

**Instituto Tecnológico de Costa Rica**

**Escuela de Ingeniería Electrónica**



**Sistema de reporte de eventos en la Estación Cerro San José**

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

Jorge Salguero Quirós

Cartago, Setiembre 10

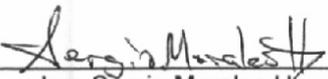
II semestre de 2008

**INSTITUTO TECNOLÓGICO DE COSTA RICA**  
**ESCUELA DE INGENIERÍA ELECTRÓNICA**  
**PROYECTO DE GRADUACIÓN**  
**TRIBUNAL EVALUADOR**

Proyecto de Graduación defendido ante el presente Tribunal Evaluador como requisito para optar por el título de Ingeniero en Electrónica con el grado académico de Licenciatura, del Instituto Tecnológico de Costa Rica.

**Miembros del Tribunal**

  
\_\_\_\_\_  
Ing. Anibal Coto Cortés  
Profesor lector

  
\_\_\_\_\_  
Ing. Sergio Morales Hernández  
Profesor lector

  
\_\_\_\_\_  
Ing. Julio Stradi Granados  
Profesor asesor

Los miembros de este Tribunal dan fe de que el presente trabajo de graduación ha sido aprobado y cumple con las normas establecidas por la Escuela de Ingeniería Electrónica

Escuela de Ingeniería Electrónica, setiembre 8, 2008

Declaro que el presente Proyecto de Graduación ha sido realizado enteramente por mi persona, utilizando y aplicando literatura referente al tema e introduciendo conocimientos propios.

En los casos en que he utilizado bibliografía, he procedido a indicar las fuentes mediante las respectivas citas bibliográficas.

En consecuencia, asumo la responsabilidad total por el trabajo de graduación realizado y por el contenido del correspondiente informe final.

Cartago, 10 de Setiembre de 2008

A handwritten signature in black ink, appearing to read 'Jorge Salguero Quirós', written in a cursive style.

Jorge Salguero Quirós

Cédula: 5-332-430

## **Resumen**

En la Estación Cerro San José (CSJ) se encuentra equipo de transmisión e interconexión que permite el traslado de información (telefonía, Internet, etc.) entre distintas ciudades de la zona de la provincia de Guanacaste. Cuando un equipo reporta un fallo debe ser reparado en la brevedad de tiempo posible, por el personal del ICE para que, esto no afecte el servicio prestado a sus usuarios. Cada uno de estos equipos posee un punto de alarma que indica un fallo. Además, cambios en el nivel CA de la red de alimentación puede provocar un error de configuración en los equipos de la estación.

Como solución propuesta para agilizar la reparación de un error o evento en la estación, se implementó un sistema de comunicación cliente/servidor vía TCP/IP, entre la estación a monitorear (CSJ) y la oficina de supervisión de equipos de transporte (encargada de vigilancia en el funcionamiento). En este documento el lector puede encontrar los detalles de implementación.

**Palabras clave:** Cliente, servidor, tensión rms, TCP, IP, PIC.

## **Summary**

In the Station Hill San José (CSJ) there are transmission devices and interconnection that it allows the transmission of information (telephony, Internet, etc.) among different cities of the area of the county of Guanacaste. When a device report a failure the quickest thing it should be repaired possible for the personnel of the ICE, so that this doesn't affect the borrowed service to its users. Each one of these devices possesses an alarm point that it indicates a failure. Also, changes in the level CA of the feeding net can cause a configuration error in the devices of the station. As solution proposed to speed up the repair of an error or event in the station, a system of communication client/servant was implemented using TCP/IP, between the station to verify (CSJ) and the office GRM Transmission (in charge of surveillance in the operation). The implementation details are detailed in this document.

**Keywords:** Client, servant, voltage rms, TCP, IP.

## **Dedicatoria**

*A mi familia y a todas aquellas personas  
que de una u otra forma me tendieron  
su mano.*

