

**Instituto Tecnológico de Costa Rica**

**Escuela de Ingeniería en Electrónica**



**Banco Central de Costa Rica  
BCCR**

**“Diseño e implementación de un prototipo de sistema de alarma para armarios de equipos de redes incorporado sobre una red TCP/IP existente”**

**Informe de Proyecto de Graduación para optar por el título de Ingeniero en Electrónica con el grado académico de Licenciatura**

**Julio Campos Jaén  
Gustavo Monge Ujueta**

**CARTAGO, Diciembre del 2003**

**INSTITUTO TECNOLÓGICO DE COSTA RICA  
ESCUELA DE INGENIERÍA EN ELECTRÓNICA  
PROYECTO DE GRADUACIÓN**

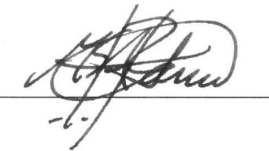
**TRIBUNAL EVALUADOR**

Ing. William María  
Profesor Asesor  
Escuela de Ingeniería en Electrónica  
I.T.C.R.

Firma: 



Ing. Mario Alabí Ramos  
Asesor por la empresa  
Banco Central de Costa Rica

Firma: 



Los miembros de este tribunal dan fe de que el presente trabajo de graduación ha sido aprobado y cumple con las normas establecidas por las Instituciones involucradas en la ejecución de este proyecto.

San José, 14 de noviembre del 2003  
Banco Central de Costa Rica, 9:00 a.m.

## Resumen

En este informe se presenta el desarrollo de un prototipo de sistema de alarma para armarios de equipos de redes para el Banco Central de Costa Rica.

El objetivo del desarrollo de este prototipo es informar al personal de seguridad acerca del posible acceso al armario de estos equipos, esto debido a cuestiones de seguridad, ya que personas no autorizadas podrían dañar la red o incluso prestarse para el robo de información de alta confidencialidad, ya que dicha red conforma el Sistema Interbancario de Pagos Electrónicos (SINPE) del banco, prestando servicios a diversas instituciones.

Entre los resultados importantes alcanzados se pueden mencionar las funciones implementadas en el módulo electrónico, donde es posible enviar y recibir paquetes por la red interna. El sistema maneja diferentes protocolos como el de dar respuesta a una petición ARP o al comando PING, y el envío de paquetes específicos por medio del protocolo IP y UDP. Con este último se implementó el envío de traps (mensajes SNMP), que es la forma de avisar a la estación central de cualquier alerta, usando un programa de monitoreo de dispositivos de red denominado "WhatsUp®", el cual fue configurado para enviar un mensaje por correo electrónico y radiolocalizador a los respectivos encargados del banco.

Como parte del proyecto se realizó una aplicación en Microsoft Visual Basic 6.0®. Mediante ese programa se configuran propiedades del módulo por puerto serie o por la red misma. Parámetros como las direcciones físicas e IP, los puertos utilizados, el rango de temperatura que permite el armario de redes y los mensajes de alerta de los paquetes a enviar son fácilmente modificables, haciendo de esta aplicación una gran herramienta para el correcto funcionamiento del sistema.

**Palabras clave:** Avisos por red, monitoreo remoto, SNMP, ARP, TRAPS, Visual Basic, UDP, WhatsUp, radiolocalizador, correo electrónico y control por redes ethernet.

## ABSTRACT

This abstract presents an alarm system proto- type development for network equipments rooms to the Costa Rica Central Bank.

The objective of this proto- type development is to inform the security personal about the possible access to closet where these equipments are, this is for security reasons, because non- authorized people could damage the network or steal top security information, due to this network is part of the electronics interbanking payments system (SINPE), offering its services to diverse institutions. Among the results obtained are implemented functions in the electronic module where is possible to send and receive packages in the intranet. The system handle differents protocols as to give response to a request ARP or the command PING, and the send of specifics packets by the protocol IP and UDP. This one implemented the traps sending (messages SNMP), which is the way to send an alarm to the central station using a monitoring dispositive network program named "WhatsUp®", which was configured to send a message by e- mail and beeper to the people in charge in the bank.

As part of the project mentioned a Microsoft Visual Basic 6.0® application was made. Through this program are configured properties of the modules by serial port o the network itself. Parameters such as the physical addresses an IP, the serial ports used, the temperature range which made possible the network closet and the packages alarm messages to send, are easy configured, making this applications a useful tool the appropriate system functioning.

**Key words:** Announcements by networks, remote monitoring, SNMP, ARP, TRAPS, Visual Basic, UDP, WhatsUP, radiolocalizador, electronic mail and control by ethernet networks.

## **Dedicatoria**

*A Dios, por brindarnos la fortaleza  
necesaria por culminar con éxito  
el presente proyecto.*

*A nuestros padres...  
Por todo el esfuerzo y apoyo brindado  
durante los años de estudio.*

## **Agradecimiento**

Agradecemos profundamente todo el apoyo y la colaboración brindada por todo el personal docente de la Escuela de Electrónica.

A los ingenieros Carlos Badilla, Pedro Murillo y Luis Paulino Méndez por su colaboración en el transcurso de nuestra vida académica.

A los señores ingenieros Mario Alabí, Luis Carballo, Omar Brenes y demás personal del Banco Central de Costa Rica por su apoyo durante la realización del proyecto.

## ÍNDICE GENERAL

<b>CAPÍTULO 1</b>	
<b>INTRODUCCIÓN.....</b>	<b>1</b>
<b>1.1 Descripción de la empresa .....</b>	<b>1</b>
<b>1.2 Definición del problema y su importancia .....</b>	<b>3</b>
<b>1.3 Objetivos .....</b>	<b>5</b>
<b>CAPÍTULO 2</b>	
<b>ANTECEDENTES .....</b>	<b>6</b>
<b>2.1 Estudio del problema a resolver .....</b>	<b>6</b>
<b>2.2 Requerimientos de la empresa .....</b>	<b>6</b>
<b>2.3 Solución propuesta.....</b>	<b>7</b>
<b>CAPÍTULO 3</b>	
<b>PROCEDIMIENTO METODOLÓGICO.....</b>	<b>9</b>
<b>CAPÍTULO 4</b>	
<b>DESCRIPCIÓN DEL HARDWARE UTILIZADO .....</b>	<b>13</b>
<b>4.1 Programador de microcontroladores de la familia Microchip PICSTART Plus .....</b>	<b>13</b>
<b>4.2 Computadora.....</b>	<b>14</b>
<b>CAPÍTULO 5</b>	
<b>DESCRIPCIÓN DEL SOFTWARE UTILIZADO.....</b>	<b>15</b>
<b>5.1 MPLAB IDE v6.30® .....</b>	<b>15</b>
<b>5.2 PCW Compiler® .....</b>	<b>16</b>
<b>5.3 Ethereal.....</b>	<b>17</b>
<b>5.4 WhatsUp®.....</b>	<b>18</b>

<b>CAPÍTULO 6</b>	
<b>ANÁLISIS Y RESULTADOS</b> .....	<b>21</b>
<b>6.1 Explicación del diseño</b> .....	<b>21</b>
<b>6.1.1 Hardware</b> .....	<b>21</b>
<b>6.1.2 Software</b> .....	<b>28</b>
<b>6.2 Alcances y limitaciones</b> .....	<b>47</b>
<b>CAPÍTULO 7</b>	
<b>CONCLUSIONES Y RECOMENDACIONES</b> .....	<b>48</b>
<b>7.1 Conclusiones</b> .....	<b>48</b>
<b>7.2 Recomendaciones</b> .....	<b>49</b>
<b>BIBLIOGRAFÍA</b> .....	<b>50</b>
<b>APÉNDICES</b> .....	<b>51</b>
<b>Apéndice A.1 Glosario</b> .....	<b>51</b>
<b>Apéndice A.2 Abreviaturas</b> .....	<b>53</b>
<b>Apéndice A.3 Manual de usuario del SMR- NET 8900</b> .....	<b>54</b>
<b>A.3.1 Especificaciones del sistema.</b> .....	<b>54</b>
<b>A.3.2 Instalación.</b> .....	<b>54</b>
<b>A.3.3 Instalación del software para la configuración del módulo.</b> .....	<b>54</b>
<b>A.3.4 Configuración del módulo por el puerto serie.</b> .....	<b>55</b>
<b>A.3.5 Configuración del módulo por red.</b> .....	<b>59</b>
<b>ANEXOS</b> .....	<b>62</b>
<b>Anexo B.1 Especificaciones y diagrama del controlador CS8900A.</b> .....	<b>62</b>
<b>Anexo B.2 Especificaciones técnicas y diagrama del microcontrolador PIC16F877</b>	<b>64</b>
<b>Anexo B.3 Diagrama del MAX233 para comunicación por puerto serie.</b> .....	<b>65</b>
<b>Anexo B.4 Especificaciones técnicas y diagrama del convertidor de temperatura a voltaje de alta precisión</b> .....	<b>66</b>



## ÍNDICE DE FIGURAS

<b>Figura 1.1</b>	Diagrama de la red existente con el Banco Central de Costa Rica.....	3
<b>Figura 1.2</b>	Distribución del equipo en cada armario donde el banco presta servicios.....	4
<b>Figura 3.1</b>	Sistema de alarma SMR- NET 8900 terminado, parte frontal. ....	11
<b>Figura 3.2</b>	Sistema de alarma SMR- NET 8900 instalado en un armario de equipo de redes del banco. ....	12
<b>Figura 4.1</b>	Programador PICSTART Plus de Microchip®.....	13
<b>Figura 5.1</b>	Vista de pantalla para el programa de desarrollo MPLAB v6.30®.....	16
<b>Figura 5.2</b>	Vista de pantalla para el programa de desarrollo PCW Compiler IDE ver 3.4®. ....	16
<b>Figura 5.3</b>	Vista de pantalla para el programa analizador de red Ethereal. ....	17
<b>Figura 5.4</b>	Vista de pantalla del programa WhatsUp® en modo de mapeo.....	18
<b>Figura 5.5</b>	Diálogo de configuración general de un dispositivo de red colocado en el programa WhatsUp®. ....	19
<b>Figura 5.6</b>	Diálogo de configuración de eventos para un dispositivo de red colocado en el programa WhatsUp®. ....	19
<b>Figura 5.7</b>	Diálogo de configuración de notificación por correo electrónico ante un evento de un dispositivo de red colocado en el programa WhatsUp.....	20
<b>Figura 6.1</b>	Diagrama lógico de conexión entre el microcontrolador PIC16F877 y el controlador ethernet CS8900A.....	21
<b>Figura 6.2</b>	Módulo adquirido para el desarrollo del proyecto.....	22
<b>Figura 6.3</b>	Diagrama de conexión de la tarjeta “Easy Ethernet CS8900A”.....	24
<b>Figura 6.4</b>	Diagrama de flujo del procedimiento setipaddr(). ....	31
<b>Figura 6.6</b>	Diagrama de Flujo para el procedimiento RpppL. ....	32
<b>Figura 6.7</b>	Diagrama de Flujo para los procedimientos WpppL y WpppL.....	33
<b>Figura 6.8</b>	Diagrama de Flujo para el procedimiento UDP. ....	34

<b>Figura 6.9</b>	Manejo de interrupción para recepción de tramas por medio de Service_ISQ.....	35
<b>Figura 6.10</b>	Diagrama de flujo para el procedimiento Get_Frame.....	36
<b>Figura 6.11</b>	Diagrama de Flujo para el procedimiento SNMP_TX.....	37
<b>Figura 6.12</b>	Diagrama de flujo para el procedimiento Sensado.....	38
<b>Figura 6.13</b>	Diagrama de Flujo para el procedimiento Puerto Serie.....	40
<b>Figura 6.14</b>	Esquema de la trama realizada usando una unidad de datos TRAP del protocolo SNMP.....	42
<b>Figura 6.15</b>	Interfaz realizada en Visual Basic 6.0®.....	44
<b>Figura 6.16</b>	Cuadro de propiedades del control MSComm de Visual Basic®.....	44
<b>Figura 6.17</b>	Diálogo de propiedades del control Winsock de Visual Basic®.....	45
<b>Figura A.1</b>	Vista en pantalla del programa de configuración del módulo SMR-NET 8900..	55
<b>Figura A.2</b>	Ventana que permite introducir la dirección IP del sistema.....	56
<b>Figura A.3</b>	Ventana que permite introducir la dirección MAC destino.....	57
<b>Figura A.4</b>	Vista en pantalla de la ventana para la configuración de los puertos.....	57
<b>Figura A.5</b>	Vista en pantalla de la ventana para la variación del rango de temperatura.....	58
<b>Figura A.6</b>	Ventana que permite visualizar la configuración realizada al sistema, por puerto serie.....	58
<b>Figura A.7</b>	Ventana para introducir la dirección IP remota y los puertos del módulo.....	59
<b>Figura A.8</b>	Ventana para verificar la correcta configuración del sistema de alarma, por red.	60
<b>Figura A.9</b>	Ventana que permite enviar traps de prueba por la red.....	61
<b>Figura A.10</b>	Ventana que permite cambiar la comunidad de los traps.....	61
<b>Figura B1.1</b>	Características y descripción general del controlador ethernet CS8900A.....	62

<b>Figura B1.2</b>	Diagrama del controlador ethernet CS8900A.....	63
<b>Figura B2.1</b>	Características y descripción del microcontrolador PIC 16F877 .....	64
<b>Figura B2.2</b>	Características y descripción del microcontrolador PIC 16F877 .....	64
<b>Figura B3.1</b>	Diagrama del manejador MAX233.....	65
<b>Figura B4.1</b>	Características y descripción del sensor de temperatura TC1046 .....	66
<b>Figura B4.2</b>	Curva característica del sensor de temperatura TC1046.....	67

## ÍNDICE DE TABLAS

<b>Tabla 6.1</b>	Mapa de Memoria para el Controlador ethernet en modo I/O .....	25
<b>Tabla 6.2</b>	Variables utilizadas en la programación del microcontrolador .....	29

## **CAPÍTULO 1**

### **INTRODUCCIÓN**

---

#### **1.1 Descripción de la empresa**

##### a) Descripción general

El Banco Central de Costa Rica fue creado en 1950 mediante la ley 1130 y es el órgano financiero con mayor autoridad que permite el orden ante la intensificación de la actividad económica y bancaria del país. Este cumple funciones como:

- el mantenimiento del valor de la moneda;
- administración de las reservas monetarias internacionales;
- la gestión como consejero y banco cajero del estado;
- emisión de billetes y monedas.

Para cumplir con los diversos objetivos tales como un sistema de intermediación financiera estable, eficiente y competitivo, promover la eficiencia del sistema de pagos internos y externos, promover un ordenado desarrollo de la economía costarricense, entre otros, se ha hecho necesario modernizar su sistema de funcionamiento, tendiendo a nuevas tecnologías como lo son las redes de computadoras y sistemas de seguridad de mayor confiabilidad.

La estructura del banco central de Costa Rica está conformada por los siguientes niveles jerárquicos:

- Nivel Superior: nivel ejecutivo del banco, con alto grado de autoridad.
- Nivel Divisional: responsable de las funciones que procuran alcanzar los objetivos de la institución. En este nivel existen siete divisiones:
  - económica;
  - servicios financieros;
  - gestión de activos y pasivos;
  - administrativa;
  - secretaria general;
  - asesoría jurídica;
  - tecnologías de información.
- Nivel Departamental: ejecutan funciones de tipo sustantivo, pero también de apoyo administrativo.
- Nivel Seccional: para realizar funciones de aspecto técnico.

b) Descripción del departamento donde se realizó el proyecto de graduación

El proyecto se realizó en el departamento de “Tecnologías del Sistema de Pagos”, propiamente en el sector de telecomunicaciones. Es un departamento del banco que actúa en forma individual, bajo la autoridad de la división de servicios en el área de telecomunicaciones, en la cual laboran cinco personas, la mayoría ingenieros en electrónica. La persona de mayor autoridad es el Ing. Mario Alabí, el cual será el asesor por parte de la empresa.

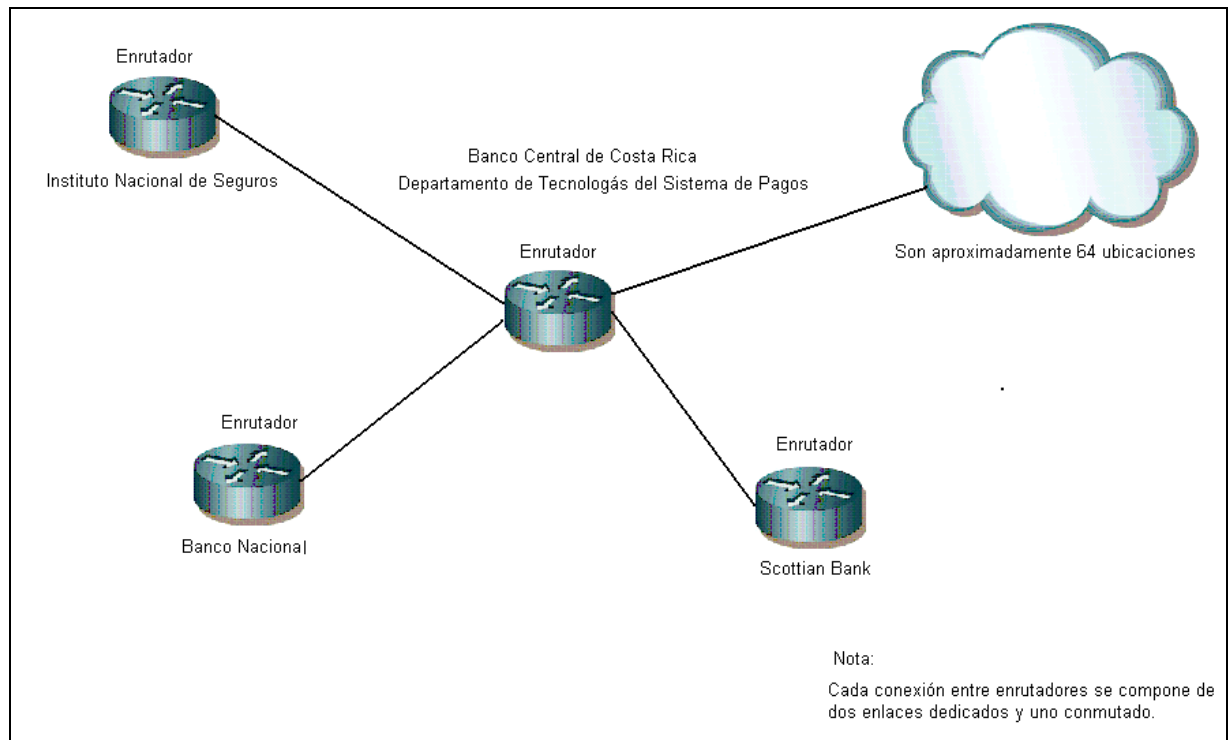
Esta sección es la encargada de todos los sistemas de comunicación del Banco Central para el sistema de pagos, sus funciones se pueden dividir en 4 áreas principales:

- a. cableado estructurado;
- b. redes locales;
- c. red WAN;
- d. centrales telefónicas.

En general, este departamento se encarga del soporte y mantenimiento de la red del Sistema Financiero Nacional.

## 1.2 Definición del problema y su importancia

El Banco Central de Costa Rica presta servicios a aproximadamente 64 ubicaciones, entendiéndose estas como cajeros ATH, bancos, Ministerio de Hacienda, la C.C.S.S, etc. Esto conforma el Sistema Interbancario de Pagos Electrónicos (SINPE) que es administrado por el departamento de Tecnologías del Sistema de Pagos del Banco Central de Costa Rica. En general puede verse como una red de nodos como se muestra en la figura 1.1.

