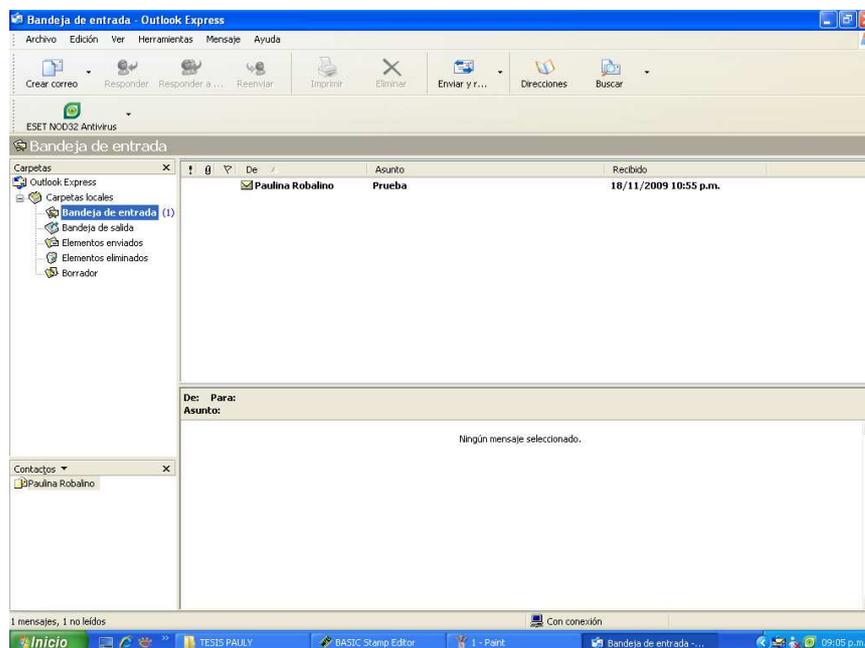


**Figura 80: Mensaje enviado con éxito**

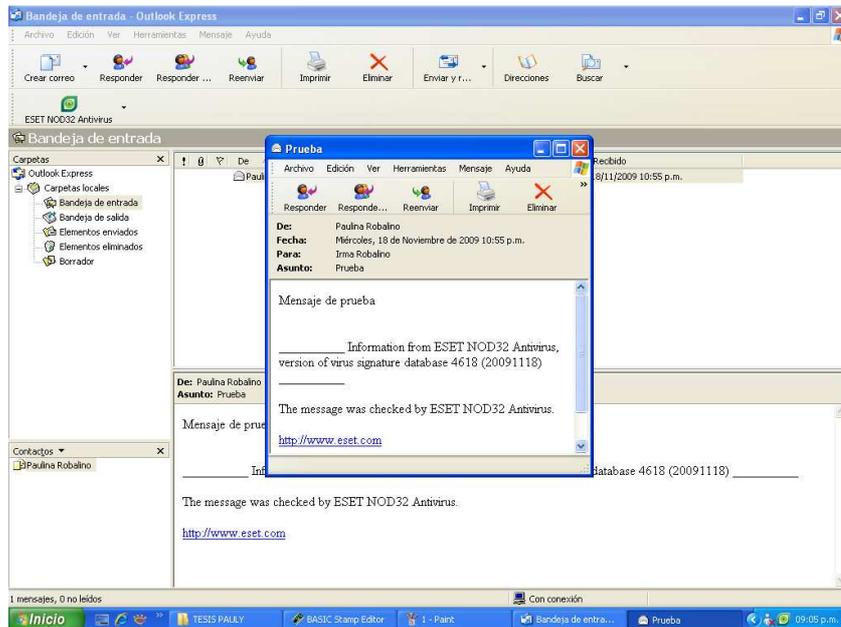


**Figura 81: Mensaje enviado con éxito**

- En la otra máquina se pudo comprobar que se recibió el mensaje con éxito (Figuras 82 y 83):