## **Agradecimiento**

*Al Ing. Julio Stradi por su colaboración  
durante el desarrollo del proyecto.*

## ÍNDICE GENERAL

Capítulo 1: Introducción .....	10
1.1 Problema existente e importancia de su solución .....	10
1.1.1 Definición del problema .....	12
1.1.2 Importancia de resolver el problema .....	13
1.2 Solución seleccionada .....	13
Capítulo 2: Meta y objetivos .....	15
2.1 Meta .....	15
2.2 Objetivo general .....	16
2.3 Objetivos específicos .....	17
2.3.1 Objetivos de hardware .....	17
2.3.2 Objetivos de software .....	17
2.3.3 Objetivos de documentación .....	18
Capítulo 3: Marco teórico .....	19
3.1 Descripción del sistema a mejorar .....	19
3.2 Antecedentes Bibliográficos .....	21
3.2.1 A nivel de programa en el PC .....	21
3.2.2 A nivel del programa en la Unidad de Control .....	21
3.3 Descripción de los principios físicos y/o electrónicos relacionados con la solución del problema .....	22
3.3.1 Conceptos generales de redes de computadoras .....	22
Capítulo 4: Procedimiento Metodológico .....	27
4.1 Reconocimiento y definición del problema .....	27
4.2 Obtención y análisis de la información .....	27
4.3 Evaluación de las alternativas y síntesis de una solución .....	28
4.4 Implementación de la solución .....	28
4.5 Reevaluación y rediseño .....	29
Capítulo 5: Descripción detallada de la solución .....	30
5.1 Análisis de soluciones y selección final .....	30
5.1.1 Establecimiento de la comunicación .....	30
5.1.2 Selección de lenguajes de programación .....	32
5.1.3 Implementación y diseño .....	33
5.2 Descripción del hardware .....	34
5.2.1 Acople de tensiones .....	34
5.2.2 Detección de cambios en el nivel CA .....	38
5.3 Descripción del software .....	43
Capítulo 6: Análisis de Resultados .....	52
6.1 Resultados Experimentales y Análisis .....	52
6.1.1 A nivel de software: .....	52
6.1.2 A nivel de hardware: .....	56
Capítulo 7: Conclusiones y recomendaciones .....	61
7.1 Conclusiones .....	61
7.2 Recomendaciones .....	63
Bibliografía .....	64
Apéndices y anexos .....	66
A1. Glosario .....	66

A2.	Manual de usuario.....	67
A3.	Manual de mantenimiento .....	79

## ÍNDICE DE FIGURAS

Figura 1.1.	Estación Cerro San José como punto de interconexión.....	11
Figura 1.2.	Esquema del equipo en la estación. ....	11
Figura 1.3.	Diagrama general del sistema implementado. ....	14
Figura 3.1.	Sistema de reporte actual de fallos en Estación Cerro San José. ...	19
Figura 3.2.	Diagrama de flujo de reporte de fallos en la Estación Cerro San José. ....	20
Figura 3.3.	Diagrama general de una red WAN. ....	23
Figura 5.1.	Espectro de frecuencia de voz.....	30
Figura 5.2.	Acople entre canal de 64 kbits/s y equipo de transmisión-recepción. ....	31
Figura 5.3.	Sistema de comunicación cliente-servidor.....	32
Figura 5.4.	Diagrama del sistema cliente/servidor implementado en la red ICE. ....	33
Figura 5.5.	Etapa de acople entre las tensiones de alarmas y la UC.....	35
Figura 5.6.	Placa de acople para 32 entradas de alarmas. Cuatro módulos.....	36
	de 8 entradas cada uno. ....	36
Figura 5.7.	Diagrama de bloques del circuito implementado para detectar cambios de +/- 10% en el nivel CA de la señal de la red de alimentación. ....	38
Figura 5.8.	Diagrama de tiempos del circuito implementado para detectar cambios de +/- 10% en el nivel CA de la señal de la red de alimentación. ....	40
Figura 5.9.	Diagrama de circuitos comparadores.....	42
Figura 5.10.	Diagrama de flujo del programa principal.....	44
Figura 5.11.	Algoritmo implementado para el monitoreo de alarmas y envío de datos al servidor.....	45
Figura 5.12.	Diagrama de flujo del método Servidor(), implementado a nivel del PC.....	48
Figura 5.13.	Diagrama de flujo del método CaseAlarma(), implementado a nivel del PC.....	49
Figura 5.14.	Diagrama de flujo del método CaseBase(), implementado a nivel del PC.....	50
Figura 6.1.	Interfaz gráfica implementada en el PC, utilizando el lenguaje Visual C# Express Edition.....	52
Figura 6.2.	Mensaje de error en equipo remoto de la Estación Cerro San José.....	53
Figura 6.3.	Mensaje de evento en la Estación Cerro San José.....	54
Figura 6.4.	Mensaje de ingreso de persona a la Estación Cerro San José.....	54
Figura 6.5.	Forma de onda de la señal CA rectificada. Fotografía tomadas de un ORC y procesadas con Paint de Windows.....	56
Figura 6.6.	Señal de salida del detector de cruce por cero y salida de monoestable 1. Fotografía tomadas de un ORC y procesadas con Paint de Windows.....	57
Figura 6.7.	Señales de salida de circuitos monoestables. Monoestable 1 (arriba), monoestable 2 (abajo). Fotografía tomadas de un ORC y procesadas con Paint de Windows.....	57
Figura 6.8.	Sincronización de módulo para obtener el valor pico de la señal CA. Fotografía tomadas de un ORC y procesadas con Paint de Windows.....	58

Figura 6.9. Tensión almacenada en el capacitor. Fotografía tomadas de un ORC y procesadas con Paint de Windows .....	59
Figura 6.10. Línea de tendencia del valor pico en la señal CA al variar el.....	60
valor Vrms de la red de alimentación. Figura hecha con Excel de Microsof. ....	60