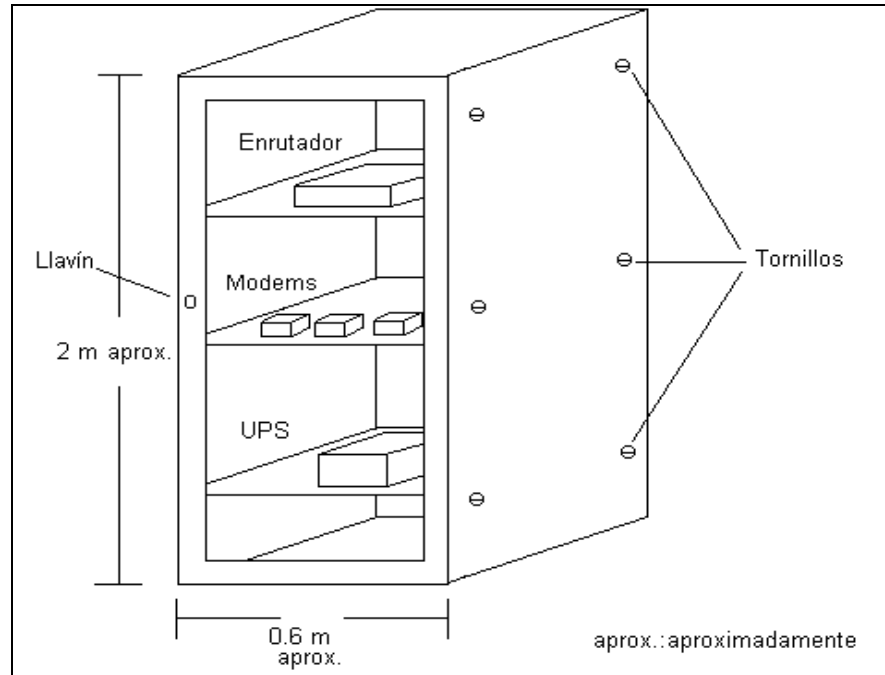


**Figura 1.1** Diagrama de la red existente con el Banco Central de Costa Rica

El equipo de redes que se ubica en cada lugar es administrado por el Banco Central de Costa Rica. Para proteger este equipo se usan armarios similares al que se muestra en la figura 1.2. Cada uno de estos compartimentos guarda, entre otras cosas, un enrutador, tres modems y una UPS<sup>1</sup>. Las puertas delantera y la trasera son hechas de vidrio, mientras que las paredes laterales son de metal, soportadas por tornillos. Para poder acceder al equipo se debe de abrir la puerta con una llave, que es exclusiva del departamento del Banco Central.

---

<sup>1</sup> Uninterrupted Power Supply (fuente de poder ininterrumpida).



**Figura 1.2** Distribución del equipo en cada armario donde el banco presta servicios.

El acceso al equipo por personas no autorizadas se podría realizar de diversas maneras, ya sea por las paredes laterales, o con la copia de la llave cuando se presta por situaciones especiales, con lo cual existe la posibilidad de generar daños al sistema o alguna posible captura de datos de alta confidencialidad que transitan en dicha red. Según información del personal del área de telecomunicaciones, se sabe que hay individuos que tienen un duplicado de estas llaves.

A causa de estos motivos no se tiene un buen control de seguridad del equipo que administra el banco, ya que la entrada al equipo por personas sin autorización se sabría sólo cuando se generen problemas en la red, por daños del compartimiento donde se guarda el equipo, o en el peor de los casos por uso de los datos confidenciales. Lo ideal sería un sistema que informe antes de que se generen problemas graves.

El problema es de índole económico, ya que se pretende cuidar una red que es de suma importancia para el banco, la cual debe mantenerse siempre activa, porque cualquier desconexión, por mínima que esta sea, podría generar grandes pérdidas de tipo monetario, además de la protección de la información de alta confidencialidad cuya manipulación indebida puede prestarse para posibles fraudes o robos. Estas situaciones aún no se han presentado, pero la idea del proyecto surge como una precaución para evitar cualquier delito que se pueda presentar en el futuro.



### **1.3 Objetivos**

#### **1.3.1 Objetivo general**

Realizar un prototipo de sistema de alarma para armarios de equipos de redes incorporado sobre una red TCP/IP existente.

#### **1.3.2 Objetivos específicos**

1. Conocer la estructura de las tramas de red SNMP y de los encabezados de los protocolos UDP e IP.
2. Analizar el funcionamiento de los componentes de la tarjeta conformada por el microcontrolador 16F877 y el controlador ethernet Cirrus CS8900A.
3. Realizar las rutinas del microcontrolador para:
  - I. Crear el paquete SNMP con un trap como unidad de datos.
  - II. Realizar los encabezados UDP e IP.
  - III. Ensamblar la trama IEEE 802.3 ethernet.
  - IV. Manejar el controlador ethernet.
  - V. Realizar la lectura de los contactos magnéticos y del sensor de temperatura.
  - VI. Realizar la comunicación por puerto serie y por red para el cambio de configuración del sistema.
4. Implementar el sistema de sensores y alarma sonora para el armario del equipo de redes.
5. Desarrollar un programa en lenguaje Visual Basic® para el cambio y visualización de la configuración del módulo diseñado a través del puerto serie y por red.
6. Configurar el programa de computadora en la estación de administración para el envío de mensajes de alerta por medio de radiolocalizador y correo electrónico.
7. Realizar pruebas de conexión del enrutador con el sistema realizado.

## **CAPÍTULO 2**

### **ANTECEDENTES**

---

#### **2.1 Estudio del problema a resolver**

Los equipos de redes que se desean vigilar están ubicados en lugares distantes de la estación central, ubicada en el Banco Central de Costa Rica, dentro de armarios con cerradura y paredes laterales desmontables, las cuales, como se mencionó anteriormente, no son muy seguras. Por lo tanto, el sistema desarrollado debe poder enviar avisos de alerta de forma remota, preferiblemente sobre TCP/IP.

Además, como en dichos armarios hay un enrutador, se tiene a disposición una conexión ethernet, la cual puede ser utilizada para que el sistema de alarma que se desarrolle pueda conectarse a red. El sistema realizado debe ser capaz de adecuarse a la red interna del banco, ya que el envío de alertas se debe realizar hacia la estación central, donde el personal del área de tecnología de pagos debe ser informado instantáneamente de cualquier irregularidad.

A través del sistema realizado también se debe poder enviar cierta información, tal como la identificación del sensor activado, entidad afectada, y variaciones de la temperatura del armario que exceda el rango permitido.

Algo muy importante es que este sistema debe ofrecer la posibilidad de ser manejado y monitoreado desde la estación central, en el banco, y debe tener un comportamiento similar al de un dispositivo de red de área local, es decir, ser identificado lógicamente y físicamente.

#### **2.2 Requerimientos de la empresa**

El personal encargado de la administración de los armarios requería que el sistema desarrollado tuviese modularidad, a fin de poder ser colocado en cada ubicación con facilidad. Tenía que ser instalado en la red interna del banco, de forma que cada paquete generado por dicho sistema fuese enviado desde la entidad infringida hasta la estación central de monitoreo.

En general, los requerimientos necesarios para la correcta solución del problema fueron los siguientes:

- Enviar un mensaje de alerta por correo electrónico y radiolocalizador mediante el uso del programa WhatsUp®. Este es el software que se utiliza en el departamento de Sistemas de Tecnología de Pagos para observar el estado del equipo de redes de cada entidad.

- Generar un aviso sonoro en el lugar de origen de la alerta, de manera que algún encargado de seguridad cercano al armario acuda de inmediato al lugar.
- Identificación del lugar geográfico de origen de la alerta. Debe de recordarse que hay entidades que se encuentran a varios kilómetros de distancia del banco.
- Identificación del lugar del armario donde se infringió el acceso, es decir, alguna puerta o pared.
- Envío de una alerta cuando el valor de temperatura del armario de equipo de redes esté fuera del rango permitido.
- Fácil configuración de los parámetros del módulo que se diseñe Esta configuración debe poder realizarse de manera presencial o de manera remota. Entre los parámetros que la empresa exigió configurables se encuentran:
  - i. Direcciones lógicas y físicas de red.
  - ii. Puertos.
  - iii. Rango de temperatura del interior del armario.
  - IV. Mensajes para cada aviso.

### **2.3 Solución propuesta**

Para resolver el problema se planteó realizar un aviso directo del sistema electrónico desarrollado a la estación central. Para ello se hace uso del SNMP, (protocolo de administración de red simple). La razón de utilizarlo es porque en el banco se usa este protocolo para administrar los dispositivos de la red interna, en cada entidad. En este protocolo se define la estructura de mensajes de aviso llamados traps, los cuales son mensajes que se envían por la red cuando ocurre una situación de alerta en el lugar de origen. Estos utilizan el protocolo IP y el protocolo UDP como capas de red y transporte respectivamente.

De esta manera, lo que se pretende es que cada sensor ubicado en el armario del equipo de redes se comporte como una variable, que al cambiar su estado, hacen reaccionar al sistema para el envío del trap con un mensaje específico de dicha variable. Por lo tanto, se propone implementar el trap, utilizar el protocolo SNMP para que dicho trap sea su mensaje, y enviarlo por red, adjuntándole previamente los encabezados UDP e IP.

La recepción del trap en la estación central se hace por medio del programa WhatsUp®, el cual se debe configurar de forma adecuada para recibir los mensajes por el puerto UDP 162 y tomar las acciones del caso, como mandar un mensaje por correo electrónico y por radiolocalizador a los encargados del banco.

Para poder realizar la configuración del módulo se implementó un programa en lenguaje Visual Basic® que permitió, a través de una comunicación serie con el microcontrolador, o por red, cambiar parámetros importantes como las direcciones IP, las direcciones físicas y los puertos de envío y recepción. Así mismo, como parte de la solución, se propuso dotar a este programa de una herramienta de diagnóstico, que permitiera a los encargados del banco monitorear al sistema de alarma para verificar su adecuado funcionamiento.

## **CAPÍTULO 3**

### **PROCEDIMIENTO METODOLÓGICO**

---

Luego de una exhaustiva investigación se logró determinar que para realizar el proyecto era adecuado implementar un sistema con dos partes principales: una que pudiera tomar información de los sensores del armario de equipo de redes y otra con la posibilidad de sacar paquetes para su posterior enrutamiento a través de la red interna del banco. Se decidió adquirir una tarjeta que cumpliera con estos requerimientos, ya que estaba conformada por un microcontrolador de la familia Microchip® y por un controlador de red de la casa fabricante Cirrus Logic. Dentro de las especificaciones del controlador estaban que trabajaba con el estándar IEEE 802.3 ethernet para el envío y recepción de paquetes, por lo que era propicio para colocarlo en una red.

Debe de aclararse que no se trataba de una tarjeta para configurar, ya que se necesitaba programar el microcontrolador adecuándolo a la necesidad que se le quisiese dar.

Para llevar a cabo la idea propuesta en la solución se inició realizando un estudio del protocolo SNMP. Era de suma importancia investigar la conformación de los paquetes de un datagrama SNMP con un trap como unidad de datos, ya que se requería armar toda la trama, y el mensaje principal era el trap. Se debía iniciar desde las capas inferiores del modelo de red TCP/IP, añadiendo los respectivos encabezados UDP e IP que le permitiesen al trap viajar desde una determinada entidad hasta la estación central, ubicada en el Banco Central. Además era necesario para ello conocer la conformación de una trama IEEE 802.3 ethernet. Todo el paquete se debía construir utilizando programación.

Se realizaron varias pruebas utilizando el programa Ethereal de visualización de paquetes, para observar los campos de que consta un mensaje SNMP. Para ello, se usaron emuladores implementados en Visual Basic®.

Seguidamente se debía conocer el funcionamiento del controlador Cirrus CS8900A. Para ello se utilizó el manual que la casa fabricante pone a disposición en Internet en su página Web. Se comenzó analizando la estructura y la configuración de los registros internos y señales de entrada y salida, así como el procedimiento seguido durante una recepción y una transmisión de paquetes en una red. También se determinó la forma de dotar al controlador de sus propias direcciones MAC e IP.

Una vez que se tenía conocimiento total del controlador y de la trama SNMP se comenzó a realizar la extensa programación que cumpliera con el manejo del CS8900A y con la implementación y transmisión de los traps. El microcontrolador

que la tarjeta traía por defecto era el PIC 16F877. Por lo tanto, no se debió hacer investigación de los microcontroladores existentes que pudiesen manejar el controlador ethernet, pero si se tuvieron que analizar los

puertos y la memoria disponible. La programación se realizó en lenguaje C, utilizando el compilador PCW® y la herramienta MPLAB IDE v6.30® que Microchip® tiene para la programación de sus distintos microcontroladores.

Las primeras pruebas permitieron ajustar cada uno de los tamaños de los campos del paquete, de manera que pudiesen ser reconocidos y recibidos en una estación vecina. Uno de los cálculos más complicados fue el de las dos sumas de verificación, ya que sin la suma de chequeo correcto no se podrían recibir los traps, por la sencilla razón de que el mensaje SNMP no era reconocido por el computador remoto.

Los primeros traps correctos se recibieron utilizando un programa hecho en Visual Basic®, que simulaba una estación agente.

A continuación se procedió a conocer el programa WhatsUp® que se utiliza en el banco para el monitoreo de los dispositivos de red de las 64 ubicaciones o entidades. Se configuró que las alertas llegasen a una cuenta de correo establecida, y que cada trap llevase un mensaje de aviso. De esta manera, ya se podían recibir los traps con el software de la estación central.

Posteriormente se implementaron las rutinas para permitirle al microcontrolador recibir las señales de los contactos magnéticos y del sensor de temperatura. Este último se realizó un poco tarde, ya que fue uno de los requerimientos que el personal del Departamento de Tecnología de Pagos exigió en una etapa avanzada del proyecto. Casi en paralelo con estas rutinas también se realizó la programación necesaria para que el microcontrolador pudiese comunicarse por puerto serie con un computador, con la finalidad de poder variar la configuración de la tarjeta sin necesidad de tener que programar el microcontrolador usando herramientas de Microchip®. Básicamente se realizaron rutinas de programación para el manejo de comandos y parámetros recibidos por el microcontrolador a través del puerto serie.

Una vez realizada la programación del microcontrolador, se inició la implementación de los procedimientos en lenguaje Visual Basic® poder recibir y enviar los parámetros de configuración del microcontrolador. Primero se diseñaron de los formularios de interfaz con el usuario, y luego se trabajó en la comunicación serie y por red del sistema. Se asignaron posiciones en la memoria ROM del microcontrolador para que variables como las direcciones MAC e IP, así como puertos y la temperatura pudiesen variarse y mantenerse una vez configurados. El sistema de alarma se puede observar en las figuras 3.1 y 3.2, ya con el acabado final.



**Figura 3.1** Sistema de alarma SMR- NET 8900 terminado, parte frontal.

Como último paso de la metodología, se aplicaron pruebas finales al sistema completo mediante activación de los sensores y la visualización del evento en el programa de computadora WhatsUp®. En la figura 3.2 se observa un armario de equipo de redes con el sistema ya instalado.



**Figura 3.2** Sistema de alarma SMR- NET 8900 instalado en un armario de equipo de redes del banco.



## CAPÍTULO 4

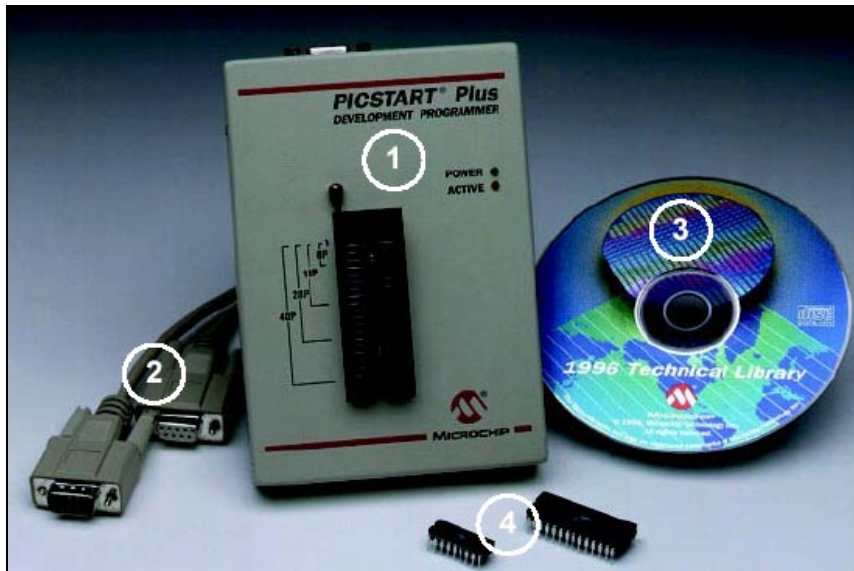
### DESCRIPCIÓN DEL HARDWARE UTILIZADO

---

#### 4.1 Programador de microcontroladores de la familia Microchip PICSTART Plus

Este hardware se utilizó para la programación del microcontrolador PIC16F877. En la figura mostrada se pueden ver las partes que conforman todo el paquete:

1. Programador.
2. Cable serial para conexión con el computador.
3. Disco compacto con el software para programación.
4. Diversos microcontroladores que se pueden programar con éste hardware.



**Figura 4.1** Programador PICSTART Plus de Microchip®.

Entre las principales características de éste programador están:

- Conexión por puerto serie RS- 232 con la computadora;
- Utilización de software MPLAB de Microchip® para la programación de los microcontroladores de la familia de Microchip®. Este viene incluido con el programador, y trabaja bajo la plataforma de Windows de Microsoft;

- Permite leer, programar y verificar la memoria de datos y la programación;
- Permite leer, programar y verificar toda la configuración de bits del programa realizado;
- Programa y verifica por rangos de dirección;
- Permite borrar el contenido del microcontrolador a programar;
- Posee una base que permite programar diversos microcontroladores sin dañar en lo más mínimo sus patillas;
- Posee LED indicador de programación en curso;
- Requiere nueve voltios para su alimentación.

#### **4.2 Computadora**

Este hardware fue indispensable para el desarrollo de este proyecto, ya que se necesitó para ejecutar los programas necesarios para la programación, edición y verificación del programa del microcontrolador, y para visualizar mediante diversas pruebas el envío y recepción de paquetes por parte del controlador ethernet CS8900A. Algunas de las características de la computadora utilizada son las siguientes:

- Pentium II con procesador de 350 MHz;
- disco duro con capacidad de 12 gigabytes;
- 256 megabytes de memoria RAM;
- conexión a Internet disponible;
- puertos USB, paralelo y serie debidamente habilitados;
- unidad de CDROM con velocidad 32X.

## **CAPÍTULO 5:**

### **DESCRIPCIÓN DEL SOFTWARE UTILIZADO**

---

#### **5.1 MPLAB IDE v6.30®**

Esta es la herramienta de desarrollo para programación del microcontrolador PIC escogido de la casa fabricante Microchip®. En la figura 5.1 se puede ver la interfaz gráfica con el usuario del MPLAB v6.30®.

Entre las características de este software se destaca:

- Corre bajo sistemas operativos Windows 98 SE, Windows 2000 SP2, Windows NT SP6, Windows ME, Windows XP Pro;
- soporta sistemas de desarrollo de microchip como PICSTART Plus, PRO MATE II, MPLAB ICD 2, MPLAB ICE 2000 y MPLAB ICE 4000;
- soporta los microcontroladores PIC16CXXX, PIC16FXXX, PIC18CXXX, PIC18FXXX, 12CXXX y 12FXXX;
- soporta herramientas de desarrollo complementarias como Hi-Tech, IAR, CCS, Byte Craft, MicroEngineering Labs y B. Knudsen compilers;
- soporta memorias EEPROM seriales;
- soporta seguridad de datos del producto.

En el proyecto este programa se utiliza tanto para la programación del microcontrolador 16F877 como para la simulación con el sistema MPLAB SIM.

Los pasos para la programación del PIC son los siguientes:

1. Realizar el programa en compilador PCW y generar el archivo \*.hex.
2. En la pestaña Configure, opción Select Device, escoger el microcontrolador usado. Ir a la pestaña File.
3. Escoger la opción de Import.
4. Seleccionar el archivo.
5. Ir a la pestaña Programmer, Select Programmer, y activar la opción de PICSTART PLUS.
6. En la pestaña Programmer escoger la opción Program.

Los pasos para la simulación son los siguientes:

1. Crear un proyecto en la opción Project, Project Wizard.
2. En la pestaña View y en la opción Watch, escoger las variables a analizar.
3. En la pestaña Debug, Select Tool, escoger la opción MPLAB SIM.
4. Dar la opción Build all, en la pestaña Debugger.
5. Con la tecla F7 se corre el programa paso a paso.



### 5.3 Ethereal

Este programa pertenece a los denominados “sniffer”, debido al hecho de que es capaz de capturar las tramas que viajan en una red de área local (LAN). Se utiliza en el proyecto para la verificación de los bits que se envían por parte del dispositivo diseñado a la red. Durante el desarrollo del proyecto fue útil porque permitió verificar que los diversos campos de la trama SNMP, donde se incluye el trap, tuviesen los valores adecuados para ser enviados debidamente por la red. En la siguiente figura se observa el programa durante una captura de paquetes.