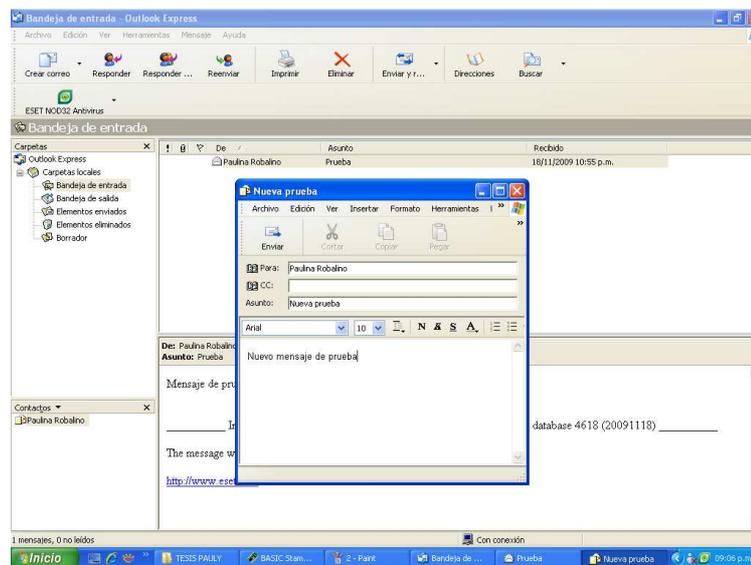


**Figura 82: Mensaje recibido con éxito**

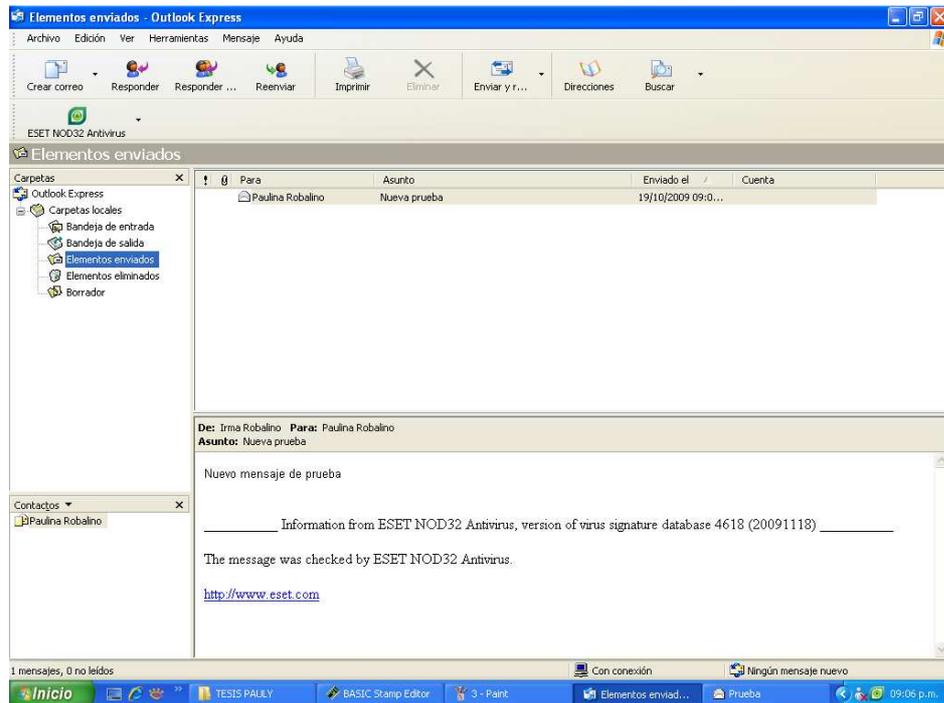


**Figura 83: Mensaje recibido con éxito**

- ☑ Se volvió a realizar otra prueba pero ahora el mensaje es enviado desde el cliente [irma@tesis.com](mailto:irma@tesis.com) hasta el servidor [paulina@tesis.com](mailto:paulina@tesis.com) (Figuras 84 y 85).

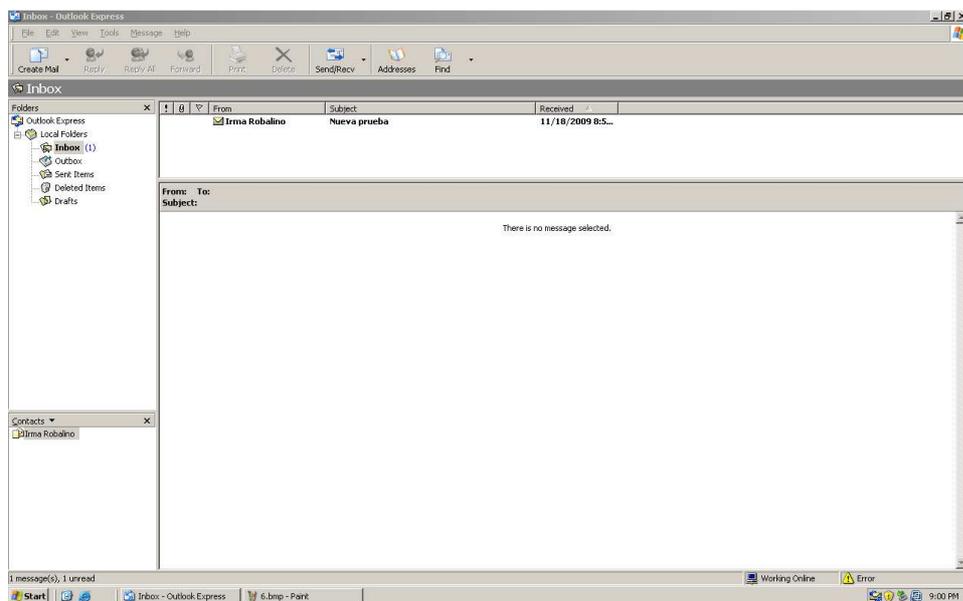


**Figura 84: Mensaje enviado desde la máquina cliente**

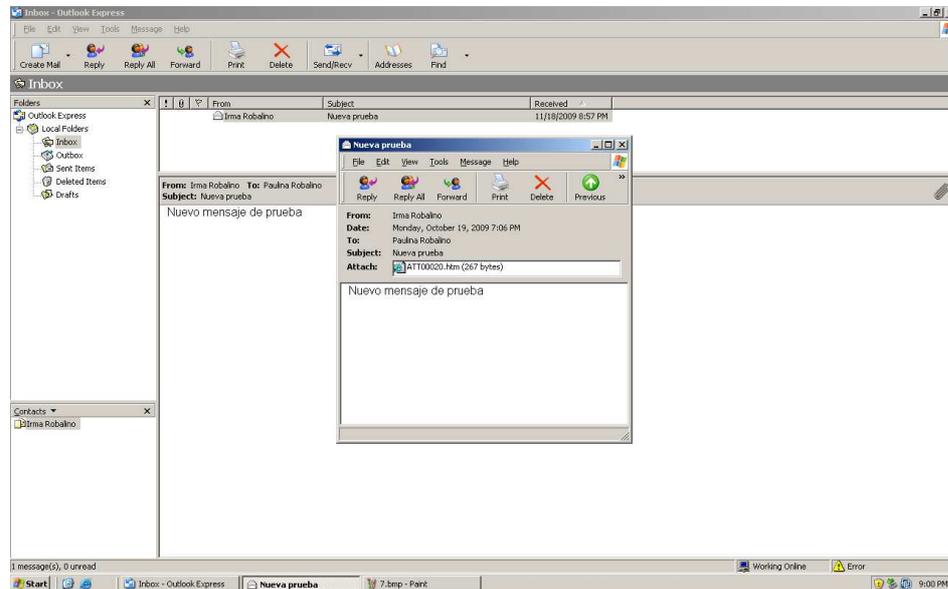


**Figura 85: Mensaje enviado desde la máquina cliente**

- Y se puede comprobar que el mensaje llegó hasta el servidor con éxito (Figuras 86 y 87).



**Figura 86: Mensaje recibido desde la máquina cliente**



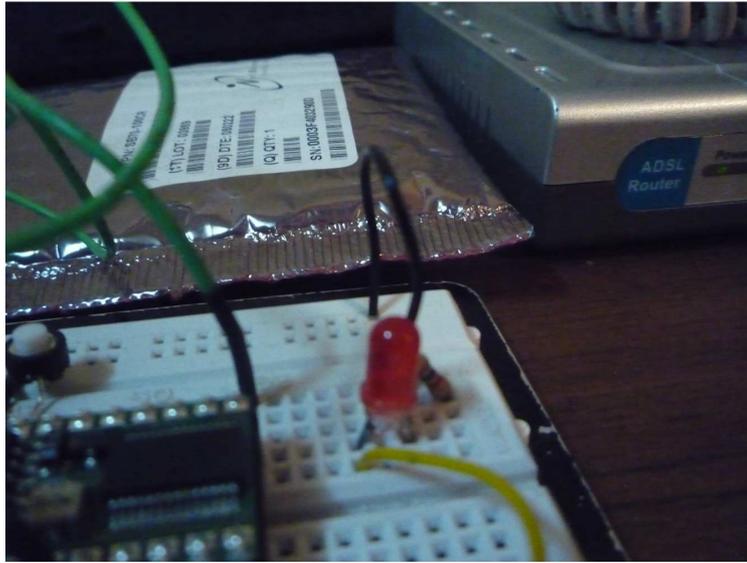
**Figura 87: Mensaje recibido desde la máquina cliente**

De esta forma queda confirmado de que la configuración del servidor SMTP fue hecha en forma correcta y funciona completamente.