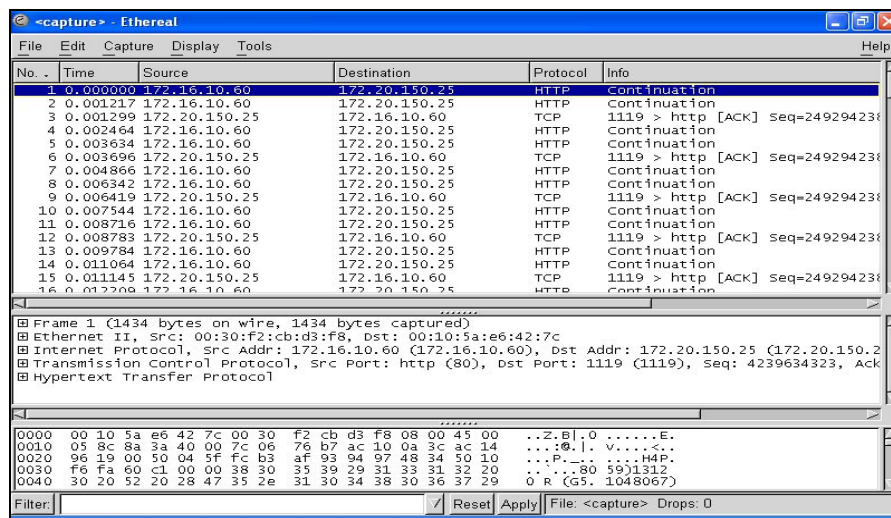
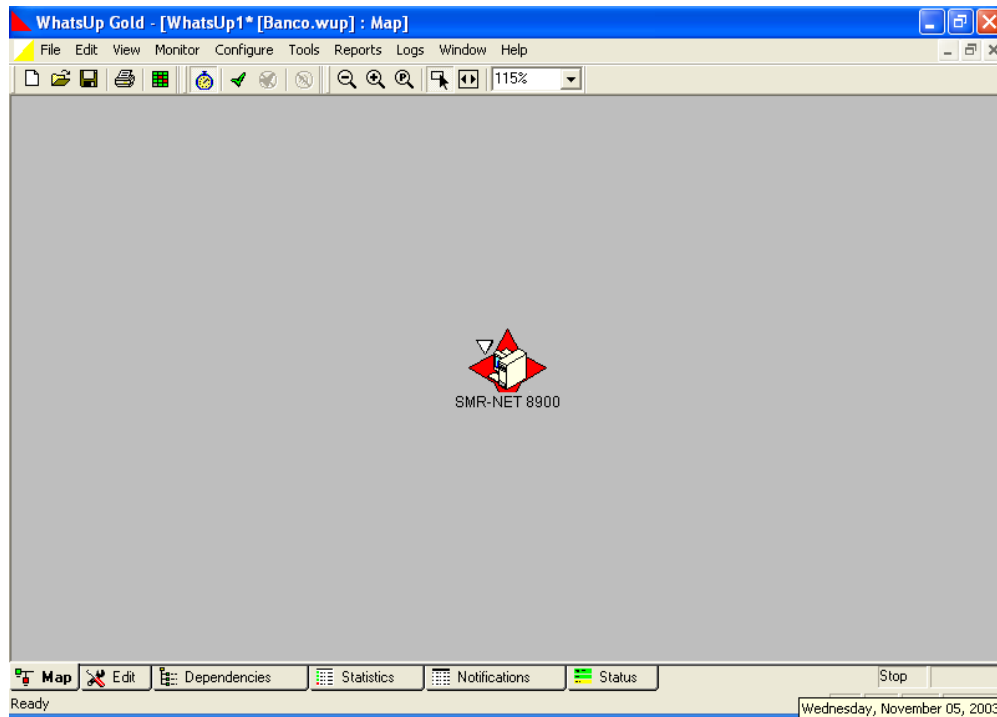


Figura 5.3 Vista de pantalla para el programa analizador de red Ethereal.

## 5.4 WhatsUp®

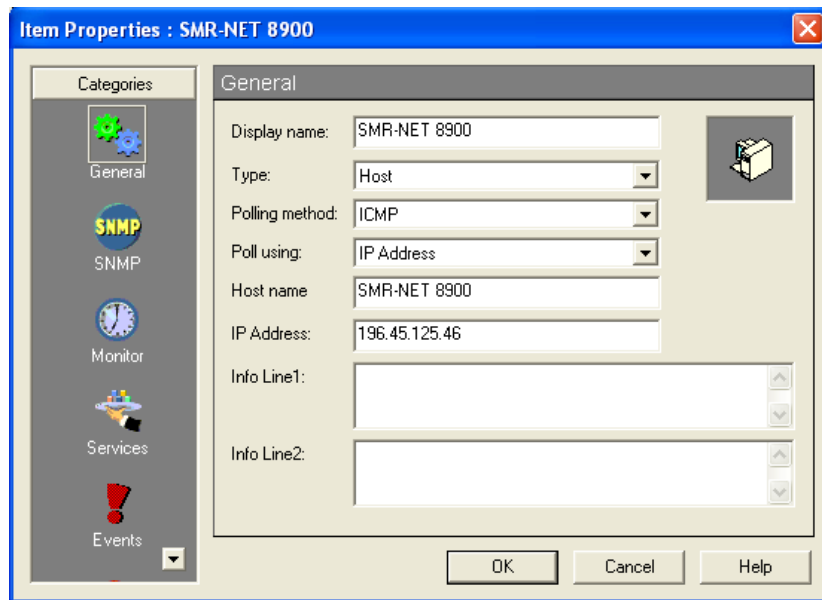
Este software, realizado por la empresa IPSWITCH®, permite monitorear y administrar fácilmente redes de computadoras de gran magnitud. El módulo realizado fue colocado como cualquier otro computador de la red interna del banco como se muestra en la figura 5.4.



**Figura 5.4** Vista de pantalla del programa WhatsUp® en modo de mapeo.

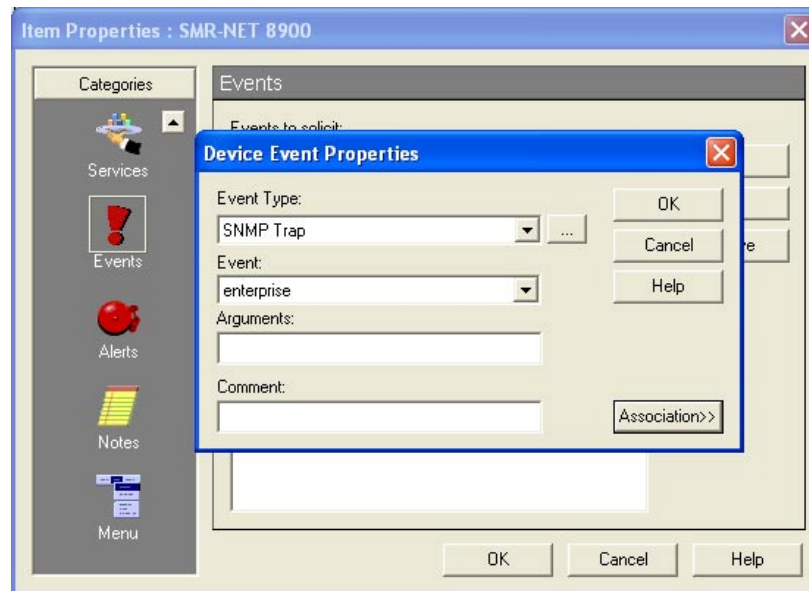
Cuando el módulo envía un trap de alerta al computador con el programa WhatsUp® instalado, este envía una notificación a través de correo electrónico y radiolocalizador a los encargados respectivos.

Al entrar en las propiedades del dispositivo creado en WhatsUp®, se entra al diálogo mostrado en la figura 5.5. En el icono general se configura la dirección IP del módulo, el nombre a mostrar, el método de monitoreo, entre otras opciones.

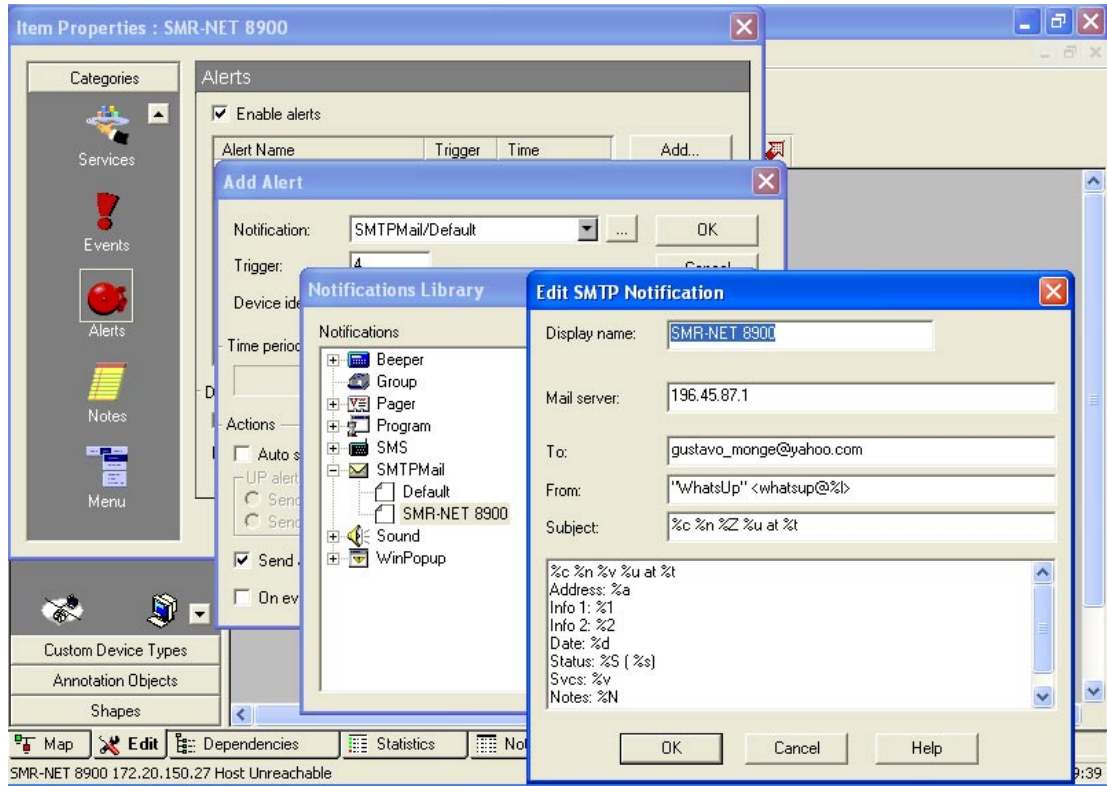


**Figura 5.5** Diálogo de configuración general de un dispositivo de red colocado en el programa WhatsUp®.

En la figura 5.6 se muestran las opciones del icono Events donde se le indica al programa el evento que se debe recibir del módulo para generar la notificación a los encargados respectivos. En estas opciones se escoge como tipo de evento el valor “Trap SNMP”, y como evento el valor “enterprise”, que identifica al trap generado como una variable propia del módulo.



**Figura 5.6** Diálogo de configuración de eventos para un dispositivo de red colocado en el programa WhatsUp®.



**Figura 5.7** Diálogo de configuración de notificación por correo electrónico ante un evento de un dispositivo de red colocado en el programa WhatsUp.

En la figura 5.7 se muestra la configuración de la notificación por medio de correo electrónico. A este diálogo se entra seleccionando las propiedades del dispositivo de red creado en WhatsUp®, icono Alerts, opción Add SMTPMail. En este diálogo se coloca la dirección de correo electrónico de la persona destino, la dirección del servidor de correo, y el asunto e información del mensaje del correo a enviar en caso de alerta.

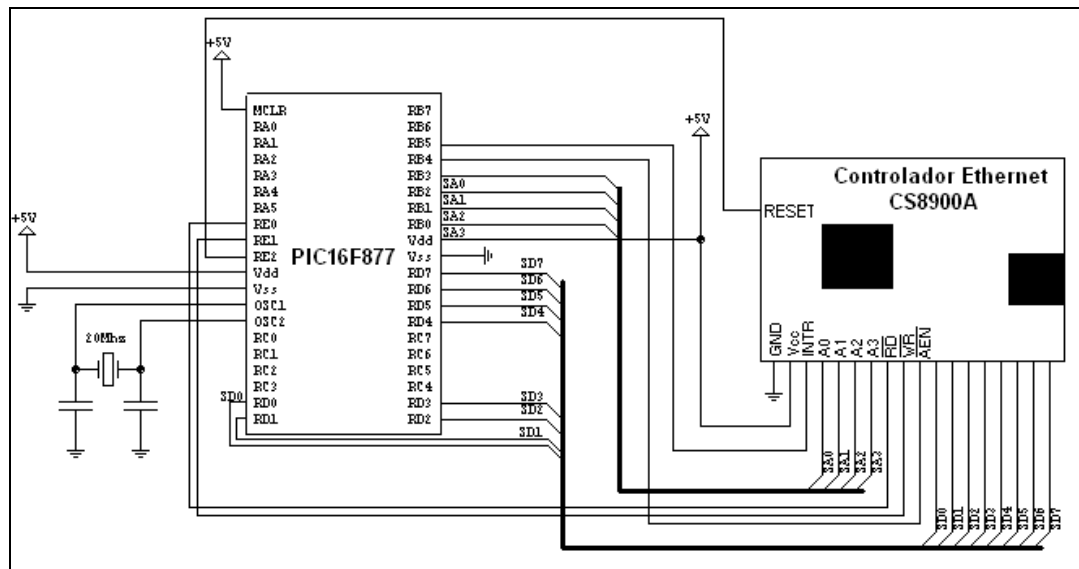


## CAPÍTULO 6

### ANÁLISIS Y RESULTADOS

#### 6.1 Explicación del diseño

##### 6.1.1 Hardware



**Figura 6.1** Diagrama lógico de conexión entre el microcontrolador PIC16F877 y el controlador ethernet CS8900A

Para implementar el sistema se eligió trabajar con un circuito conformado por un microcontrolador y un controlador ethernet, de manera que tomar variables externas se hiciera más sencillo. Además, el microcontrolador ofrece la posibilidad de manejar al controlador de red. Debe de recordarse que el paquete SNMP con el trap como mensaje debe de ser implementado, y el microcontrolador ofrece también esta posibilidad.

En la figura 6.1 se muestra la forma de conexión del microcontrolador PIC16F877 con el controlador ethernet Cirrus Logic CS8900A. Para el desarrollo de proyecto esta configuración fue adquirida como una sola tarjeta, y es la que se muestra en la figura 6.2. En ella se puede observar la disposición de los principales componentes, donde resaltan el microcontrolador PIC 16F877, el controlador de red ethernet Cirrus CS8900A y los puertos serial y RS-232.

También debe de señalarse que trae puertos disponibles del microcontrolador para conectar dispositivos externos, que en este caso se trataba de los contactos magnéticos y del sensor de temperatura.

La empresa fabricante de este hardware es EDTP<sup>2</sup>. Puede consultarse la figura 6.3, donde se muestra el diagrama de conexión de los componentes de la tarjeta.

A pesar de que el controlador ethernet consta de 100 patillas, en esta aplicación se hicieron uso solamente de 16 de ellas para su manejo por parte del microcontrolador.

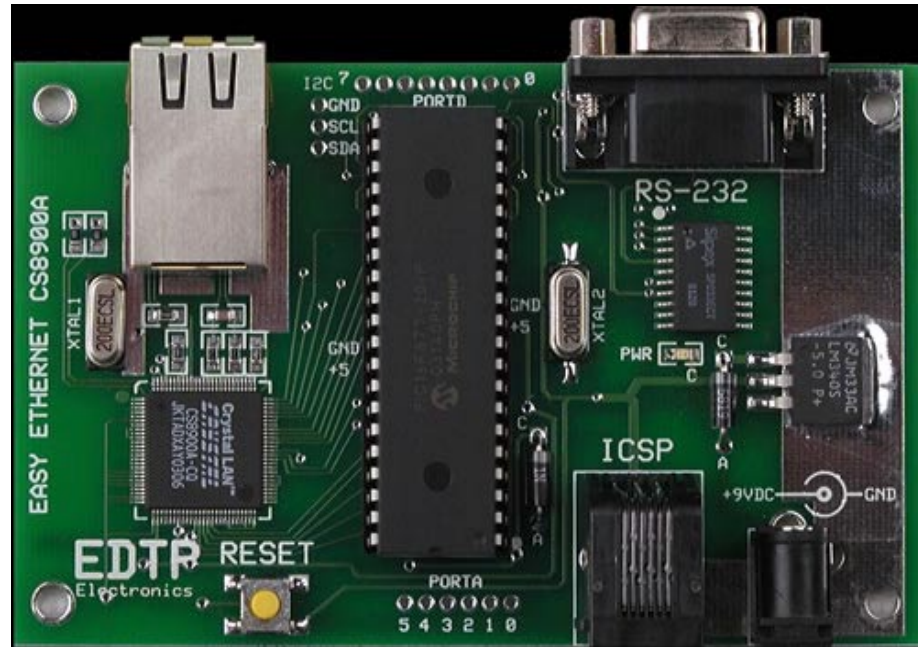


Figura 6.2 Módulo adquirido para el desarrollo del proyecto.

### 6.1.1.1 Descripción de controlador Ethernet CS8900A

El CS8900 tiene 4 kilobytes de memoria, permitiéndole recibir y enviar paquetes al microcontrolador. Para esta aplicación consta de un bus de direccionamiento de 4 bits, uno de 8 para datos y 4 señales de control. Más adelante se detallarán cada una de ellas.

Los registros internos de este controlador están agrupados en cinco grupos: registros de bus, estado y control, inicio de transmisión, recepción de tramas y envío de tramas. Estas cinco grandes categorías, además de otros de menor rango se encuentran en el área de memoria de 4 kilobytes ya mencionada, y se denomina PacketPage. Es importante mencionar que los registros de estado y control están

---

<sup>2</sup> Ver el sitio electrónico [www.EDTP.com](http://www.EDTP.com) para mayor información.

divididos en dos subgrupos: registros de configuración y control; y registros de eventos y estado. Los registros de configuración y control determinan cuántas tramas son transmitidas y recibidas. Además, determinan qué tramas serán enviadas y recibidas y qué eventos causarán una interrupción que será enviada al microcontrolador PIC16F877.

A continuación se menciona una breve descripción de las señales de entrada y salida utilizadas para el controlador Cirrus Logic CS8900A:

**SD0- SD7**, bus de datos bidireccional: con estas señales se permite enviar y recibir datos de 8 bits entre el microcontrolador PIC16F877 y el controlador ethernet.

**SA0- SA3**, bus de direcciones: son las entradas que proporcionan acceso al espacio de memoria del CS8900A, es decir, para lectura y escritura de un determinado sector del banco de memoria del controlador.

**$\overline{RD}$** , señal de entrada para lectura: al estar en estado bajo y con un valor de dirección detectado, el controlador ethernet coloca el byte específico en el bus de datos.

**$\overline{RW}$** , señal de entrada para escritura: al estar en estado bajo, el controlador ethernet lee el byte específico del bus de datos y lo almacena en el valor de dirección detectado.

**AEN**, señal de entrada para la habilitación de dirección: permanecerá inactiva al ejecutar accesos de memoria.

**INTR**, señal de salida para indicación de interrupción: al estar en estado alto indica la presencia de una interrupción. Se coloca en estado bajo al leerse solo bits ceros en el registro de cola de estado de interrupción (ISQ). Después de que la interrupción quede definida y activada, se habilita la transmisión y recepción de controlador.

“Diseño e implementación de un prototipo de sistema de alarma para armarios de equipos de redes incorporado sobre una red TCP/IP existente”

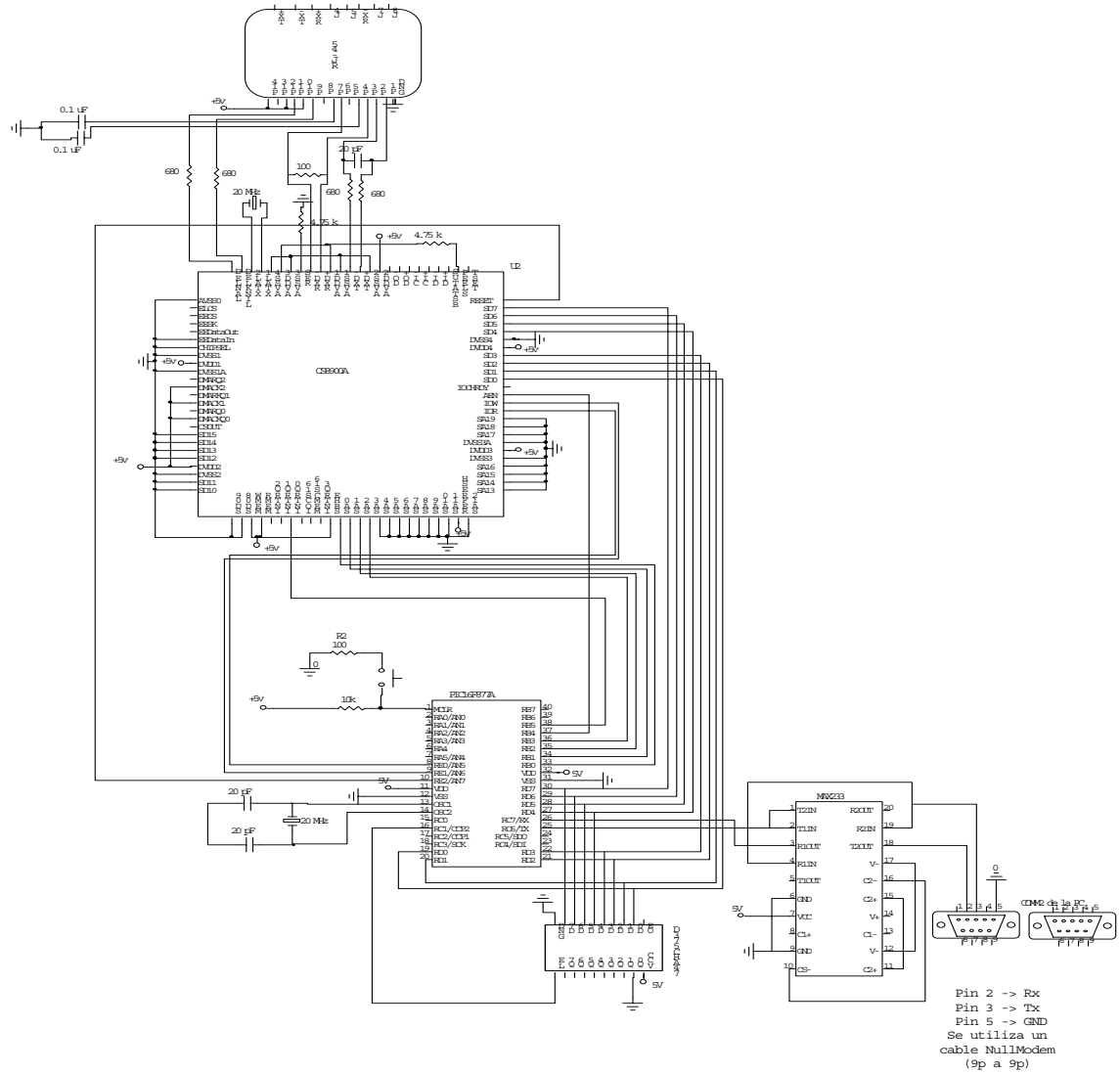


Figura 6.3 Diagrama de conexión de la tarjeta “Easy Ethernet CS8900A”.

Es importante mencionar que el motivo principal de utilizar solamente cuatro pines para el bus de direcciones es porque con ello es suficiente para lograr la decodificación necesaria para direccionar cada espacio de dicho mapa.

A continuación se muestra el mapa de memoria para operación en modo I/O, que es la utilizada para la transmisión y recepción de tramas.

**Tabla 6.1** Mapa de Memoria para el Controlador ethernet en modo I/O

<b>Desplazamiento</b>	<b>Tipo</b>	<b>Descripción</b>
0000h	Read/Write	Receive/Transmit Data (Port0)
0002h	Read/Write	Receive/Transmit Data (Port1)
0004h	Write-only	TxCMD (Transmit Command)
0006h	Write-only	TxLength (Transmit Length)
0008h	Read-only	Interrupt Status Queue
000Ah	Read/Write	PacketPage Pointer
000Ch	Read/Write	PacketPage Data (Port 0)
000Eh	Read/Write	PacketPage Data (Port 1)

El primer segmento, en la dirección 0000h, se utiliza como puerto para leer o escribir en el buffer de recepción o transmisión respectivamente. El puerto uno, segmento 0002h es utilizado para transferencias de 32 bits cuando se utiliza el controlador en modo de memoria.

El TxCMD corresponde al comando de transmisión, e indica la configuración e inicio de envío de datos al buffer de transmisión.

El puerto TxLength, que comienza en el espacio de memoria 0006h, indica el largo de la trama a transmitir y es un parámetro que el controlador ethernet utiliza para terminar de almacenar datos en el buffer de transmisión.

El PacketPage Pointer es el que contiene el puntero que sirve para acceder a cada uno de los registros internos del controlador CS8900A. Es de 16 bits, de ellos, los primeros 12 bits, proporcionan la dirección interna del registro que se desea acceder, los siguientes tres corresponden a un identificador único para cada registro y el último se mantiene en cero para poder acceder a los espacios de memoria adecuadamente.

Los PacketPage Data es donde se almacenan los datos a colocar en los registros internos del controlador, direccionados con el PacketPage Pointer. En este caso sólo se usa el puerto cero ya que se utiliza el controlador ethernet en modo I/O de ocho bits.

### 6.1.1.2 Señales y puertos utilizados del microcontrolador PIC16F877

El microcontrolador necesita 16 pines para poder manejar los buses de control, datos y direcciones del controlador Cirrus CS8900A. La tarjeta adquirida dispone de una conexión con éste requerimiento, y para tal necesidad, hace uso del microcontrolador PIC16F877. La ventaja de este microcontrolador es que puede ser fácilmente programado usando un pequeño número de componentes externos. En esta tarjeta se deja disponible el puerto A y D del microcontrolador para el manejo de señales. Cabe mencionar además que, por los requerimientos de fácil configuración de parámetros, como direcciones IP, se usaron los pines correspondientes a transmisión serie del microcontrolador. A continuación, se mencionan la distribución de los pines en cada puerto utilizados para la conexión con el CS8900A:

**RA0:** El pin RA0 es utilizado como entrada analógica para hacer una lectura de voltaje proporcional al valor de temperatura. Esta señal analógica es generada por el sensor TC1046 fabricado por Microchip®.

**RA1- RA5:** Los pines de RA1 a RA5 se usan como señales digitales generadas por contactos magnéticos colocados en la puerta y paredes del armario de equipos de redes.

**RB0- RB3:** los cuatro primeros bits del puerto B se usan como salidas para manejar las entradas correspondientes al bus de direcciones del controlador ethernet.

**RB4:** este pin del microcontrolador es usado como salida para servir de habilitador de dirección en el CS8900A. Por lo tanto, su conexión es directamente con la patilla que tiene la señal de AEN.

**RB5:** el sexto pin del puerto B se utiliza como entrada para que el controlador ethernet pueda informar al microcontrolador PIC sobre una petición de interrupción. Por lo tanto, está conectado a la terminal INTR.

**RC1:** Este pin es utilizado para habilitar el registro 74AHCT573.

**RC6 y RC7:** corresponden a las señales TXD y RXD del microcontrolador. Se utilizan para poder visualizar o cambiar la configuración de la tarjeta a través de una comunicación serie con una computadora.

**RD0- RD7:** todo el puerto D del microcontrolador se usa como puerto bidireccional para leer o escribir del bus de datos del controlador Cirrus Logic CS8900A. Por lo tanto, la transmisión y recepción está limitada a datos de un byte, en ésta aplicación. Además, a este puerto está conectado el registro 74AHCT573, el cual permite utilizar indirectamente este puerto como salidas digitales.

**RE0:** pin del puerto E utilizado como salida para habilitar lectura de datos en el controlador ethernet. Como se observa en la figura 6.1, está conectado a la patilla que maneja la señal IOR.

**RE1:** este pin, es una salida que se encarga de habilitar la escritura en el puerto del bus de datos, por lo tanto, está conectado a IOW del CS8900A.

### **6.1.1.3 Señales utilizadas del registro 74AHCT573**

La tarjeta posee una conexión del puerto D del microcontrolador con este registro. La finalidad es para poder tener la posibilidad de utilizar dicho puerto como salidas analógicas.

**LE:** entrada que permite habilitar el registro, además de mantener el dato recibido por D0 a D7 cuando se deshabilita.

**D0-D7:** estas entradas sirven como lectura del puerto D del microcontrolador cuando LE está habilitado.

**Q0-Q7:** puerto de salida para el dato leído de D0 a D7.

## **6.1.2 Software**

### **6.1.2.1 Descripción de la programación realizada para el microcontrolador**

#### **6.1.2.1.1 Manejo de variables y de espacio de memoria**

Para el manejo de peticiones ARP<sup>3</sup> y de cambios de configuración del módulo por UDP<sup>4</sup>, se utiliza un paquete de datos definido en el área de memoria RAM del microcontrolador, donde se asignan etiquetas equivalentes a los nombres de los espacios definidos en los datagramas de los protocolos utilizados. Al recibir una trama, los datos son guardados temporalmente en este paquete, donde por medio de programación se realizan las funciones necesarias para dar respuesta según el tipo de trama recibida.

Para el envío de traps, se utiliza espacio de la memoria flash del microcontrolador, donde se define una estructura única con un largo de mensaje constante. Cada variable o sensor del microcontrolador tiene asignado un mensaje particular, grabado en dicha memoria del microcontrolador, con lo que en el momento de transmitir se debe añadir a la trama el mensaje correspondiente a esa variable.

---

<sup>3</sup> ARP: protocolo de resolución de direcciones. Se utiliza para encontrar una dirección MAC a partir de una dirección IP conocida.

<sup>4</sup> UDP: protocolo de datagrama de usuario. Pertenece a la capa de transporte del modelo TCP/IP y presta sus servicios a diferentes capas de aplicación para su comunicación.



### 6.1.2.1.2 Descripción de las variables utilizadas

En la tabla 6.2 se muestra una descripción breve de las variables que se utilizaron en la programación del microcontrolador.

**Tabla 6.2** Variables utilizadas en la programación del microcontrolador

Variable	Tipo de dato	Tamaño (bytes)	Significado o utilización
Pageheader	Espacio de memoria	4	Valores generados por el controlador ethernet cuando el microcontrolador atiende una interrupción de recepción de trama
Packet	Espacio de memoria	96	Guarda tramas recibidas o que van a ser enviadas
ppoffsetH,ppoffsetL	Número	1	Para direccionamiento de registros del controlador ethernet
*addr	Número	1	Almacena una dirección de la RAM
isq_L, isq_H	Número	1	Tipo de interrupción generada en el controlador ethernet
data_H,data_L	Número	1	Utilizado como variables temporales o para la comunicación con el controlador ethernet
i	Número	2	Utilizado como un índice en las funciones
Txlen	Número	2	Almacena el valor del largo de la trama a transmitir
Rxlen	Número	2	Almacena el valor del largo de la trama recibida
Checksum16	Número	2	Almacena el valor de suma de chequeo
Hdrlen	Número	2	Largo de encabezado para cálculos en funciones
AddrROM	Número	2	Almacena una dirección de la ROM
hdr_checksum	Número	4	Acumula la suma de chequeo del protocolo IP y UDP
MYIP	Número	4	Almacena la dirección IP del módulo
MYMAC	Número	4	Almacena la dirección MAC del módulo
TxTotal;	Número	1	Largo total a transmitir por la red

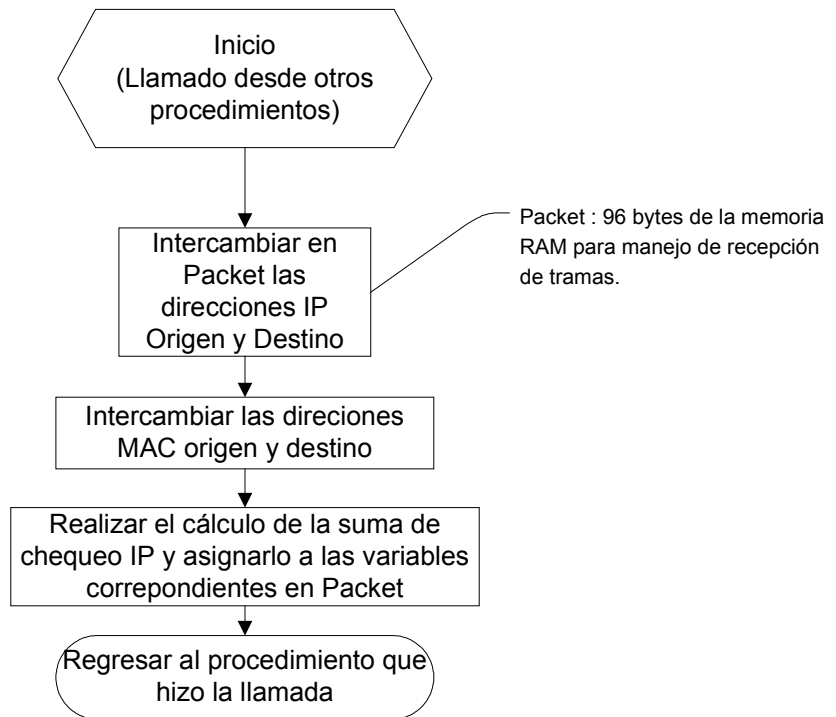
**Tabla 6.2** Continuación

Variable	Tipo de dato	Tamaño (bytes)	Significado o utilización
ContChar	Número	1	Contador de caracteres
ContDir	Número	1	Contador para direcciones de la ROM
DatoPuerto	Carácter	66	Valor de entrada del puerto serie o por red a través de UDP
cmdPuerto	Carácter	1	Almacena temporalmente el carácter que llega por puerto serie
ContPuerto	Número	1	Cuenta el número de caracteres que llega al puerto serie
Senal	Número	1	Almacena el valor de un pin del microcontrolador
ValorPortA	Número	1	Almacena el valor de puerto A del microcontrolador
PinRev1A0,PinRev2A0 PinRev1A1,PinRev2A1 PinRev1A2,PinRev2A2 PinRev1A3,PinRev2A3 PinRev1A4,PinRev2A4 PinRev1A5,PinRev2A5	Bit	1/8	Utilizado para enviar traps solo una vez al activar algún contacto magnético
ValorADC	Número	16	Valor que se lee del convertidor de analógico a digital
Tempe	Número	16	Almacena temporalmente los valores del rango de temperatura
EstadoAlarma	Número	8	Variable para desactivar la alarma
BanderaUDPConfig	Número	8	Se analiza para saber si la configuración se envía por puerto serie o por red
UDP_Source_Low, UDP_Source_High	Número	8	Valor del puerto origen para uso del protocolo UDP
UDP_Dest_Low, UDP_Dest_High	Número	8	Valor de puerto destino para uso del protocolo UDP
Temp_Lim_Inf, Temp_Lim_Sup	Número	8	Guarda el rango permisible de temperatura del armario
Comunidad	Carácter	6	Valor de comunidad del Trap de alerta

### 6.1.2.1.3 Funciones y procedimientos importantes

#### setipaddrs()

Hace un intercambio de las direcciones MAC e IP. En este procedimiento también se realiza un cálculo de la suma de verificación de IP para la respuesta de petición ARP de algún dispositivo de la red. En la figura 6.4 se muestra el diagrama de flujo para este procedimiento.



**Figura 6.4** Diagrama de flujo del procedimiento setipaddrs().

#### cksum()

Realiza la suma en palabras de 16 bits, de un cierto número de bytes, a partir de un lugar de memoria de la RAM especificado.

#### cksumROM()

Realiza la suma en palabras de 16 bits, de un cierto número de bytes, a partir de un lugar de memoria flash especificado.

El diagrama de flujo para estos procedimientos de cálculo de suma de chequeo es mostrada en la figura 6.5.

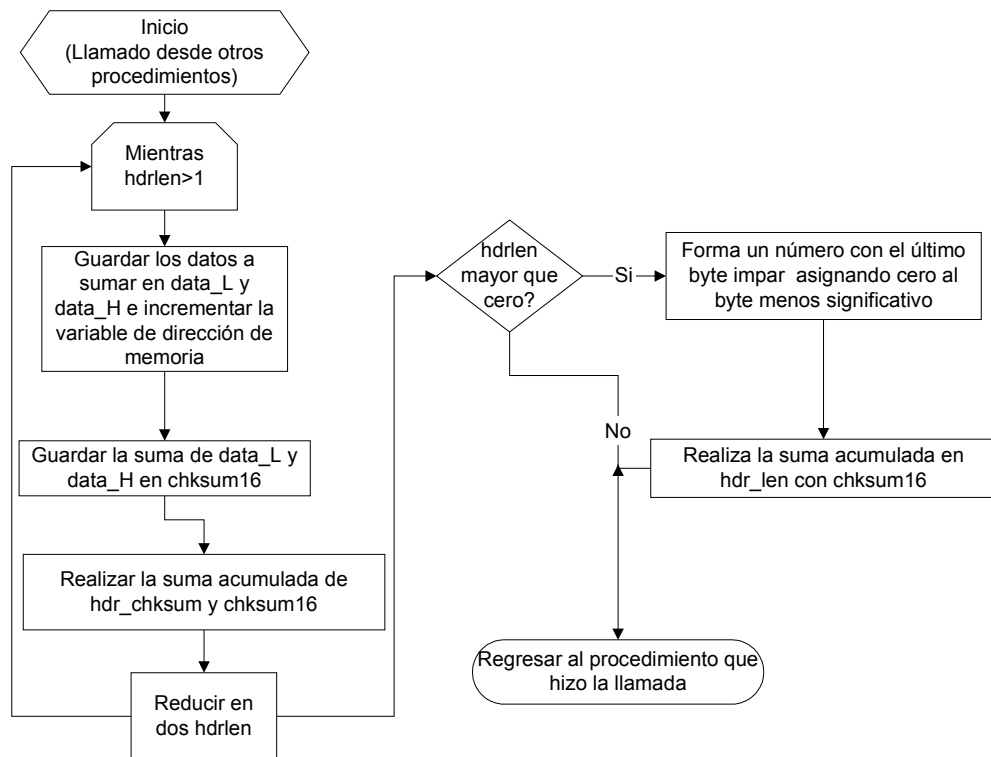


Figura 6.5 Diagrama de flujo para los procedimientos cksun() y cksunROM

RpppL()  
RpppH()

Realizan una lectura de un byte en una dirección de memoria dada para el CS8900A. Debido a que el controlador ethernet maneja palabras de 16 bits, con RppL se lee el byte bajo, mientras que con RppH se lee el byte alto de esa palabra.

En la figura 6.6 se muestra el diagrama de flujo del procedimiento RppL.

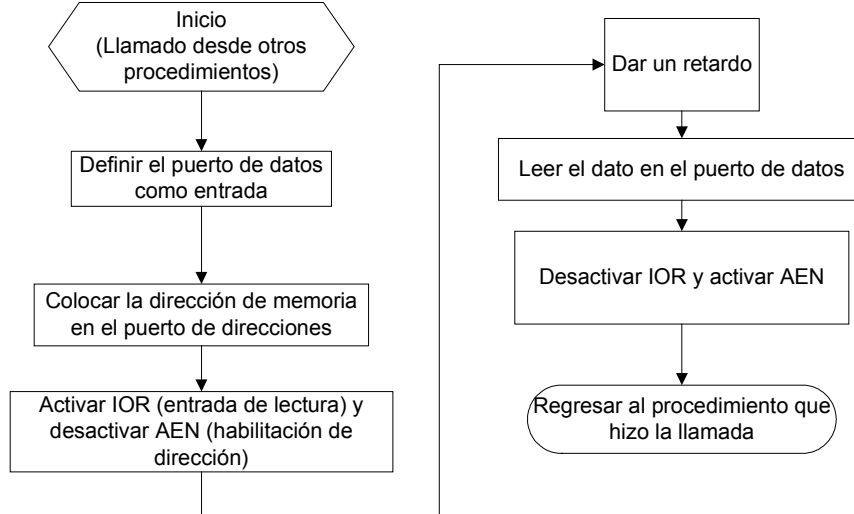
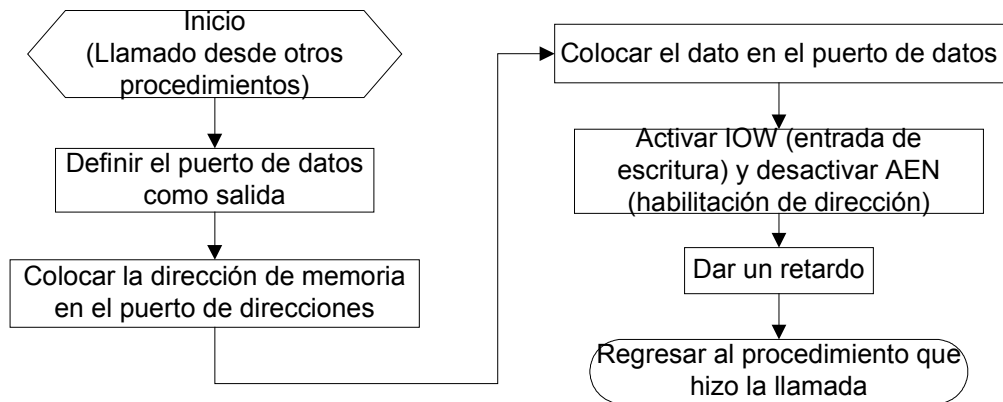


Figura 6.6 Diagrama de Flujo para el procedimiento RpppL.