## **7.2 Prueba de detección de intrusos a través del sensor de aproximación PIR**

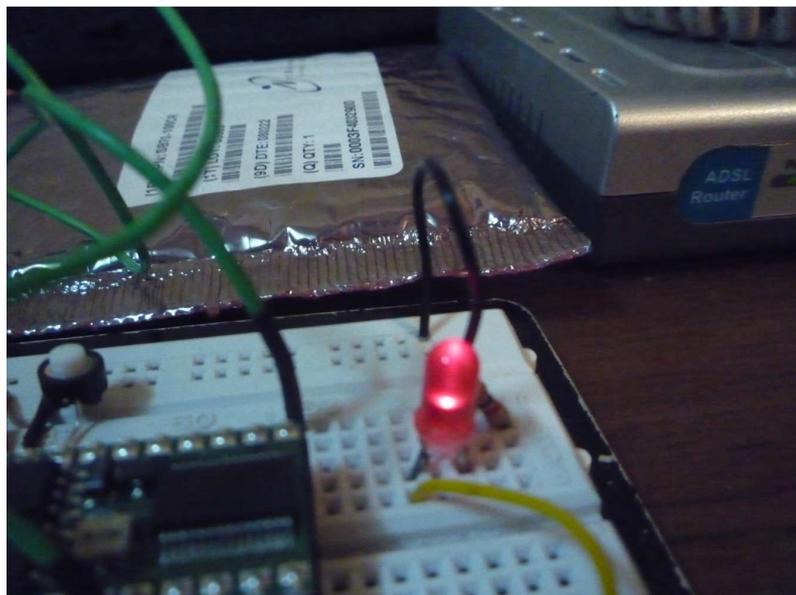
Como se mencionó anteriormente, al diseño del circuito se le agregó un led que permita detectar visualmente la activación o no del sensor.

- En la figura 88 se muestra el led determinando que el sensor PIR se encuentra inactivo.



**Figura 88: Led que muestra que el sensor está inactivo**

- En ésta nueva imagen (Figura 89), se ve claramente como el led muestra que el sensor PIR se ha activado ya que ha detectado la aproximación.



**Figura 89: Led encendido como comprobación de la activación del sensor**

Se comprueba entonces que el sensor se encuentra funcionando correctamente.

### **7.3 Prueba de funcionamiento del modulo PINK.**

La prueba de funcionamiento de este modulo es sencilla ya que se la identifica a través del encendido de un led propio del modulo. La figura 90 muestra que el modulo PINK se encuentra funcionando correctamente.

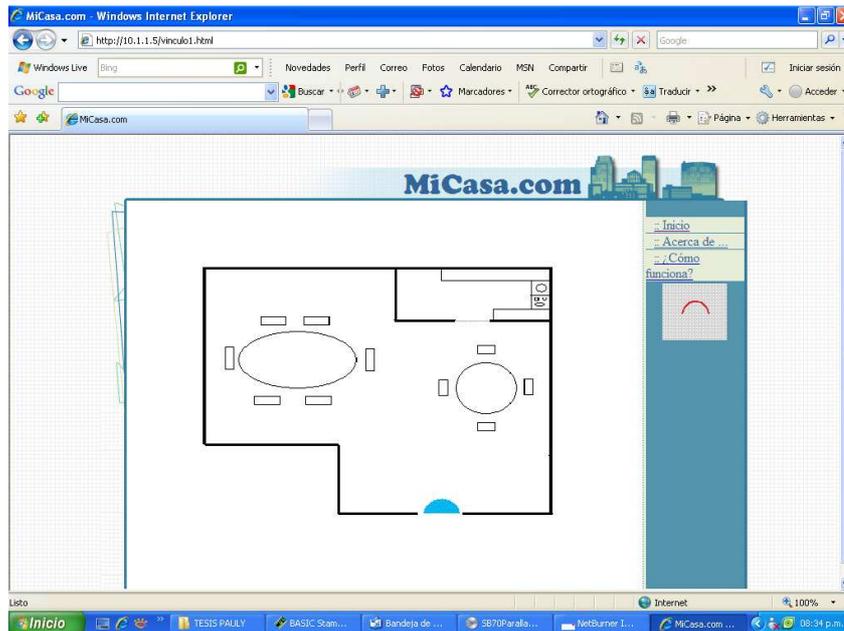


*Figura 90: Funcionamiento correcto del modulo PINK*

### **7.4 Prueba de alerta de intrusos a través de la página web Micasa.com**

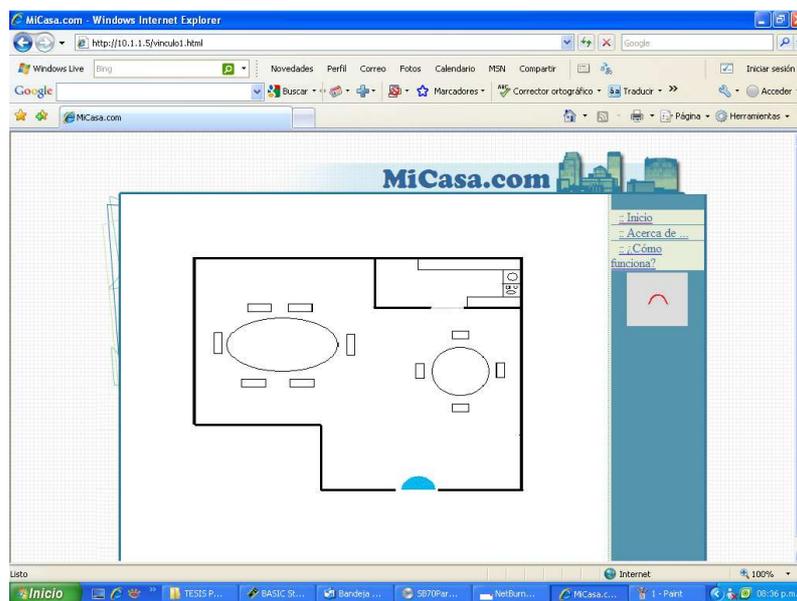
Al acceder a la opción de Micasa en el menú de la página inicial, se puede identificar la imagen del plano de la vivienda donde se instalará el sistema y además en el lado derecho se puede identificar claramente las imágenes de las huellas que mostrarán gráficamente la activación del sensor.