WpppL()

WpppH()

Realizan una escritura de un byte en una dirección de memoria dada para el CS8900A. Debido a que el controlador ethernet maneja palabras de 16 bits, con RppL se lee el byte bajo mientras que con RppH se lee el byte alto de esa palabra. En la figura 6.7 se muestra el diagrama de flujo para estos procedimientos.



**Figura 6.7** Diagrama de Flujo para los procedimientos WpppL y WpppH.

PPRead()

PPWrite()

Utilizan los procedimientos WpppL, WpppH, RpppL y RpppH para hacer lecturas o escrituras completas de una palabra de una localidad de memoria del CS8900A.

RPP (int16 ppoffset)

WPP (int16 ppoffset, int16 datum)

Separan una dirección de memoria de 16 bits del CS8900A recibida como palabra y realiza la correspondiente llamada para la lectura o escritura en el controlador ethernet, utilizando para ello los procedimientos PPRead() y PPWrite().

ARP()

Asigna los valores del datagrama ARP al paquete ubicado en la RAM del microcontrolador.

echo\_packet()

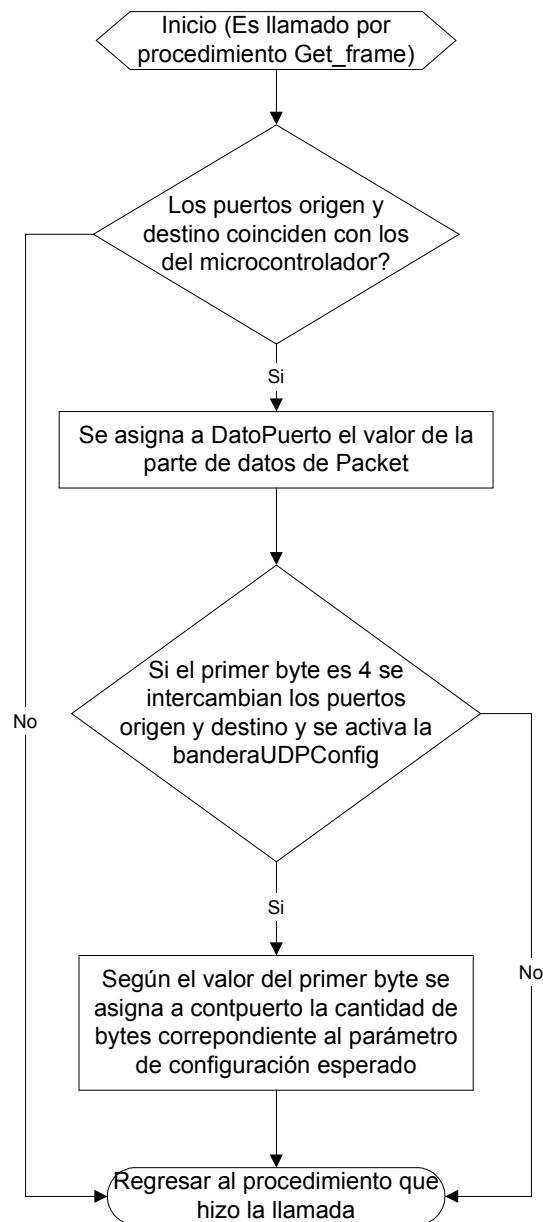
Da respuesta de eco ante un ping solicitado por algún dispositivo de la red, transmitiendo una trama por el medio.

icmp()

Da respuesta de eco ante un ping solicitado por algún dispositivo de la red añadiendo los datos necesarios a la trama a ser transmitida por echo\_packet().

UDP()

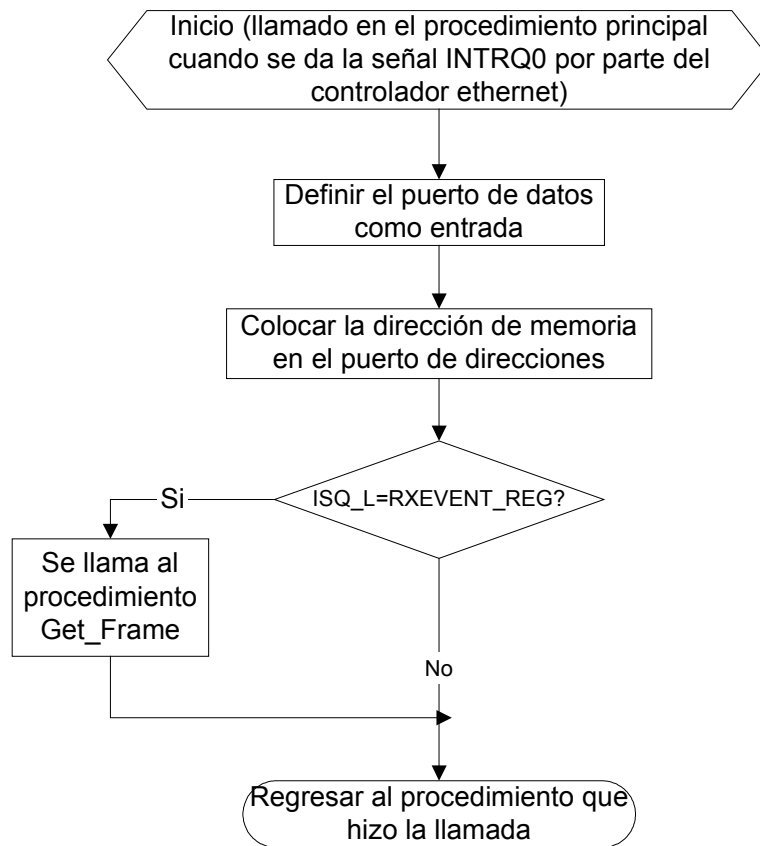
Analiza los diversos puertos del paquete UDP que se ha recibido para el módulo ejecutando las tareas correspondientes. En la figura 6.8 se muestra el diagrama de flujo del procedimiento.



**Figura 6.8** Diagrama de Flujo para el procedimiento UDP.

### ServiceISQ()

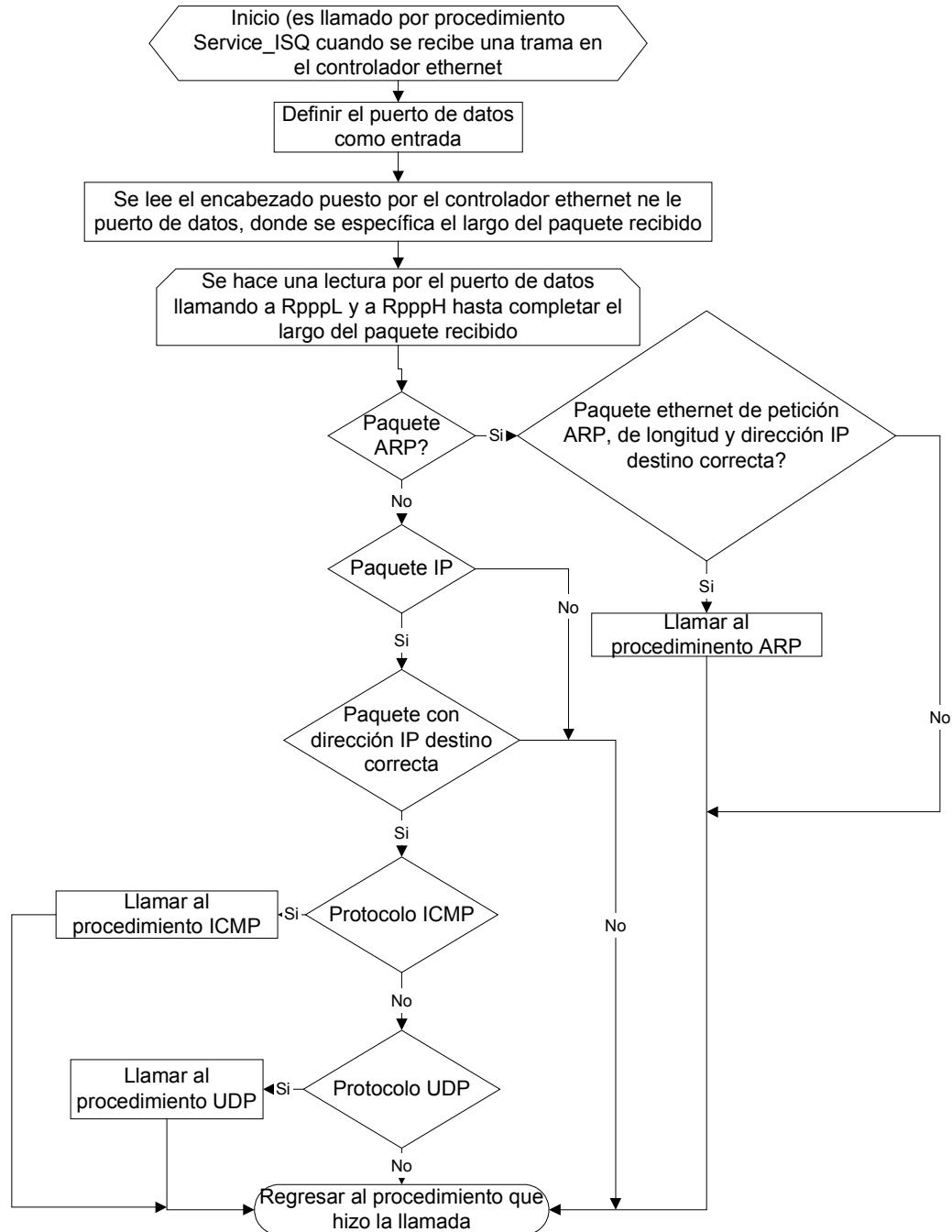
Analiza cuando se da una interrupción por llegada de datos al buffer de recepción del controlador ethernet. La patilla del CS8900A que avisa de dicha interrupción es INTRQ0. En la figura 6.9 se muestra el diagrama de flujo para el manejo de interrupciones.



**Figura 6.9** Manejo de interrupción para recepción de tramas por medio de Service\_ISQ.

### Get\_Frame()

Analiza el tipo de trama que ha llegado al buffer de recepción del controlador ethernet, llamando así a los procedimientos que analicen dicha trama. En la figura 6.10 se muestra el diagrama de flujo para get\_frame().

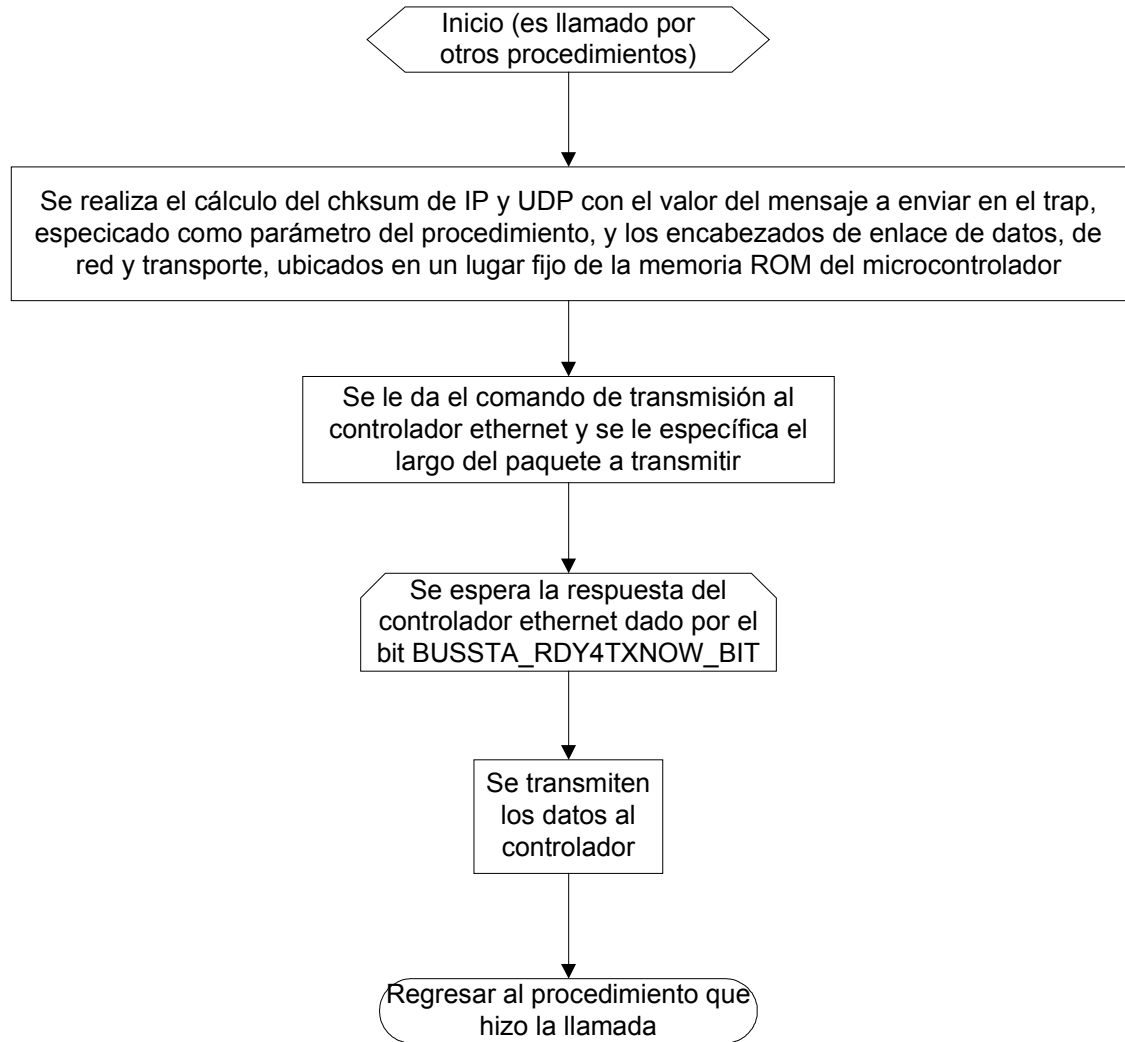


**Figura 6.10** Diagrama de flujo para el procedimiento Get\_Frame.



SNMP\_TX(int16 puntMensaje)

Permite enviar a través de la red ethernet un TRAP SNMP, el cual contiene un mensaje ubicado en la dirección de la ROM recibida como parámetro. El diagrama de flujo es mostrado en la figura 6.11.



**Figura 6.11** Diagrama de Flujo para el procedimiento SNMP\_TX.

### Sensado()

Analiza las variables o sensores del microcontrolador. Al activarse alguna, transmite el trap con el mensaje correspondiente por medio de SNMP\_TX. En la figura 6.12 se muestra el diagrama correspondiente.

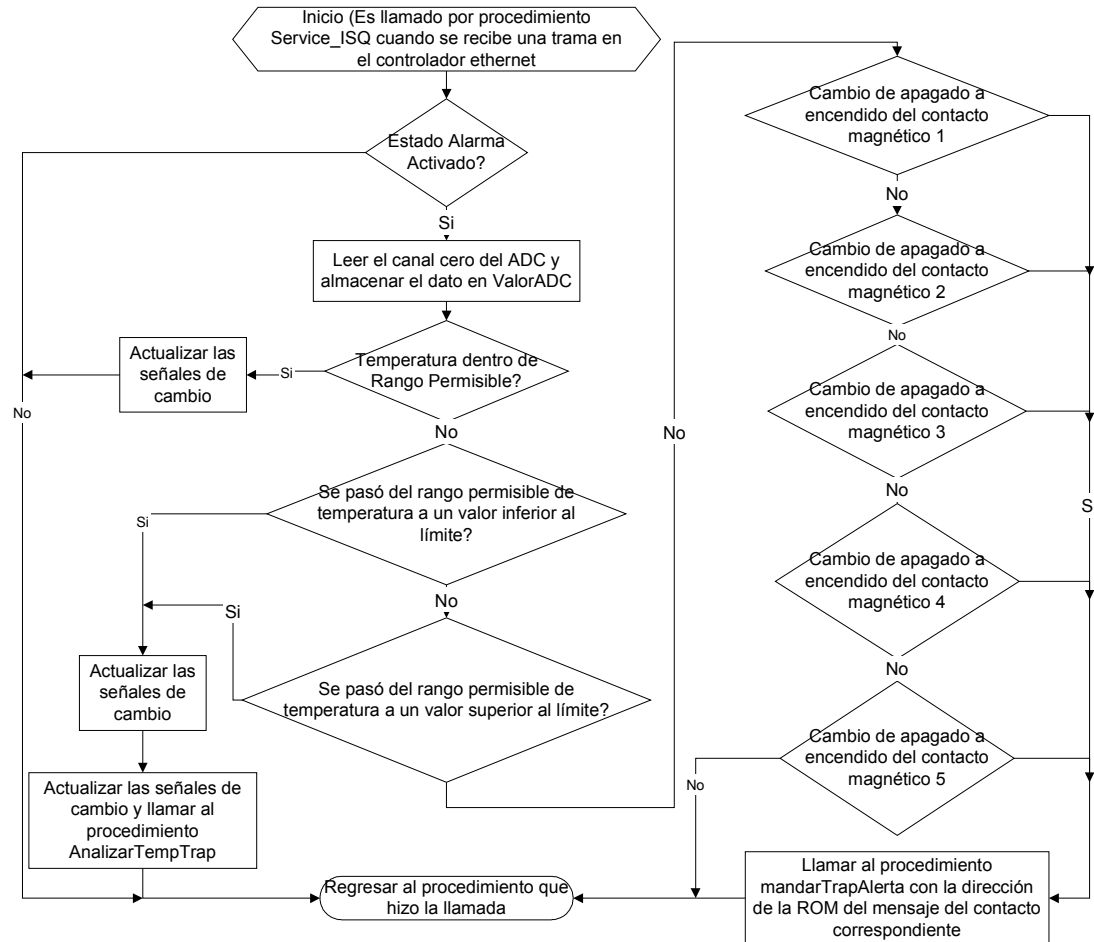


Figura 6.12 Diagrama de flujo para el procedimiento Sensado.

### EncenderBuzzer()

Permite hacer un parpadeo en el LED de alarma como aviso de alerta además de hacer sonar el buzzer durante dos segundos.

### MandarTrapAlerta(int16 puntMens)

Coloca dos traps en la red con el mensaje en el que apunta puntMens.

### TempTrap()

Escribe el valor de temperatura en la memoria flash del microcontrolador, específicamente en el mensaje de la primera variable, para luego mandar el trap correspondiente.

### AnalizarTempTrap()

Permite enviar un trap de alerta en caso de que el rango de temperatura excediese sus límites.

### Sensado()

Analiza cuando hay un cambio en un contacto magnético para mandar el trap con el mensaje correspondiente.

### EnviarConfig()

Hace lectura de la memoria flash del microcontrolador para enviar por el puerto serie los datos de direcciones IP origen y destino, además de los mensajes de las variables que se pueden dar.

### EnviarConfigRed()

Este procedimiento permite enviar la configuración del módulo (direcciones MAC e IP origen y destino, rango de temperatura, comunidad y Puertos UDP ) a través de la red. Estos valores son guardados como parte de un paquete UDP que es interpretado por un computador remoto que ejecuta la aplicación realizada en Visual Basic®.

### CambiarMACDestino()

Realiza una escritura en la ROM del microcontrolador para cambiar la dirección MAC del primer equipo al que se le debe enviar el trap de alerta SNMP.

### CambiarEstadoAlarma()

Cambia la variable que determina el estado de activado o desactivado de la alarma del sistema.

### CambiarRangoTemp()

Realiza una escritura en la ROM del microcontrolador para cambiar el rango de temperatura permisible dentro del armario de equipos de redes.

### CambiarMACOrigen()

Realiza una escritura en la ROM del microcontrolador para cambiar la dirección MAC del módulo.

### Cambiar UDP Puertos()

Realiza una escritura en la ROM del microcontrolador para cambiar los valores de los puertos UDP por los que se recibe comandos en forma remota.

### CambiarComunidad()

Realiza una escritura en la ROM del microcontrolador para cambiar la comunidad que se utiliza en el mensaje SNMP

### PuertoSerie()

Analiza los comandos recibidos por el puerto serie para tomar las acciones en cada caso. En la figura 6.13 se muestra el diagrama de flujo para este procedimiento.

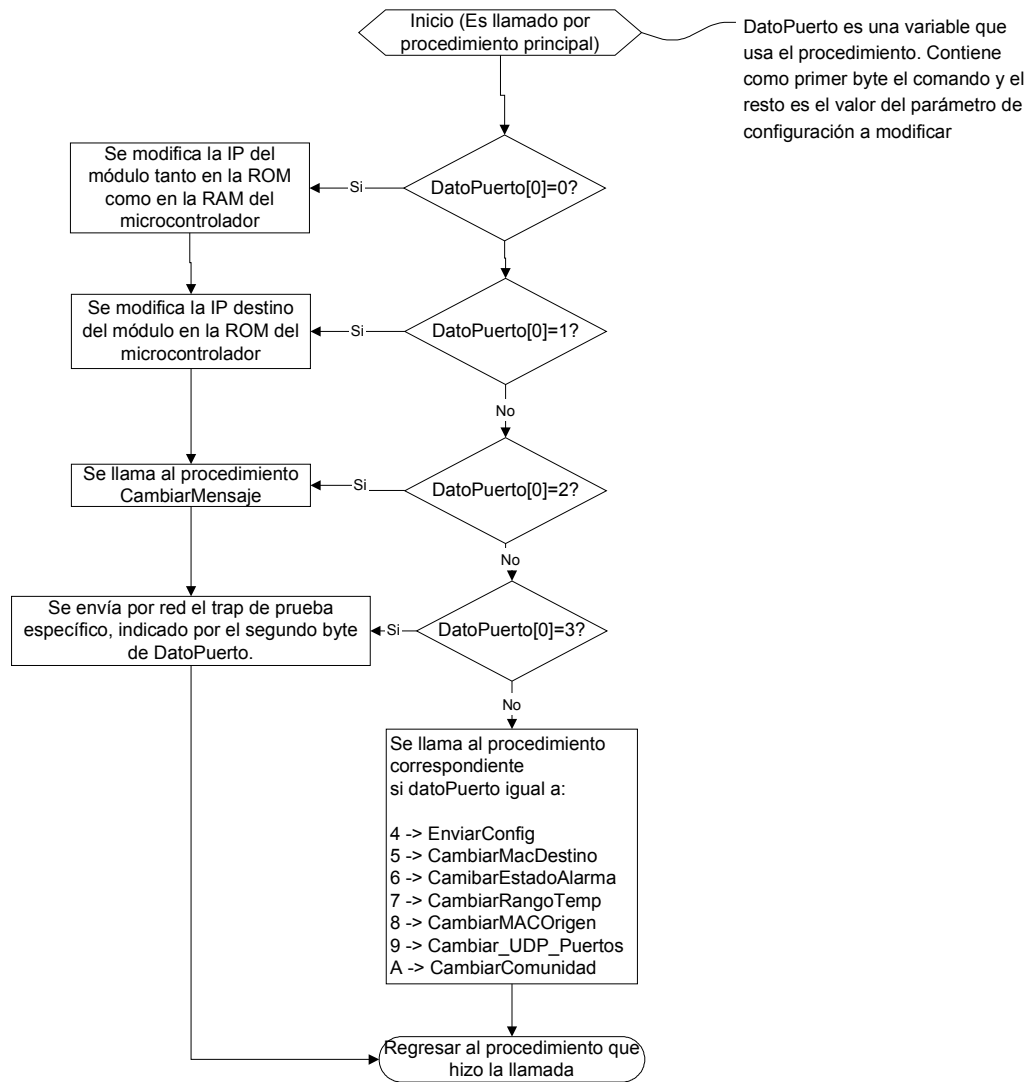


Figura 6.13 Diagrama de Flujo para el procedimiento Puerto Serie

#### ActualizarPuertos temperatura()

Este procedimiento realiza una lectura de la memoria ROM del microcontrolador al reiniciarse el sistema, con el objetivo de actualizar las variables de la RAM del rango de temperatura, comunidad SNMP del trap y puertos UDP.

En la figura 6.14 se muestra un esquema que describe las partes que constituyen la trama de alerta implementada, el cual es un paquete trap del protocolo SNMP.

En los diferentes campos del protocolo SNMP es necesario especificar el tipo de dato y el largo de este antes de colocar el valor. Debido a ello se agregan dos bytes de más a lo requerido para el valor del dato a especificar dentro de la trama ensamblada. Es de mencionar que el protocolo SNMP es un protocolo de la capa de aplicación del modelo TCP/IP<sup>5</sup>, por lo cual se ubica al final de los valores de los encabezado IP y UDP.

El tiempo de respuesta promedio al comando PING por parte del módulo implementado es de 750 milisegundos. En ciertos casos, como cuando el microcontrolador está realizando rutinas de encendido de la alarma, puede ocurrir que el sistema de tardía la respuesta a una petición externa dada por un computador remoto.

Para el cálculo de la suma de verificación de los paquetes se aplicó la siguiente fórmula:

$$\text{chksum16} = \sim(\text{hdr\_chksum} + ((\text{hdr\_chksum} \& 0\text{xFFFF}0000) \gg 16)) \quad (\text{ec. 6.1})$$

donde se realiza el complemento a uno de la suma de palabras de 16 bits que conforman todos los campos dentro de la trama involucrados en la suma, almacenado en `hdr_chksum` y los acarrees generados por dicha suma. Esto es almacenado en `chksum16` para luego ser pasado a los campos de suma de verificación de las capas de red y transporte según sea el caso. En la capa de red sólo se utiliza el encabezado de esa capa, mientras que en la capa de transporte se realiza un pseudoencabezado con los valores de las direcciones de red, el encabezado UDP y el valor de los datos de la capa UDP.

Otros campos de la trama son rellenados con valores por defecto, con lo cual son constantes en todo momento.

---

<sup>5</sup> Ver glosario para la definición del modelo TCP/IP.

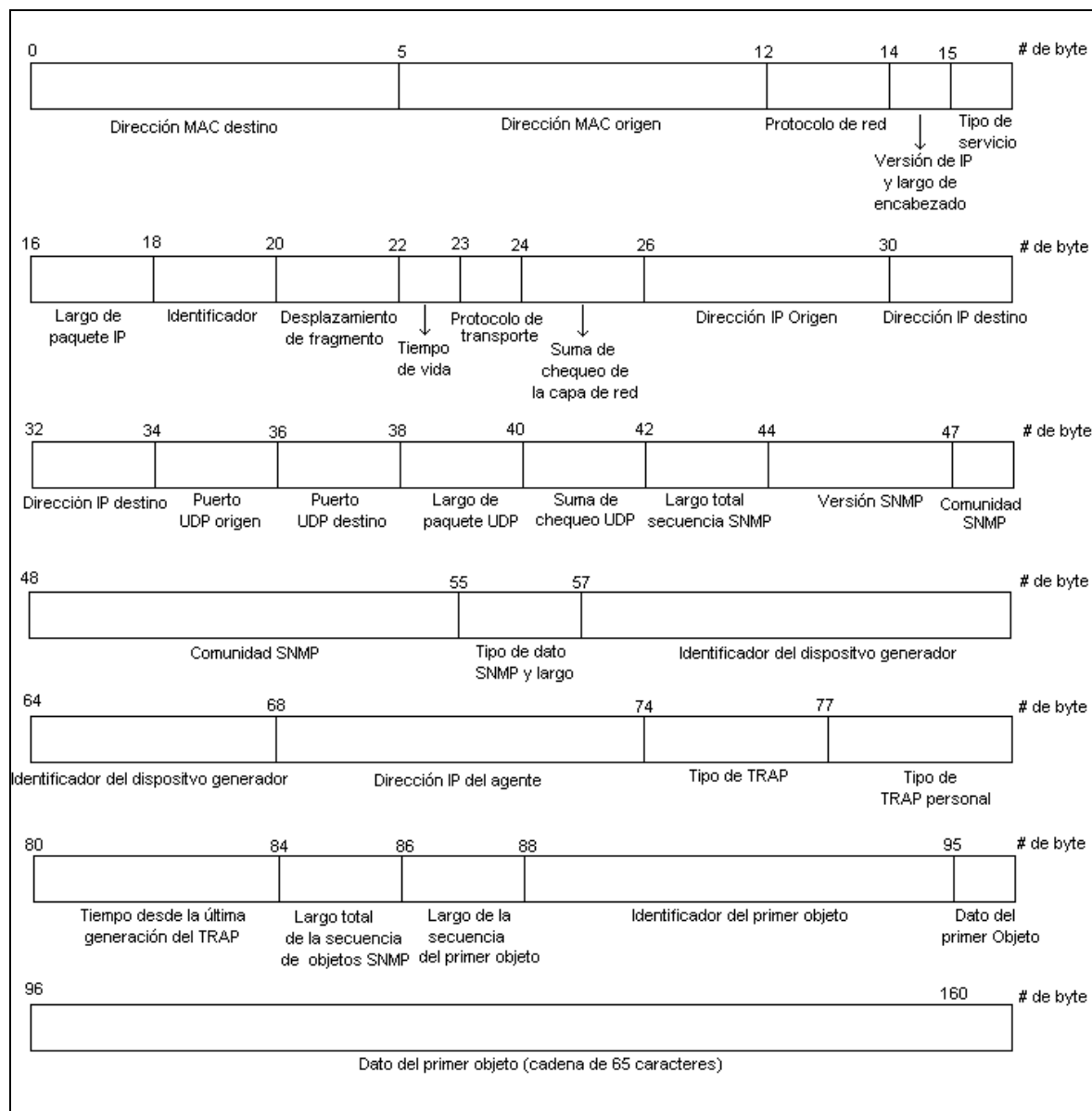


Figura 6.14 Esquema de la trama realizada usando una unidad de datos TRAP del protocolo SNMP

### **6.1.2.2 Descripción de la programación realizada en Visual Basic 6.0®**

A través de una interfaz amigable con el usuario, figura 6.15, se permite enviar comandos al microcontrolador.

Las funciones que se realizan por medio de dicha interfaz son las siguientes:

- Modificación de su configuración por puerto serie o por red.
- Variar la dirección IP del módulo.
- Cambiar la dirección IP donde se deben enviar los traps en caso de activación de alguna variable.
- Cambiar el mensaje que se envía al activar una variable.
- Envío por la red conectada de un trap con el mensaje específico de alguna variable.
- Envío de información sobre su configuración actual al computador por el puerto serie.
- Modificar los puertos UDP por los que escucha y envía el módulo.
- Seleccionar el rango de temperatura permisible dentro del armario de equipos de redes.
- Activación y desactivación de la alarma del módulo.
- Cambio de la comunidad por la que se envía los traps.

La programación realizada en este lenguaje de programación se facilita mucho por el uso de objetos, lo cuales tienen especificados sus procedimientos, propiedades y eventos. Lo más importante a destacar son los controles para el manejo de puerto serie y la tarjeta de red, que se explican en las páginas siguientes.

“Diseño e implementación de un prototipo de sistema de alarma para armarios de equipos de redes incorporado sobre una red TCP/IP existente”

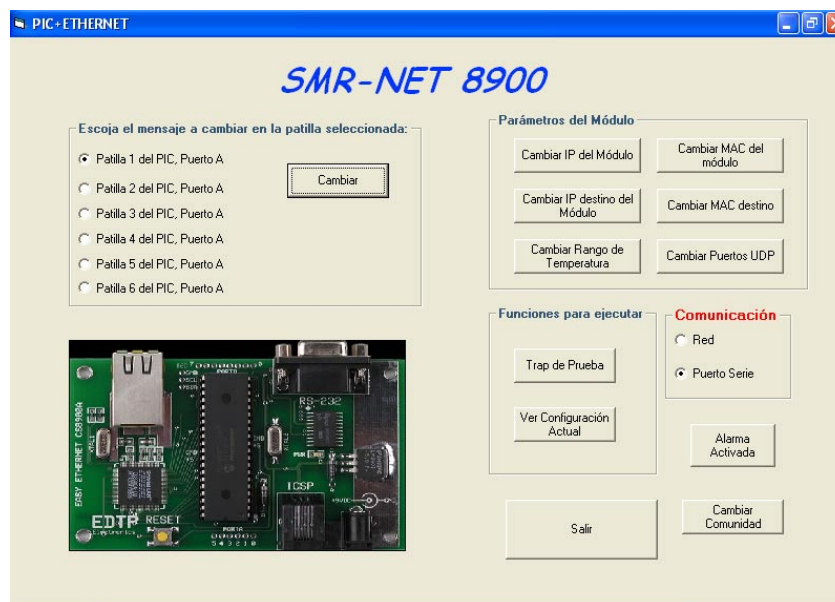


Figura 6.15 Interfaz realizada en Visual Basic 6.0®

### El control MSComm

Para el manejo del puerto serie de la computadora se usó el control MSComm. Sus propiedades fueron definidas como se muestra en la figura 6.16.

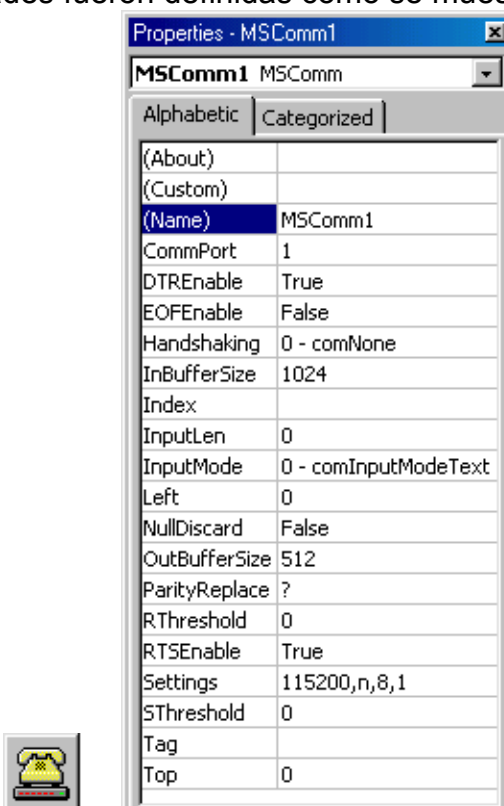


Figura 6.16 Cuadro de propiedades del control MSComm de Visual Basic®.



“Diseño e implementación de un prototipo de sistema de alarma para armarios de equipos de redes incorporado sobre una red TCP/IP existente”

---

### Procedimientos para el control MSComm

“MSComm1.Output = ValorPuerto”: Permite enviar un cadena de caracteres por el puerto serie del computador.

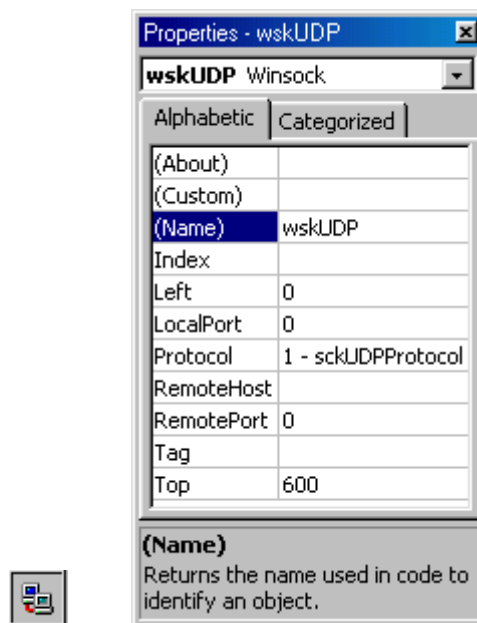
“MSComm1.PortOpen = True”: Permite habilitar el Puerto serie del computador

“MSComm1.PortOpen = False”: Permite deshabilitar el Puerto serie del computador.

La llegada de datos por el puerto serie se maneja por medio de la propiedad “MSComm1.InBufferCount”, la cual permite saber cuantos caracteres han sido recibidos. A través de un cronómetro se espera un tiempo prudencial para que el módulo pueda transmitir el largo de mensaje deseado.

El control Winsock de visual basic® fue el usado para poder transmitir datos a través de la tarjeta de red del computador. Sus principales características fueron definidas en el diálogo de propiedades como se muestra en la figura 6.17.

### El control Winsock



**Figura 6.17** Diálogo de propiedades del control Winsock de Visual Basic®.

### Procedimientos principales del control winsock de visual basic®

“Main.wskUDP.SendData ValorPuerto”: Permite enviar un datos por UDP a través de la tarjeta de red conectada. El dato ValorPuerto, es una cadena de caracteres que lleva la información deseada.

“Main.wskUDP.Close”: Permite deshabilitar el número de puerto usado para escuchar tramas UDP por el control.

“Main.wskUDP.RemoteHost = txtIPHostRemoto”: Permite especificar la dirección IP a la que se envían los datos.

“Main.wskUDP.RemotePort = Main.PuertoDestino”: Permite especificar el número de puerto usado para enviar los datos UDP. Este procedimiento reserva este puerto para ser usado únicamente por la aplicación en uso.

“Main.wskUDP.Bind Main.PuertoOrigen”: Determina el puerto de escucha de mensajes UDP por parte del control winsock. Este puerto es el que se necesita para transportar datos por medio del protocolo UDP.

### Eventos del control winsock

“wskudp\_DataArrival \_”: Utilizado para obtener la configuración del módulo en forma remota. Cuando llega una trama al computador el programa realizado llama a una subrutina que se encarga de procesar los datos de dicha trama.

Las peticiones de configuraciones son realizadas por parte del usuario a través de formularios con una interfaz gráfica amigable para el usuario. En el manual de usuario, en el apéndice A.3, se muestra el uso de los mismos. En general, al escribir en una caja texto el parámetro deseado, este se envía por puerto serie o por red, según la pestaña de comunicación activa. A ese dato se le agrega como encabezado un carácter que es interpretado por PIC como un comando para realizar las tareas correspondientes en cada caso.

## **6.2 Alcances y limitaciones**

El dispositivo es capaz de colocar traps con mensajes específicos en la red conectada, esto cuando uno de los contactos magnéticos, sensores de alarma en este caso, es activado al abrir una puerta o pared del armario del equipo de redes. Además cuenta con diversos parámetros de configuración que pueden ser modificados por puerto serie o en forma remota, a través de la red conectada. Los parámetros más importantes desde el punto de vista de seguridad son los puertos UDP, los cuales son analizados por el módulo para aceptar o rechazar alguna trama. Al sistema se le puede cambiar la dirección del computador remoto al cual se debe enviar los traps de alerta, el mensaje correspondiente a cada sensor y el rango permisible de temperatura dentro del armario de equipos de redes. Otras funciones importantes es la posibilidad de obtener y observar la configuración del módulo en cualquier momento y la opción de desactivar la alarma, esto en caso de que algún encargado deba ir al lugar a realizar reparaciones de los equipos.

## **CAPÍTULO 7**

### **CONCLUSIONES Y RECOMENDACIONES**

---

#### **7.1 Conclusiones**

1. El controlador ethernet CS8900A permite una fácil adaptación para ser manejado por microcontroladores con una pequeña cantidad de conexiones.
2. El protocolo de administración de red simple (SNMP) permite crear por medio de mensajes trap un sencillo aviso de alarma que es interpretado por dispositivos de red remotos adecuados a dicho protocolo.
3. Con el uso del protocolo UDP se puede realizar una fácil comunicación de datos a través de una red.
4. Por medio de sistemas de este tipo se pueden adecuar las diversas capas del modelo TCP/IP a la aplicación requerida.
5. Por medio del sistema implementado es posible obtener un monitoreo de variables externas a los dispositivos de red.

## 7.2 Recomendaciones

El sistema diseñado hace uso de un microcontrolador PIC, el cual cuenta con diversas cualidades, como por ejemplo una comunicación serial sincrónica y asincrónica, temporizadores, comparadores, entradas analógicas, patillas de entrada y salida, etc, con lo cual es posible hacer interfaz con otros dispositivos más especializados para así poder crear infinidad de aplicaciones. Además, debido a que es un dispositivo reprogramable, se pueden implementar otros tipos de protocolos de la pila TCP/IP. Entre las recomendaciones de aplicaciones de red se puede mencionar:

- monitoreo de equipos, para hacer lecturas de parámetros importantes como son temperatura, humedad, corriente, voltaje, presión, etc.
- Administración remoto de diversos dispositivos como el encendido o apagado, cambio de parámetros, apertura de cerraduras electrónicas, etc.