- Es así que se muestra una huella roja inmóvil (Figura 91) cuando el sensor se encuentra en estado de reposo.



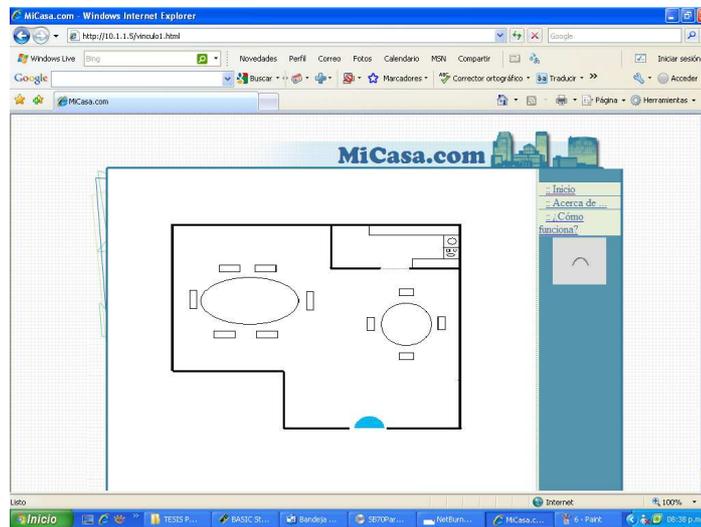
**Figura 91: Huella en reposo debido a que el sensor está inactivo**

- Esta huella se mantiene hasta que el sensor se active y muestre en el instante de su activación una huella roja en movimiento (Figura 92).



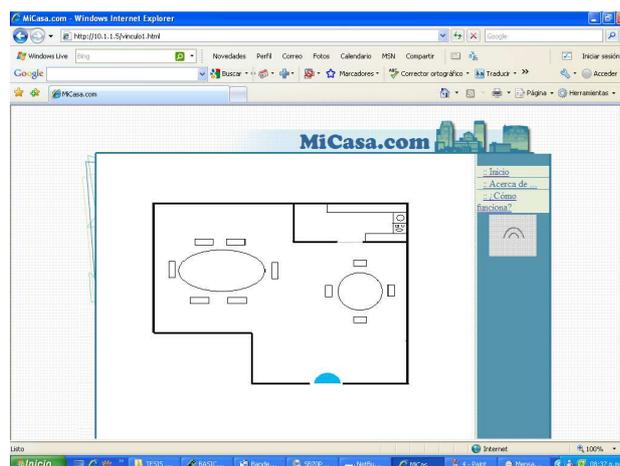
**Figura 92: Huella roja en movimiento, muestra la activación del sensor**

- ☑ Transcurridos treinta segundos después de la activación del sensor, la huella cambia de estado a una de color gris pero también en movimiento (Figura 93).



**Figura 93: Huella gris en movimiento, activación del sensor hace 30 segundos**

- ☑ Finalmente luego de 60 segundos luego de que el sensor haya detectado aproximación, la huella cambiará a una color gris en estado inmóvil, terminando el proceso de detección a través de la página web.



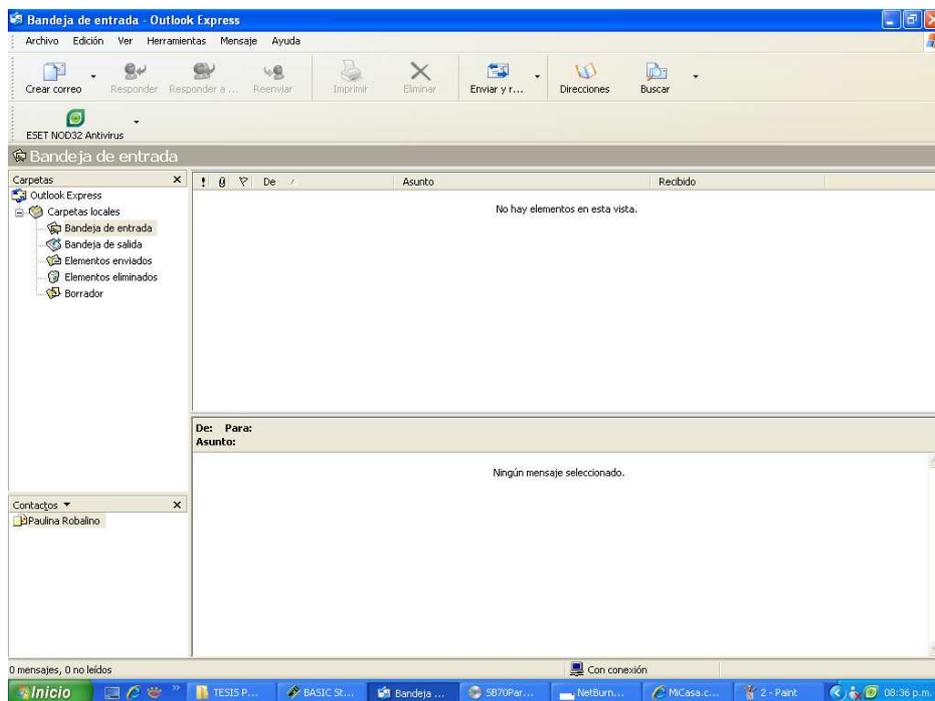
**Figura 94: Huella gris sin movimiento, activación del sensor hace 60 segundos**

La prueba de detección de intrusos a través de la página web se concluye con éxito.

### 7.5 Prueba de alerta de intrusos a través del envío de e-mail.

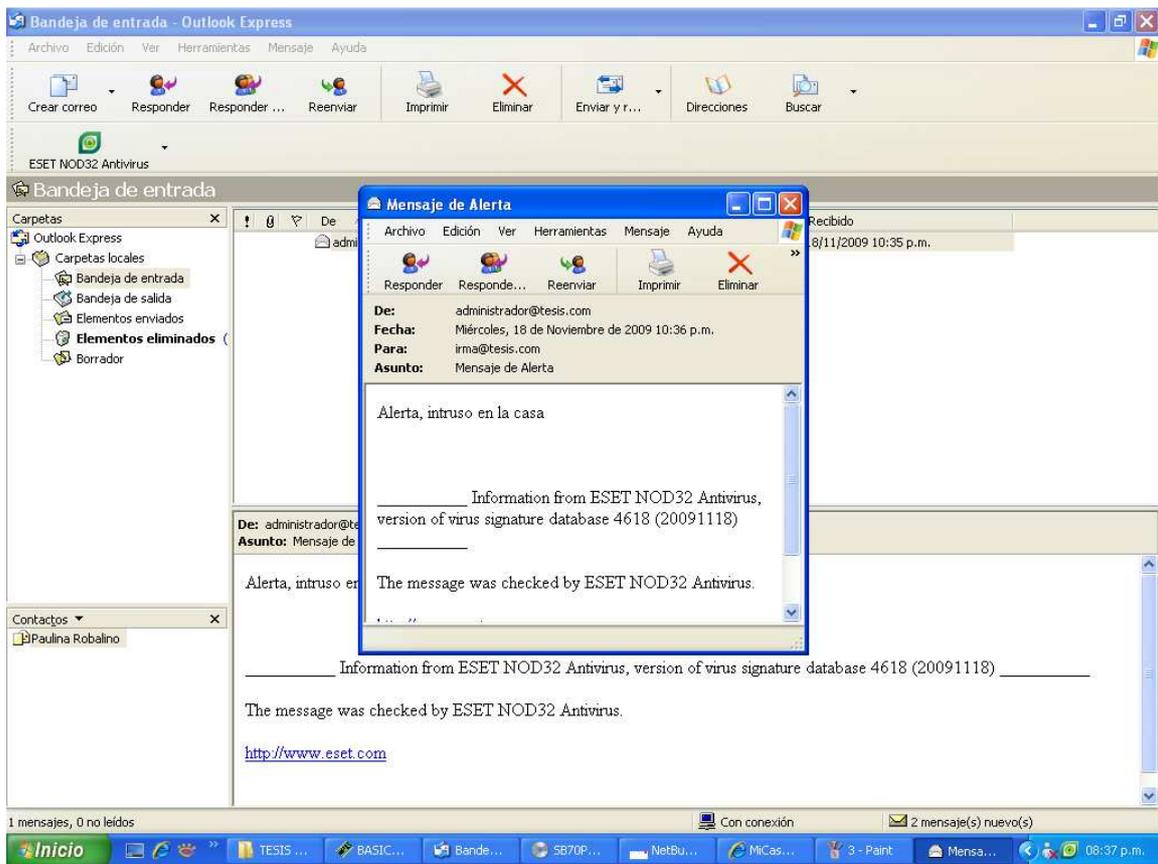
Finalmente se realiza la prueba para determinar si el modulo PINK y el BS2 están trabajando correctamente para enviar un mensaje de alerta al usuario en el momento en que el sistema detecta la presencia de intrusos en la vivienda.

- Pantalla que muestra que el correo se encuentra vacío.



**Figura 95: Pantalla sin correos recibidos**

- Pantalla que muestra que el envío y recepción de mensajes es correcto, y el usuario si recibe un e-mail de alerta de que en su vivienda existe la presencia de intrusos.



**Figura 95: El mensaje de alerta es recibido con éxito**

## ANALISIS DE RESULTADOS

El sensor PIR es el dispositivo encargado de monitorizar el entorno captando la información que transmite el sistema, se ha comprobado entonces que el sensor de aproximación está enviando correctamente la señal que alerta al usuario de la presencia de intrusos.

El Basic Stamp hace el trabajo de un controlador, ya que se comprueba que es el dispositivo encargado de gestionar el sistema según la programación y la información que recibe.

La interfaz en este caso es el Internet, y luego de las pruebas se puede observar que muestra la información del sistema al usuario interactuando con él tanto en forma gráfica dentro de la página web, como a través de e-mails de alerta.

De las pruebas realizadas se desprenden los siguientes resultados:

- El programa realizado en el Basic Stamp y la configuración del modulo PINK permiten que la interconexión entre los elementos del circuito proporcionen la información necesaria sobre los acontecimientos en la vivienda, en este caso la presencia de intrusos.
- La información recogida por el sensor del sistema se constituye en la base para que el usuario se encuentre informado de lo que sucede en su hogar en todo momento.
- Gracias a la correcta implementación se alcanza el objetivo de disponer de un sistema de alarma doméstico conectado a Internet.

Es preciso destacar que todos los dispositivos del presente proyecto no tienen que estar físicamente separados ya que varias funcionalidades pueden ser combinadas en el

mismo. En este caso, el sistema de alarma está compuesto por un controlador, un sensor y una interfaz como se mencionó anteriormente.

Luego del análisis realizado, se puede concluir que el sistema diseñado es un sistema de alarma de arquitectura centralizada, ya que dispone de un controlador centralizado que envía la información a la interfaz según el programa, la configuración y la información que recibe del sensor y de los sistemas interconectados.

## CONCLUSIONES

- a)** Se concluye, que a pesar de las capacidades potenciales que tienen, tan sólo una parte ínfima de los microcontroladores están preparados por el fabricante para intercambiar datos con otros dispositivos u ordenadores, es así que luego de las consideraciones necesarias (ver Anexo 5) se estableció que el Basic Stamp 2 era la opción adecuada para llevar a cabo el presente proyecto.
- b)** Luego de estudiar las funciones de un servidor web se llega a la conclusión de que este es útil, entre otras cosas, para conectar un microcontrolador como puede ser el BS a una red Ethernet, como Internet, e intercambiar datos de lo que ocurra en una vivienda brindando así un eficiente sistema de control domótico.
- c)** El desarrollo de este proyecto permite concluir que se puede establecer una relación entre los microcontroladores, el Internet y las herramientas de control domótico ya que solo se precisa de una interfaz basada en web para poder interactuar con el circuito Basic Stamp con toda la amigabilidad y facilidad que ofrece el entorno web, permitiéndole al BS2 de esta forma conectarse al mundo que le rodea.
- d)** El sistema de control domótico que se diseñó permite la interacción bidireccional entre el usuario y su hogar a través de la información que facilita la página web y los e-mails.
- e)** Se concluye que el sistema propuesto debe considerarse como una concepción básica de un sistema de alarma conectado a Internet capaz de detectar la presencia de intrusos en una determinada zona, pero que puede constituirse en una alarma potente por cuanto se puede fácilmente realizar algunas mejoras, de acuerdo con las necesidades y la disponibilidad que se tenga.

## RECOMENDACIONES

- a.** Antes de configurar el servidor SMTP debe haber la seguridad que el servidor en el que se tiene intención de instalar los servicios de correo electrónico disponga de una conexión a Internet o cuente con medios para tener acceso a Internet.
- b.** Es altamente recomendable la ausencia de personas en la proximidad del sensor de aproximación PIR mientras este se calibra.
- c.** Al momento de ubicar la contraseña para la configuración FTP se debe estar seguro de que la contraseña que se escoja sea fácil de recordar ya que si se olvida puede producir un bloqueo en el módulo.
- d.** Se recomienda que antes de conectar el modulo PINK se debe tener una buena comprensión sobre redes de computadoras y como configurar dichas redes.
- e.** Es recomendable revisar varias veces las conexiones y el valor que está enviando la fuente antes de alimentar a los elementos ya que todos ellos trabajan con 5V.