“Diseño e implementación de un prototipo de sistema de alarma para armarios de equipos de redes incorporado sobre una red TCP/IP existente”

---

## **BIBLIOGRAFÍA**

---

Miller M., Managing Internetwork with SNMP, M&T Books, USA, 1993.

Gary Desrosiers, post@embeddedethernet.com, Interface almost any circuit to a LAN, february 2001, USA. Dirección electrónica: <http://www.embeddedethernet.com/>. Consulta en agosto 2003.

Craig Peacock, Craig.Peacock@beyondlogic.org. IP & Ethernet Interfaces ,29th September 2003. Dirección electrónica: <http://www.beyondlogic.org/etherip/ip.htm>. Consulta en agosto 2003.

ETHERNET PIC CONTROLLER (NETPIC) Manual, 2001 EIX Ltd. Dirección electrónica: <http://www.eix.co.uk/Ethernet/NetpicManual.htm>. Consulta en agosto 2003.

Support@edtp.com, EDTP Electronics. Merritt Island Florida, USA 2003 . Dirección electrónica: <http://ww.edtp.com>. Consulta en agosto 2003.

## APÉNDICES

---

### Apéndice A.1 Glosario

- **Agente:** parte de un sistema de gestión de red que reside en las estaciones de trabajo u otros dispositivos de la red (llamados elementos gestores) y que recopila datos para informar sobre el estado de esos dispositivos al sistema de gestión.
- **Datagrama:** unidad fundamental de una transmisión de datos efectuada mediante el protocolo de internet.
- **Dirección MAC:** valor de 48 bits que permite dar un identificador único a cada dispositivo conectado en una red.
- **Dirección IP:** valor de 32 bits que permite dar un identificador lógico a un dispositivo dentro de una red.
- **Enrutador:** dispositivo de interconexión de redes que da servicio a paquetes o tramas que contienen determinados protocolos. Llevan a los paquetes o tramas a los enlaces adecuados para que lleguen a su destino.
- **Ethernet:** estándar de configuración de redes locales que permite la conexión de los computadores a una velocidad de transmisión de 10 megabits por segundo.
- **Modelo de red TCP/IP:** Sistema de referencia que divide en cuatro capas el funcionamiento de las redes de computadoras. Las capas son la de aplicación, transporte, internet y acceso a la red.
- **Ping:** programa de prueba asociado con el protocolo TCP/IP y utilizado para probar el canal de comunicaciones entre estaciones en internet o redes locales.
- **Sensor:** dispositivo electrónico capaz de tomar y analizar eventos externos del entorno que le rodea.
- **Sniffer:** programa que intercepta la información que transita por una red computacional y que permite analizarla y visualizarla.

- **Trap:** mensaje enviado por un agente SNMP a una consola, o terminal, para indicar la ocurrencia de un evento significativo, como una condición definida específicamente, o un umbral que ha sido alcanzado.
- **WhatsUP®:** programa que permite tener un control total sobre una red. Permite mapear dispositivos de red y monitorear eventos para dar mensajes de alarma mediante correo electrónico y radiolocalizador.



## **Apéndice A.2 Abreviaturas**

- **ARP:** Address Resolution Protocol, protocolo de resolución de direcciones. Este protocolo de internet permite encontrar una dirección MAC (dirección única de cada dispositivo de red) a partir de una dirección IP conocida.
- **IP:** Internet Protocol, protocolo de internet. Protocolo de capa de red de la pila TCP/IP que ofrece un servicio de internet sin conexión.
- **SNMP:** Simple Network Management Protocol, protocolo simple de administración de redes. El SNMP brinda una forma de monitorear, recolectar datos para estadísticas, análisis del desempeño y seguridad de una red, etc.
- **UDP:** User Datagram Protocol, protocolo de datagrama de usuario. Protocolo sin conexión de capa de transporte la pila de protocolo TCP/IP. UDP es un protocolo simple que intercambia datagramas sin confirmación o garantía de entrega y que requiere que el procesamiento de errores y las retransmisiones sean manejados por otros protocolos.
- **ICMP:** Internet Control Message Protocol, protocolo de mensajes de control en internet. Protocolo internet de capa de red que informa errores y brinda información relativa al procesamiento de paquetes IP.

### **Apéndice A.3 Manual de usuario del SMR- NET 8900**

A continuación se explican los pasos necesarios para poner en funcionamiento al sistema de alarma para armario de equipo de redes SMR-NET 8900.

#### **A.3.1 Especificaciones del sistema.**

- Entradas para contactos magnéticos de puertas: 5, normalmente cerrados.
- Entradas para el sensor de temperatura: 1, analógica.
- Puertos disponibles: 1 ethernet, RS-232. 1 serial.
- Alimentación: 9 voltios.

#### **A.3.2 Instalación.**

La instalación del módulo se debe realizar dentro del armario del equipo de redes, para evitar que sea desconectado por alguien no autorizado.

1. Coloque los cinco contactos magnéticos en las 2 paredes, las 2 puertas y el techo del armario. Conéctelos a las entradas del módulo dispuestas para tal fin, en el panel trasero.
2. Coloque el sensor de temperatura adentro, en un lugar cercano al centro del armario. Conéctelo a la entrada del módulo, en el panel trasero.
3. Disponga de un lugar adecuado para la conexión del módulo con una terminal ethernet disponible.

#### **A.3.3 Instalación del software para la configuración del módulo.**

El sistema trae adjunto el software que permite configurarlo por puerto serie o en forma remota. Por lo tanto, deberá ser instalado en la estación central del banco, y opcionalmente en algún computador en la entidad que tenga el sistema de alarma.

1. En el CD de instalación, haga doble clic en el icono denominado SMR-NET 8900. Este es el programa de instalación del software.
2. Siga los pasos que le indica el instalador.
3. Una vez instalado el programa, ya podrá ser utilizado para configurar el módulo.

### A.3.4 Configuración del módulo por el puerto serie.

La finalidad de configurar el módulo por medio de este puerto es para dejarlo preparado, desde la central del banco, para la instalación en una entidad lejana. También es necesaria cuando se requiera realizar la configuración en forma presencial por algún encargado autorizado, siempre y cuando se tenga un computador con puerto serie debidamente habilitado.

1. Conecte el módulo a la alimentación. Recuerde que debe ser 9 voltios.
2. Conecte el módulo al computador, utilizando para ello un cable serial. Verifique que el puerto serie del computador esté habilitado, en el panel de control.
3. Ejecute el programa SMR-NET 8900. En la siguiente figura se puede observar la interfaz con el usuario de este programa. Seleccione, en las opciones de comunicación, Puerto Serie.

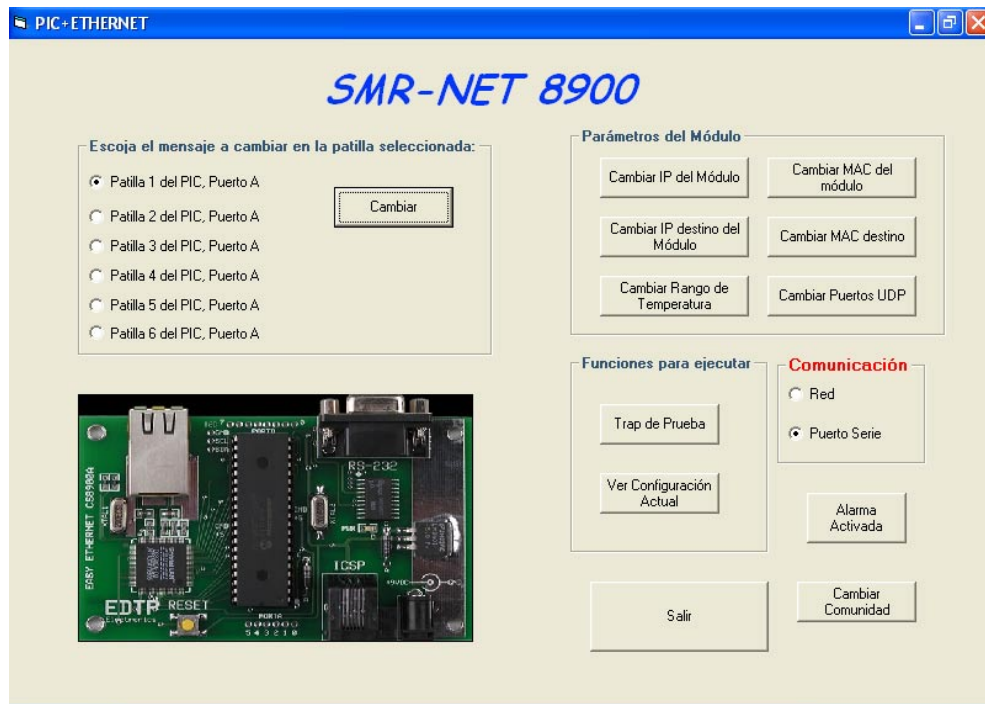
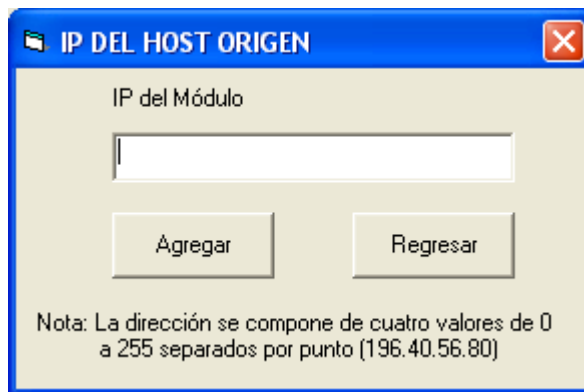


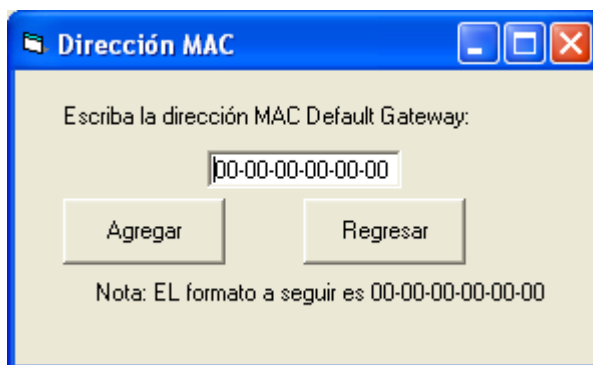
Figura A.1 Vista en pantalla del programa de configuración del módulo SMR-NET 8900.

4. Escriba cada uno de los mensajes que desea para que sean enviados en forma de avisos. Estos mensajes van de acuerdo al contacto. Así, por ejemplo, patilla dos puede ser el contacto de la puerta delantera, implicando que se debe escribir un mensaje de acuerdo a este lugar del armario.
5. Proceda a cambiar la dirección IP del módulo, presionando el botón para esta opción. Esta es la dirección que va a identificar al sistema de alarma en la red en la cual se instale. La ventana para modificar este parámetro es similar a la que se muestra a continuación:



**Figura A.2** Ventana que permite introducir la dirección IP del sistema.

6. Ahora cambie la dirección IP destino, es decir, la dirección del computador que tendrá a cargo el monitoreo de las actividades del módulo. Recuerde que tanto la dirección IP origen como la destino deben de introducirse siguiendo el formato que separa a cada número con puntos.
7. Una vez que se tienen debidamente configuradas las direcciones IP se tiene que proceder a dotar al sistema de las direcciones físicas MAC. Para cambiar la dirección MAC destino se debe de introducir la MAC del Default Gateway para el correcto enrutamiento de las tramas, y siguiendo las instrucciones de la ventana que se muestra a continuación. La dirección MAC del módulo se configura de manera similar.



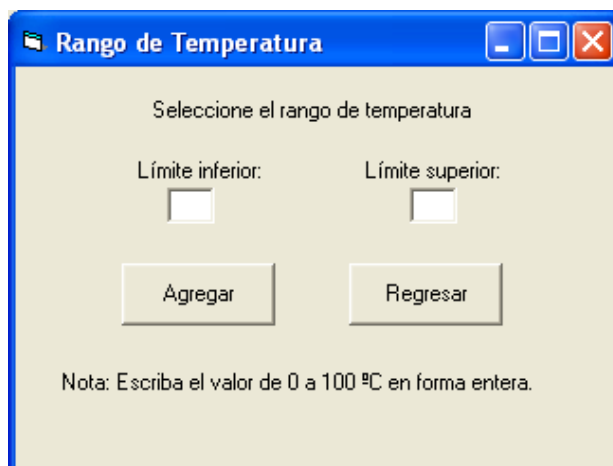
**Figura A.3** Ventana que permite introducir la dirección MAC destino.

8. Seguidamente se deben introducir los puertos origen y destino por los cuales se van a enviar los traps de alarma y de temperatura. Como se observa en la figura A.4, corresponden a los puertos de escucha y recepción del sistema. Debe de recordarse que el valor máximo permitido para cada uno es 65535.



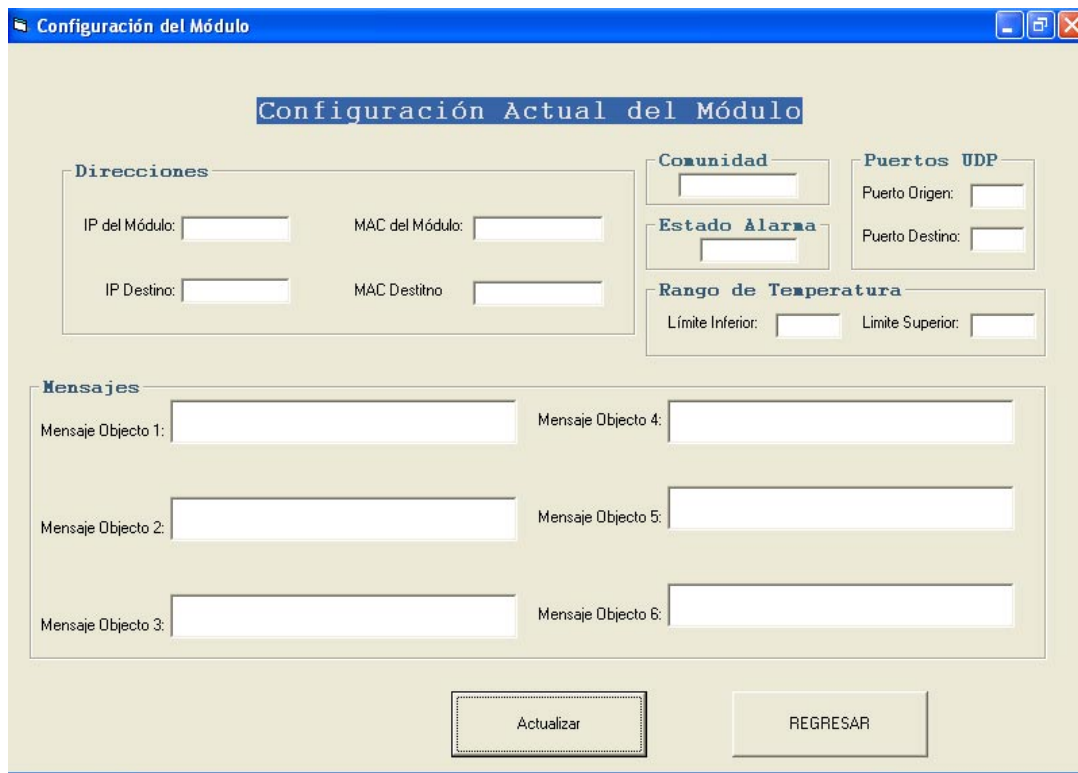
**Figura A.4** Vista en pantalla de la ventana para la configuración de los puertos.

9. Finalmente se deben introducir los valores máximo y mínimo de temperatura que el armario puede tolerar. Se modifican presionando el botón de cambiar rango de temperatura, y escribiendo los valores en la ventana hecha para este parámetro. Recuerde que como mínimo valor permitido se estableció cero grados centígrados, y como máximo cien grados centígrados. La figura A.5 muestra el aspecto de la ventana.



**Figura A.5** Vista en pantalla de la ventana para la variación del rango de temperatura.

10. De esta forma se prepara el módulo utilizando comunicación por puerto serie para que pueda ser utilizado. Para verificación, puede elegir “Ver configuración actual”, ventana similar a la que se muestra en la figura A.5.



**Figura A.6** Ventana que permite visualizar la configuración realizada al sistema, por puerto serie.

### A.3.5 Configuración del módulo por red.

Esta es la configuración que se debe realizar cuando el módulo se encuentra lejos de la estación central. Incluso, puede realizarse primero la configuración por red que por puerto serie, dependiendo de las posibilidades de que se disponga en un momento dado.

1. Un encargado en la entidad remota debe verificar la conexión del módulo a la alimentación. Recuerde que debe de ser 9 voltios.
2. El sistema debe conectarse al puerto ethernet del enrutador o hub de que se disponga en el armario de equipo de redes. También lo debe hacer el encargado de la entidad remota.
3. A continuación, en el programa de configuración SMR-NET 8900 (previamente instalado en la estación central), seleccione, en comunicación, la opción “Red”.
4. Inmediatamente le aparecerá la ventana de la figura A.7. En ella debe de escribir la dirección IP del módulo remoto de la entidad donde se instaló. También se deben incluir los dos puertos del sistema. Una vez introducidos estos tres parámetros la comunicación por red queda establecida entre el módulo y el computador de la estación central.



**Figura A.7** Ventana para introducir la dirección IP remota y los puertos del módulo.

5. A continuación debe escribir los mensajes asignados cada uno de los contactos magnéticos, las direcciones IP y físicas, los puertos y el rango de temperatura de manera similar a la explicada en los pasos del apartado A.3.4.
6. Una vez introducidos los anteriores parámetros, debe verificar que la configuración es la correcta. Para ello, presione la opción “Ver Configuración Actual”. La ventana, que se muestra en la figura A.8, no muestra los mensajes digitados para cada sensor, pero utilizando la opción “Trap de prueba” se pueden visualizar de forma remota.

The screenshot shows a window titled "Configuración del Módulo" with a blue title bar. Inside, there's a section titled "Configuración Actual del Módulo" in a blue header. Below this, there are several input fields organized into boxes:

- Direcciones:** A box containing four fields: "IP del Módulo:", "MAC del Módulo:", "IP Destino:", and "MAC Default Gateway".
- Puertos UDP:** A box containing two fields: "Puerto Origen:" and "Puerto Destino:".
- Estado Alarma:** A separate box containing one field.
- Rango de Temperatura:** A box containing two fields: "Límite Inferior:" and "Limite Superior:".

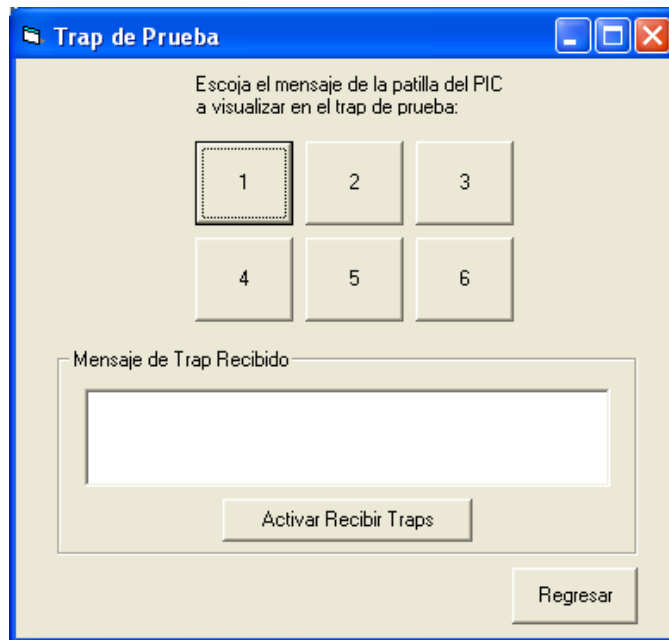
At the bottom of the window, there are two buttons: "Actualizar" (highlighted with a dashed border) and "REGRESAR".

**Figura A.8** Ventana para verificar la correcta configuración del sistema de alarma, por red.

### Opción “Trap de prueba”

Esta función del programa de configuración permite que el usuario pueda verificar el correcto envío de los traps de alerta a la estación central. La ventana, con los seis botones correspondientes a cada uno de los mensajes, es la que se muestra en la figura A.9. Debe de ejecutarse inmediatamente después de configurar el sistema de alarma, ya sea por puerto serie o por red. La forma de verificar la llegada de los traps de prueba es mediante el WhatsUp®, seleccionando la opción de Debug, en el menú Log. Si se desea verificar el sensor de temperatura, debe presionar el botón “1”.





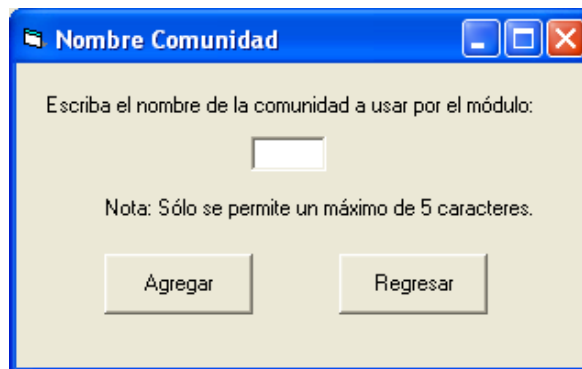
**Figura A.9** Ventana que permite enviar traps de prueba por la red.

### **Opción “Alarma activada/ desactivada”**

Esta función se debe ejecutar cuando algún encargado de soporte desee abrir una puerta del armario de equipo de redes para configurar al enrutador, limpiar el interior o cambiar algún equipo. Debe de presionarse para que la alarma se desactive mientras la puerta se encuentre abierta, para que el sistema no envíe traps de alerta.

### **Opción “Cambiar Comunidad”**


Al presionar este botón aparece en la pantalla la ventana que se muestra en la figura A.10. Permite cambiar la comunidad de los traps.



**Figura A.10** Ventana que permite cambiar la comunidad de los traps.

## ANEXOS

### Anexo B.1 Especificaciones y diagrama del controlador CS8900A.



# CS8900A

Product Data Sheet

---

### FEATURES

- Single-Chip IEEE 802.3 Ethernet Controller with Direct ISA-Bus Interface
- Maximum Current Consumption = 55 mA (5V Supply)
- 3 V Operation
- Industrial Temperature Range
- Comprehensive Suite of Software Drivers Available
- Efficient PacketPage™ Architecture Operates in I/O and Memory Space, and as DMA Slave
- Full Duplex Operation
- On-Chip RAM Buffers Transmit and Receive Frames
- 10BASE-T Port with Analog Filters, Provides:
  - Automatic Polarity Detection and Correction
- AUI Port for 10BASE2, 10BASE5 and 10BASE-F
- Programmable Transmit Features:
  - Automatic Re-transmission on Collision
  - Automatic Padding and CRC Generation
- Programmable Receive Features:
  - Stream Transfer™ for Reduced CPU Overhead
  - Auto-Switch Between DMA and On-Chip Memory
  - Early Interrupts for Frame Pre-Processing
  - Automatic Rejection of Erroneous Packets
- EEPROM Support for Jumperless Configuration
- Boot PROM Support for Diskless Systems
- Boundary Scan and Loopback Test
- LED Drivers for Link Status and LAN Activity
- Standby and Suspend Sleep Modes

### Crystal LAN™ ISA Ethernet Controller

### DESCRIPTION

The CS8900A is a low-cost Ethernet LAN Controller optimized for Industry Standard Architecture (ISA) Personal Computers. Its highly-integrated design eliminates the need for costly external components required by other Ethernet controllers. The CS8900A includes on-chip RAM, 10BASE-T transmit and receive filters, and a direct ISA-Bus interface with 24 mA Drivers.

In addition to high integration, the CS8900A offers a broad range of performance features and configuration-options. Its unique PacketPage architecture automatically adapts to changing network traffic patterns and available system resources. The result is increased system efficiency.

The CS8900A is available in a 100-pin TQFP package ideally suited for small form-factor, cost-sensitive Ethernet applications. With the CS8900A, system engineers can design a complete Ethernet circuit that occupies less than 1.5 square inches (10 sq. cm) of board space.

### ORDERING INFORMATION

CS8900A-CQ	0° to 70° C	5V	TQFP-100
CS8900A-IQ	-40° to 85° C	5V	TQFP-100
CS8900A-CQ3	0° to 70° C	3.3V	TQFP-100
CS8900A-IQ3	-40° to 85° C	3.3V	TQFP-100
CRD8900A-1			Evaluation Kit

Figura B1.1 Características y descripción general del controlador ethernet CS8900A

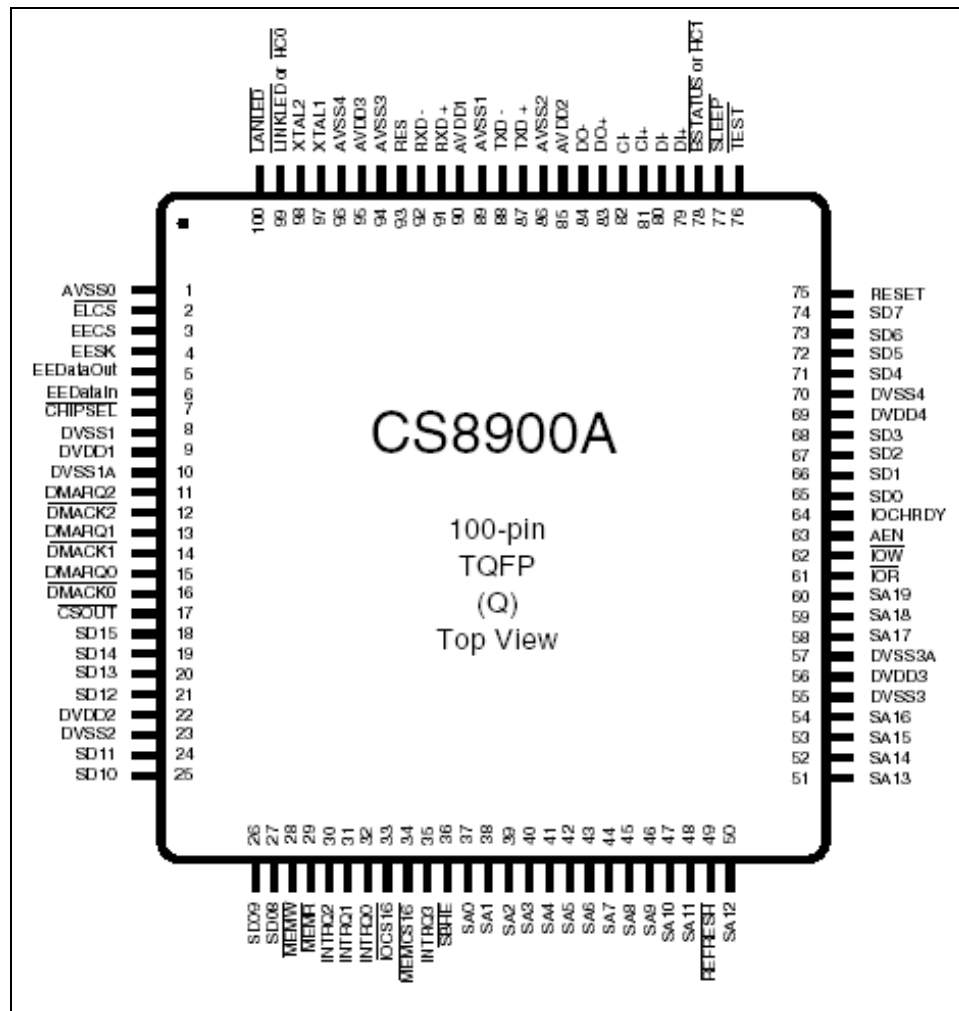


Figura B1.2 Diagrama del controlador ethernet CS8900A

## Anexo B.2 Especificaciones técnicas y diagrama del microcontrolador PIC16F877

<p><b>Devices Included in this Data Sheet:</b></p> <ul style="list-style-type: none"> <li>• PIC16F873A</li> <li>• PIC16F874A</li> <li>• PIC16F876A</li> <li>• PIC16F877A</li> </ul> <p><b>High Performance RISC CPU:</b></p> <ul style="list-style-type: none"> <li>• Only 35 single word instructions to learn</li> <li>• All single cycle instructions except for program branches, which are two-cycle</li> <li>• Operating speed: DC - 20 MHz clock input DC - 200 ns instruction cycle</li> <li>• Up to 8K x 14 words of FLASH Program Memory, Up to 368 x 8 bytes of Data Memory (RAM), Up to 256 x 8 bytes of EEPROM Data Memory</li> <li>• Pinout compatible to other 28-pin or 40/44-pin PIC16CXXX and PIC16FXXX microcontrollers</li> </ul> <p><b>Peripheral Features:</b></p> <ul style="list-style-type: none"> <li>• Timer0: 8-bit timer/counter with 8-bit prescaler</li> <li>• Timer1: 16-bit timer/counter with prescaler, can be incremented during SLEEP via external crystal/clock</li> <li>• Timer2: 8-bit timer/counter with 8-bit period register, prescaler and postscaler</li> <li>• Two Capture, Compare, PWM modules             <ul style="list-style-type: none"> <li>- Capture is 16-bit, max. resolution is 12.5 ns</li> <li>- Compare is 16-bit, max. resolution is 200 ns</li> <li>- PWM max. resolution is 10-bit</li> </ul> </li> <li>• Synchronous Serial Port (SSP) with SPI™ (Master mode) and I<sup>2</sup>C™ (Master/Slave)</li> <li>• Universal Synchronous Asynchronous Receiver Transmitter (USART/SCI) with 9-bit address detection</li> <li>• Parallel Slave Port (PSP) 8-bits wide, with external RD, WR and CS controls (40/44-pin only)</li> <li>• Brown-out detection circuitry for Brown-out Reset (BOR)</li> </ul>	<p><b>Analog Features:</b></p> <ul style="list-style-type: none"> <li>• 10-bit, up to 8 channel Analog-to-Digital Converter (A/D)</li> <li>• Brown-out Reset (BOR)</li> <li>• Analog Comparator module with:             <ul style="list-style-type: none"> <li>- Two analog comparators</li> <li>- Programmable on-chip voltage reference (VREF) module</li> <li>- Programmable input multiplexing from device inputs and internal voltage reference</li> <li>- Comparator outputs are externally accessible</li> </ul> </li> </ul> <p><b>Special Microcontroller Features:</b></p> <ul style="list-style-type: none"> <li>• 100,000 erase/write cycle Enhanced FLASH program memory typical</li> <li>• 1,000,000 erase/write cycle Data EEPROM memory typical</li> <li>• Data EEPROM Retention &gt; 40 years</li> <li>• Self-reprogrammable under software control</li> <li>• In-Circuit Serial Programming™ (ICSP™) via two pins</li> <li>• Single supply 5V In-Circuit Serial Programming</li> <li>• Watchdog Timer (WDT) with its own on-chip RC oscillator for reliable operation</li> <li>• Programmable code protection</li> <li>• Power saving SLEEP mode</li> <li>• Selectable oscillator options</li> <li>• In-Circuit Debug (ICD) via two pins</li> </ul> <p><b>CMOS Technology:</b></p> <ul style="list-style-type: none"> <li>• Low power, high speed FLASH/EEPROM technology</li> <li>• Fully static design</li> <li>• Wide operating voltage range (2.0V to 5.5V)</li> <li>• Commercial and Industrial temperature ranges</li> <li>• Low power consumption</li> </ul>
--	---

Figura B2.1 Características y descripción del microcontrolador PIC 16F877

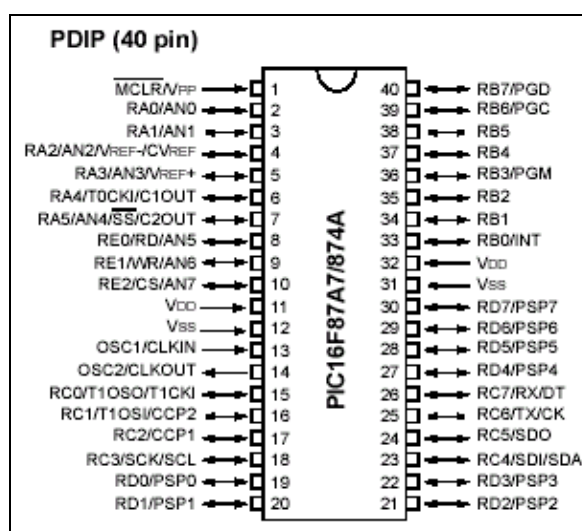
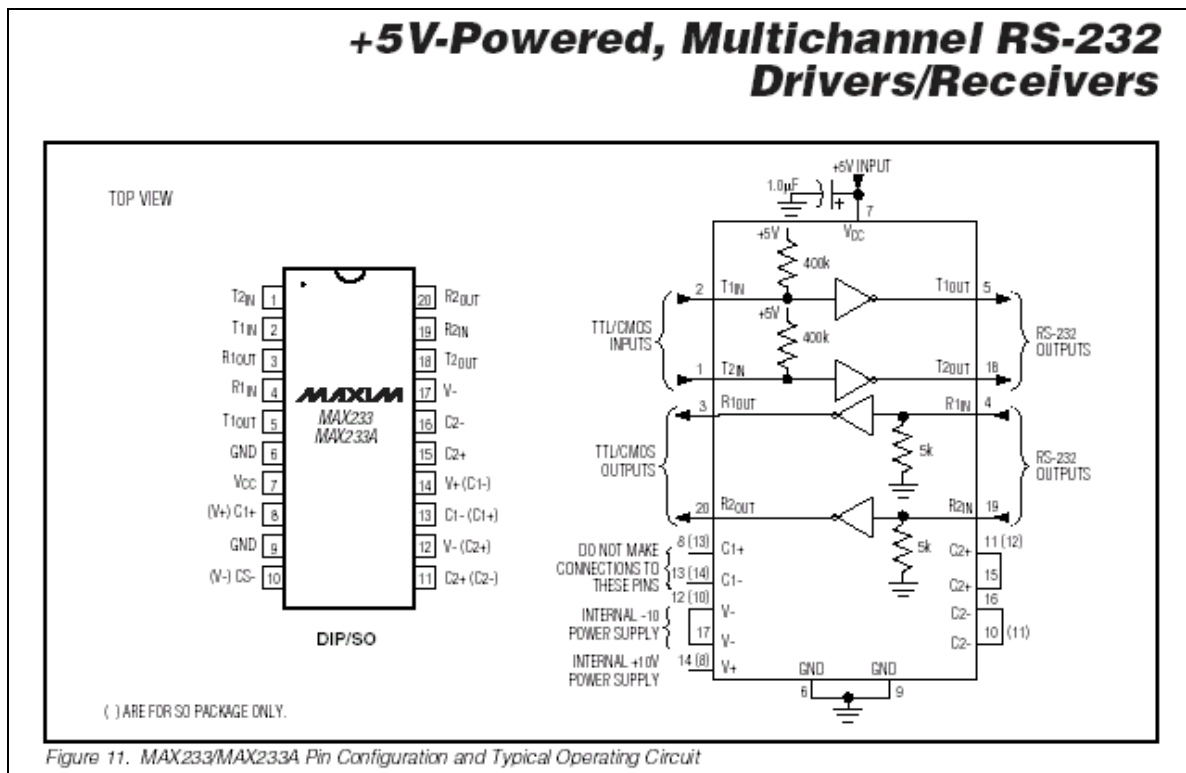



Figura B2.2 Características y descripción del microcontrolador PIC 16F877

**Anexo B.3** Diagrama del MAX233 para comunicación por puerto serie



**Figura B3.1** Diagrama del manejador MAX233

**Anexo B.4** Especificaciones técnicas y diagrama del convertidor de temperatura a voltaje de alta precisión



# TC1046

---

## High Precision Temperature-to-Voltage Converter

---

### Features

- Supply Voltage Range: 2.7V to 4.4V
- Wide Temperature Measurement Range: -40°C to +125°C
- High Temperature Converter Accuracy:  $\pm 2^\circ\text{C}$ , Max, at 25°C
- Linear Temperature Slope: 6.25mV/°C
- Very Low Supply Current: 35 $\mu\text{A}$  Typical
- Small 3-Pin SOT-23B Package

### Applications

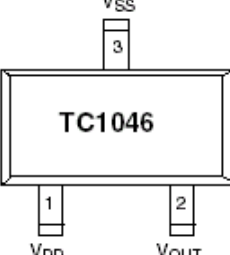
- Cellular Phones
- Power Supply Thermal Shutdown
- Temperature Controlled Fans
- Temperature Measurement/Instrumentation
- Temperature Regulators
- Consumer Electronics
- Portable Battery Powered Equipment

### Device Selection Table

Part Number	Package	Temp. Range
TC1046VNB	3-Pin SOT-23B	-40°C to +125°C

### Package Type

3-Pin SOT-23B\*



Note: \*3-Pin SOT-23B is equivalent to JEDEC TO-236

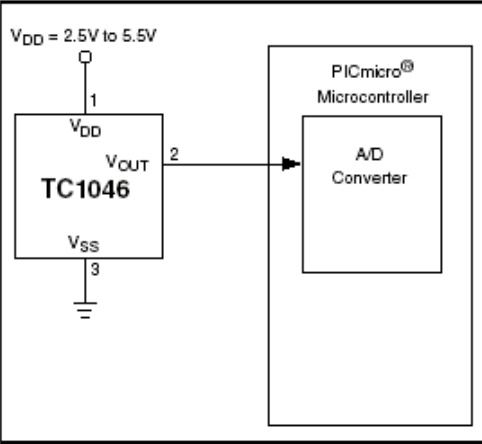
### General Description

The TC1046 is a linear output temperature sensor whose output voltage is directly proportional to measured temperature. The TC1046 can accurately measure temperature from -40°C to +125°C.

The output voltage range for these devices is typically 174mV at -40°C, 424mV at 0°C, 580 mV at +25°C, and 1205mV at +125°C. A 6.25mV/°C voltage slope allows for the wide temperature range.

The TC1046 is packaged in a 3-Pin SOT-23B package, making them ideal for space critical applications.

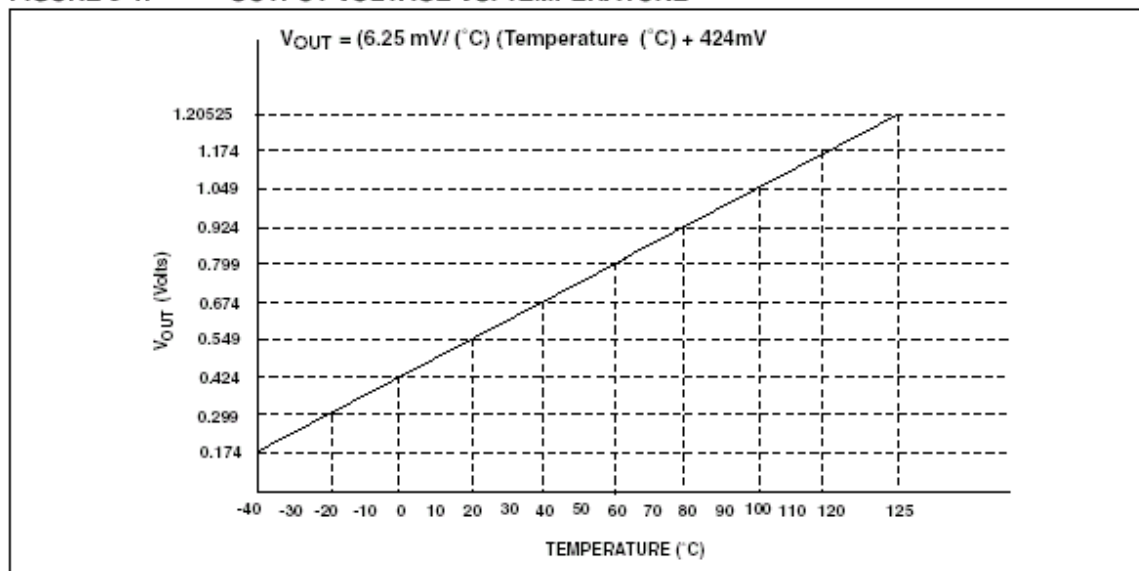
### Functional Block Diagram



**Figura B4.1** Características y descripción del sensor de temperatura TC1046

The TC1046 has an output voltage that varies linearly with temperature in degrees Celsius. See Figure 3-1, “Output Voltage versus Temperature” for the TC1046. The temperature slope is fixed at 6.25mV/°C and the output voltage at 0°C is 424mV.

**FIGURE 3-1: OUTPUT VOLTAGE VS. TEMPERATURE**



**Figura B4.2** Curva característica del sensor de temperatura TC1046