## RESUMEN

El presente sistema de alarma resulta del estudio de los servidores web embebidos y los microcontroladores, herramientas que facilitan la implementación de un sistema domótico capaz de monitorear una vivienda a través de una red Ethernet.

Para la implementación del sistema se utilizó un módulo PINK (servidor web embebido), un microcontrolador BASIC Stamp 2 y un detector de movimiento por infrarrojo (sensor PIR). El método analítico permitió dividir el objeto de estudio en varias partes, además se utilizaron las técnicas aplicadas en redes de computadores, diseño de páginas web y programación de microcontroladores.

Utilizando estas herramientas se procedió a la implementación; se alojó en el módulo PINK la página web que permite al usuario visualizar, a través de huellas consecutivas, lo que ocurre en su hogar, se programó el microcontrolador BASIC Stamp 2 que gobierna el proceso de recepción de señales enviadas por el sensor, y finalmente se dispuso el sensor de aproximación que detectará una intrusión en el hogar. Esta alarma puede detectar intrusos en un ángulo de 60° a 5 metros de distancia. Cuando se activa el sistema el usuario dispone de 10 segundos para abandonar su hogar. Al sistema le toma aproximadamente 5 segundos para enviar el e-mail de alerta.

Los estudios realizados permiten obtener un sistema de alarma conectado a Internet que detecta la presencia de intrusos en una determinada zona de una vivienda y alerta al usuario en todo momento de lo que sucede en la misma.

## SUMMARY

The present alarm system is from the study of the embedded Web servers and the microcontrollers, tools that facilitate the implementation of a domotic system able to monitor a house through a network Ethernet. For the implementation of the system a unit PINK was used (embedded Web server), a microcontroller BASIC 2 Stamp and a detector of movement by infrared (sensorial PIR). The analytical method allowed dividing the object of study in several parts, in addition the techniques applied in networks of computers, design of web pages and programming of microcontroller were used. Using these tools it was come to the implementation; the page lodged in module PINK Web that allows the user to visualize, through consecutive tracks, which happens in its home, the microcontroller BASIC Stamp 2 programmed itself who governs the process of reception of signals sent by the sensor, and finally the sensor was had approach that will detect a intrusion in the home. This alarm can detect intruders in an angle of 60° to 5 meters of distance. When the system activates the user has 10 seconds to leave his home. To the system he takes approximately 5 seconds to send the alert email. The realised studies allow obtaining an alarm system connected to Internet that detects the presence of intruders in a certain zone of a house and alerts to the user at any moment than it happens in the same.

## GLOSARIO

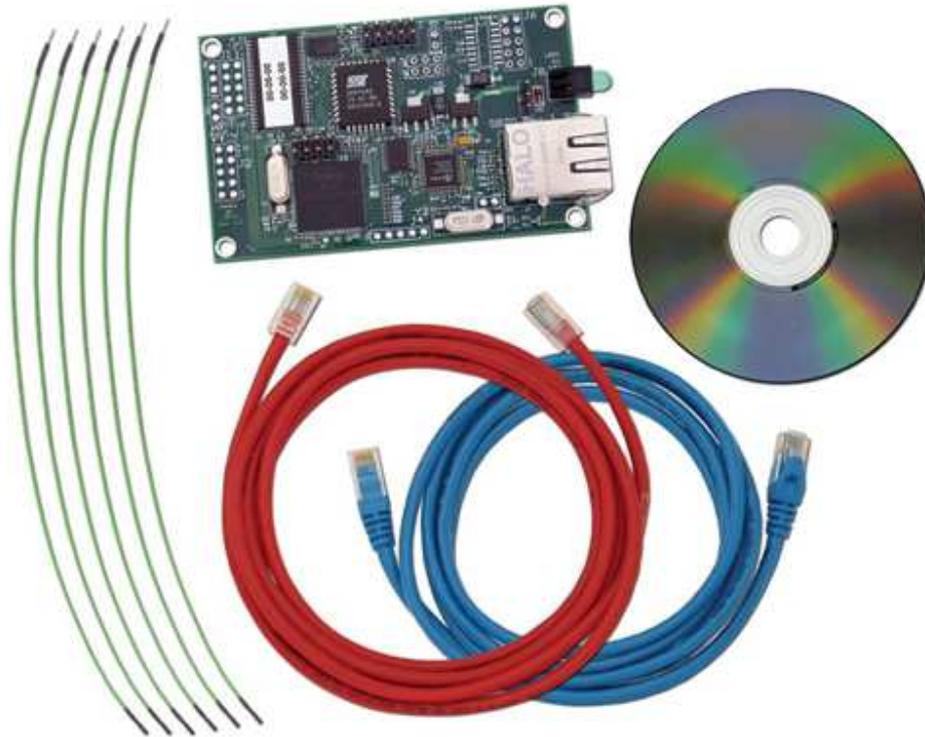
<b>BAUD RATE</b>	Es un promedio del número de bits, caracteres o bloques, que se transfieren entre dos dispositivos, por una unidad de tiempo.
<b>BS2</b>	Basic Stamp 2, es un microcontrolador programable que puede controlar y sensar motores, interruptores, etc.
<b>COMPARADOR</b>	Circuito o dispositivo que detecta cuándo la tensión de entrada es mayor que un valor límite predeterminado. La salida es una tensión alta o bien una tensión baja.
<b>DHCP</b>	Dynamic Host Configuration Protocol - Protocolo Configuración Dinámica de Servidor, es un protocolo de red que permite a los nodos de una red IP obtener sus parámetros de configuración automáticamente.
<b>DIRECCION IP</b>	Es un número que identifica de manera lógica y jerárquica a una interfaz de un dispositivo (habitualmente una computadora) dentro de una red que utilice el protocolo IP (Internet Protocol), que corresponde al nivel de red del protocolo TCP/IP.
<b>DNS</b>	Sistema de nombre de dominio (en inglés Domain Name System), este sistema asocia información variada con nombres de dominios asignado a cada uno de los participantes. Su función más importante, es traducir (resolver) nombres ilegibles para los humanos en identificadores binarios asociados con los equipos conectados a la red, esto con el propósito de poder localizar y direccionar estos equipos mundialmente.
<b>EEPROM</b>	Erasable Programmable Read Only Memory, es un tipo de memoria ROM que puede ser programada, borrada y reprogramada eléctricamente.

<b>FTP</b>	FTP (sigla en inglés de File Transfer Protocol - Protocolo de Transferencia de Archivos) en informática, es un protocolo de red para la transferencia de archivos entre sistemas conectados a una red TCP (Transmission Control Protocol), basado en la arquitectura cliente-servidor.
<b>INTERFAZ</b>	Sistema de comunicación de un programa con su usuario; la interfaz comprende las pantallas y los elementos que informan al usuario sobre lo que puede hacer, o sobre lo que está ocurriendo.
<b>MEMORIA FLASH</b>	Forma desarrollada de la memoria EEPROM que permite que múltiples posiciones de memoria sean escritas o borradas en una misma operación de programación mediante impulsos eléctricos.
<b>MICROCONTROLADOR</b>	Circuito integrado o chip que incluye en su interior las tres unidades funcionales de una computadora: unidad central de procesamiento, memoria y unidades de E/S (entrada/salida).
<b>PBASIC</b>	Es una versión de BASIC basada en microcontroladores y creada por Parallax, su lenguaje fue creado para dar facilidades en el uso de los microcontroladores y sistemas embebidos. Es un compilador para los Basic Stamp de Parallax.
<b>PINK</b>	Parallax Internet Natburner Kit, servidor Web Embebido que permite al Basic Stamp comunicarse con una red e incluso la Internet.
<b>PIR Sensor</b>	Passive Infra Red, es un dispositivo piroeléctrico que mide cambios en los niveles de radiación infrarroja emitida por los objetos a su alrededor.
<b>POP3</b>	Post Office Protocol (POP3, Protocolo de la oficina de correo). Es un protocolo estándar para recibir mensajes de e-mail. Los mensajes de e-mails enviados a un

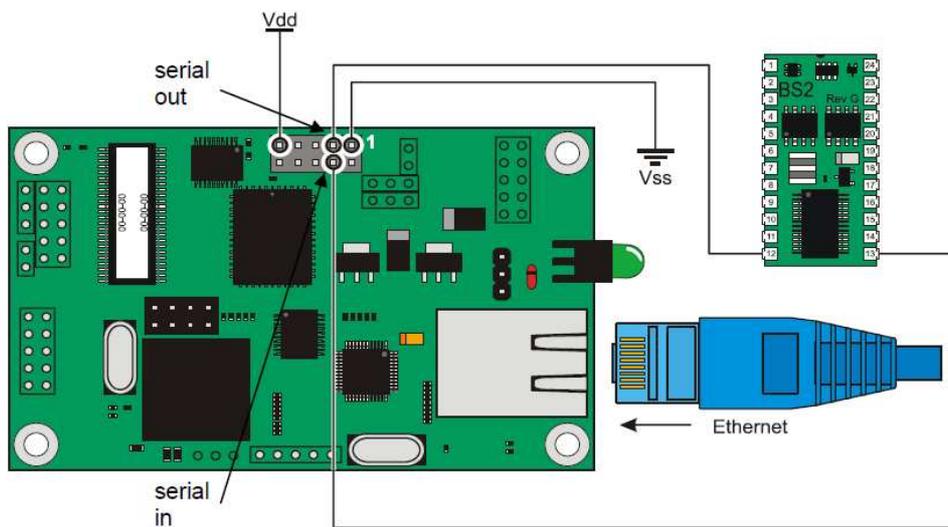
	<p>servidor, son almacenados por el servidor pop3. Cuando el usuario se conecta al mismo (sabiendo la dirección POP3, el nombre de usuario y la contraseña), puede descargar los ficheros.</p>
<b>RAM</b>	<p>Random Access Memory o Memoria de Acceso Aleatorio, es donde el computador guarda los datos que está utilizando en el momento presente. El almacenamiento es considerado temporal por que los datos y programas permanecen en ella mientras que la computadora este encendida o no sea reiniciada.</p>
<b>ROM</b>	<p>Read-only memory o <i>memoria de sólo lectura</i>, es la memoria que se utiliza para almacenar los programas que ponen en marcha el ordenador y realizan los diagnósticos. La mayoría de los ordenadores tienen una cantidad pequeña de memoria ROM (algunos miles de bytes).</p>
<b>RS232</b>	<p>Un conector de interfaz de ordenador utilizado para conectar dispositivos de puerto serie como instrumentos para la transferencia de la información.</p>
<b>SMTP</b>	<p>Simple Mail Transfer Protocol (SMTP) Protocolo Simple de Transferencia de Correo, es un protocolo de la capa de aplicación. Protocolo de red basado en texto utilizado para el intercambio de mensajes de correo electrónico entre computadoras u otros dispositivos.</p>
<b>TEMPORIZADOR</b>	<p>Se denomina temporizador al dispositivo mediante el cual se puede regular la conexión o desconexión de un circuito eléctrico durante un tiempo determinado.</p>
<b>UDP</b>	<p>Protocolo de Datagrama de Usuario (en inglés <i>User Datagram Protocol</i>) es un protocolo del nivel de transporte basado en el intercambio de datagramas. Permite el envío de datagramas a través de la red sin que se haya establecido previamente una conexión, y que el propio datagrama incorpora suficiente información de direccionamiento en su cabecera.</p>

## **ANEXOS**

## ANEXO 1



*Figura 96: Kit completo para trabajar con el modulo PINK*



*Figura 97: Circuito de prueba del modulo PINK*

## ANEXO 2

```
' {$STAMP BS2}
' {$PBASIC 2.5}

modo VAR Byte
NBVAR VAR Byte
n VAR Byte
email VAR Byte
cont3 VAR Word
time VAR Word
metros VAR Byte

SEROUT 8,396,["!NB0W03:0",CLS] 'inicializa

DO WHILE modo = 0
  IF IN14 = 1 THEN 'modo [alerta durante ausencia]
    modo = 1
    FOR N = 1 TO 10 'damos tiempo para abandonar la casa
      FREQOUT 4, 250, 3000 ' beep
      PAUSE 750
    NEXT
  ENDIF
LOOP

DO
  GOSUB PIR
LOOP

PIR:
  IF IN3 = 1 THEN
    cont3 = 1 'OJO alerta aproximación
    GOSUB EMAIL_BEEP 'envía email o beep
  ENDIF
  IF cont3 > 0 THEN
    SELECT cont3
```

```

CASE > 800
    cont3 = 0 ` vuelve a estado alerta
    SEROUT 8,396,["!NB0W03:0",CLS] 'inicializa después
de 120 segundos
    RETURN
CASE > 400
    SEROUT 8,396,["!NB0W03:3",CLS] 'sensor ha sido
activado hace más de 60 segundos-> mostrar huella
CASE > 200
    SEROUT 8,396,["!NB0W03:2",CLS] 'sensor ha sido
activado hace más de 30 segundos-> mostrar huella
CASE > 0
    SEROUT 8,396,["!NB0W03:1",CLS] 'sensor ha sido
activado
ENDSELECT
cont3 = cont3 +1 'contador en marcha (aunque no exista
aproximación)
PAUSE 10
ENDIF
RETURN
EMAIL_BEEP:
IF modo = 1 THEN FREQOUT 4, 250, 3000 ' beep
IF cont3 > 1 THEN 'envía email sólo en caso de que se
detecte un intruso en la casa
IF email = 0 THEN
    SEROUT 8,396,["!NB0WET:irma@tesis.com",CLS]
    SEROUT 8,396,["!NB0WEF:administrador@tesis.com",CLS]
    SEROUT 8,396,["!NB0WES:Mensaje de Alerta",CLS]
    SEROUT 8,396,["!NB0WEC:Alerta, intruso en la casa",CLS]
    SEROUT 8,396,["!NB0WEV:10.1.1.4",CLS]
    SEROUT 8,396,["!NB0SM"]
    email = 1 'email ya enviado
ENDIF
ENDIF
RETURN

```

## ANEXO 3

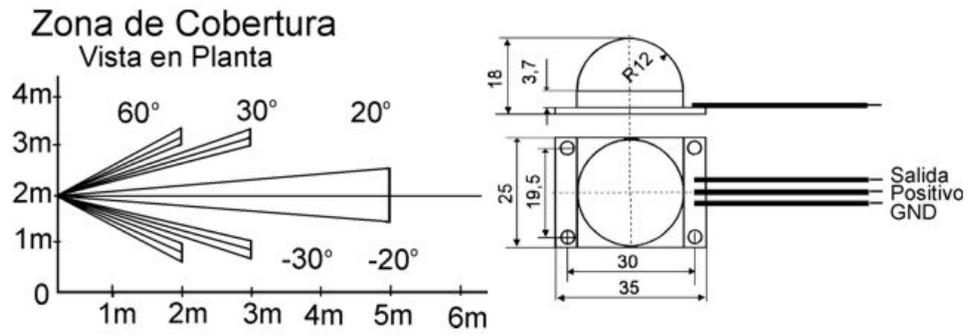


*Figura 98: Cable preparado para la descarga del programa en el BS2.*



*Figura 99: Conexión entre el BS2 y la PC a través del cable preparado*

## ANEXO 4



*Figura 100: Zona de cobertura del sensor PIR*



*Figura 101: Sensor PIR*

## ANEXO 5

### TABLA COMPARATIVA DE ALTERNATIVAS DE MICROCONTROLADORES

CARACTERISTICAS	PIC 16F877	ATMEGA 32	CSTAMP	BASIC STAMP2
Velocidad del procesador	20 MHz	16 MHz	40 MHz	20 MHz
Velocidad de ejecución del programa	-	-	10,000,000 instrucciones/seg	4000 instrucciones PBASIC/seg
RAM	368 x 8 Bytes	2048 Bytes	2 Kbytes	32 Bytes (6 E/S, 26 Variables)
EEPROM	256 x 8 Bytes	1024 Bytes	1 Kbyte	2 Kbytes; 500 instrucciones PBASIC
Número de pines E/S	40	32	-	16
Consumo de corriente	0,6 mA	-	19mA encendido, 0,7 µA apagado	3mA encendido, 50µA apagado
Comandos PBASIC	-	-	-	42
Paquete	40-pin DIP	40-pin DIP	48-pin DIP	24-pin DIP
Requisitos de alimentación	2.0 a 5.5 V	4,5 a 5,5 V	5 a 24 VDC	5,5 a 15 VDC (VIN), o 5 VDC (Vdd)
Comunicación	Paralelo (PSP)	Serie a paralelo	Serial (57600 baudios)	Serial (9600 baudios para la programación)
Dimensiones	-	-	60,96 x 27,94 x 10,16 mm	30,0 x 16,0 x 3,81 mm
Temperatura de funcionamiento	-	-40 a +85 ° C	-	-40 a +85 ° C
Precio	15 USD	65 USD	-	49 USD

# BIBLIOGRAFIA

## LIBROS

- 1.- **HUIDOBRO MOYA, J.M.** Domótica: edificios Inteligentes, Madrid: Tejedor, 2004
- 2.- **PALLÁS, R.** Microcontroladores: fundamentos y aplicaciones con PIC, Madrid: MARCOMBO, 2007
- 3.- **TANENBAUM, A.** Redes de Computadoras. 4ª. ed. México: Prentice Hall Hispanoamericana, 2003

## PAGINAS WEB

- 4.- **ARQHYS Architects Site.** Historia de la Domótica  
  
<http://www.arqhys.com/arquitectura/domotica-historia.html>  
  
2009/08/14
- 5.- **DOMOTICA VIVA, Soluciones Integrales para el Tercer Milenio.** Que es domótica  
  
<http://www.domoticaviva.com/somos.htm>  
  
2009/08/23
- 6.- **EINICIO.** Construcción Domótica  
  
<http://www.einicio.com/paginas/historia-de-la-domotica.html>  
  
2009/08/17
- 7.- **LORENZATI, M.** Pagina Web para el desarrollador de Sistemas Embebidos  
  
<http://www.marcelolorenzati.com.ar/>

2009/10/13

**8.- MONOGRAFIAS.** Introducción a los microcontroladores

<http://www.monografias.com/trabajos12/microco/microco.shtml>

2009/08/17

**9.- PARALLAX INC.** BASIC Stamp 2 Module

<http://www.parallax.com/StoreSearchResults/tabid/768/List/0/SortField/4/ProductID/1/Default.aspx?txtSearch=Basic+Stamp+2>

2009/09/10

**10.- \_\_\_\_\_.** PINK (Parallax Internet Netburner Kit)

<http://www.parallax.com/StoreSearchResults/tabid/768/List/0/SortField/4/ProductID/40/Default.aspx?txtSearch=PINK+module>

2009/09/10

**11.- \_\_\_\_\_.** PIR Sensor

<http://www.parallax.com/StoreSearchResults/tabid/768/List/0/SortField/4/ProductID/83/Default.aspx?txtSearch=PIR+sensor>

2009/09/10

**12.- PROYECTOS DOMOTICA.**

<http://www.proyectosdomotica.com/>

2009/08/19