## ÍNDICE DE TABLAS

Tabla 5.1. Selección de módulos mediante polling .....	37
Tabla 5.2. Identificación del equipo en la Estación Cerro San José.....	47
Tabla 6.1. Tensiones de límites de alarmas al darse cambios .....	59
en el nivel RMS de la red. Para una fuente CD de 5 V .....	59
Tabla 6.2. Valor pico de la señal de la red durante variaciones de la tensión en el valor $V_{rms}$ de la red de alimentación .....	59

## **Capítulo 1: Introducción**

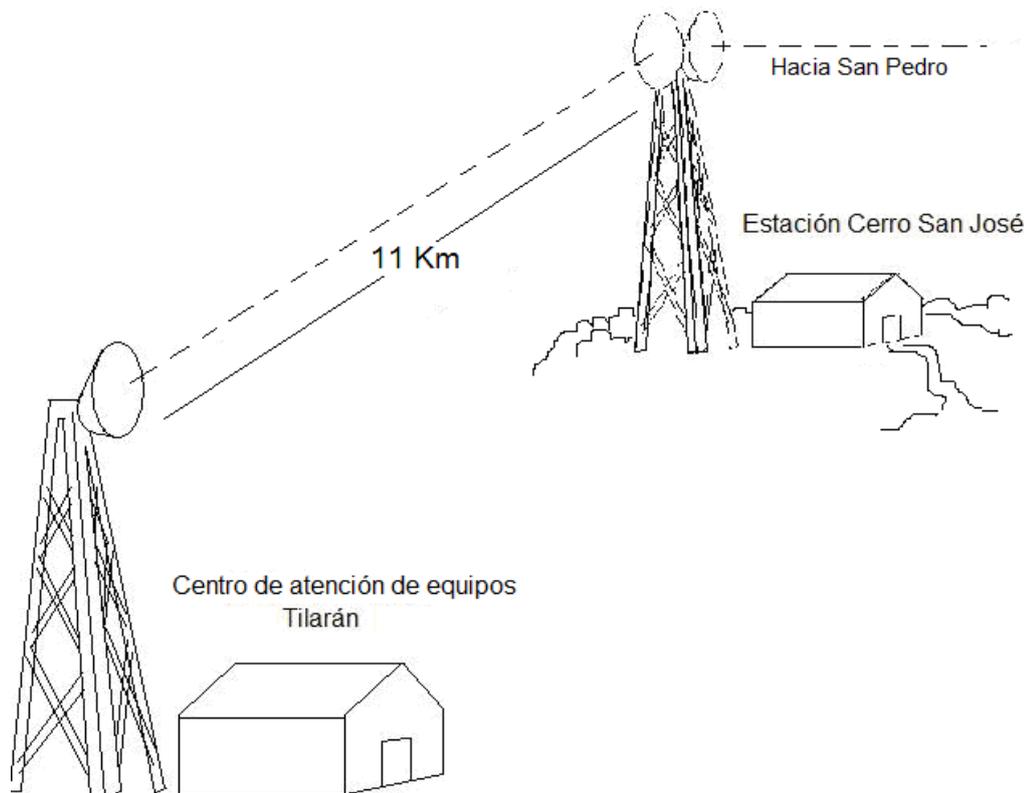
En este capítulo el lector encontrará la explicación del problema existente, el cual dio origen al desarrollo del proyecto y su importancia de solución. Además, se hará un breve resumen de la solución implementada en el proyecto, la cual se retoma con mayor detalle en el capítulo 5 de este documento.

### **1.1 Problema existente e importancia de su solución**

El Instituto Costarricense de Electricidad (ICE), es la principal compañía en la generación y distribución de electricidad en Costa Rica, para lo cual, cuenta con plantas hidroeléctricas, proyectos geotérmicos y eólicos en distintos lugares del país. Además, de los servicios de electricidad, el ICE tiene sectores dedicados al área de telecomunicaciones, tales como telefonía fija y celular, administración de Internet y servicios de redes privadas, entre otros.

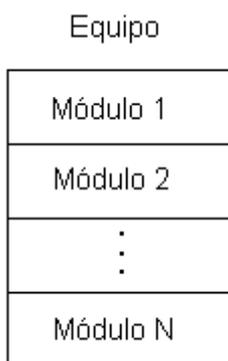
El sector de Proceso de Transporte (DTOM) del ICE es el responsable de hacer posible la comunicación entre distintos lugares, así como prestar los servicios de telecomunicaciones a lo largo del país. Además, este sector es el encargado de darle mantenimiento a cada uno de los servicios.

Debido a la geografía del país y la necesidad de comunicación entre zonas a lo largo del territorio nacional, son requeridos puntos de interconexión en lugares estratégicos entre dichas zonas. Estos puntos de conexión están ubicados en estaciones, en las cuales se encuentran los equipos y antenas para la conexión y administración de información. Para efectos de este proyecto, la estación en cuestión es conocida con el nombre de Estación Cerro San José (CSJ). Ésta es la encargada del tráfico y distribución de información hacia varias comunidades de la zona. Además, funciona como punto de interconexión con el Área Metropolitana. Se encuentra ubicada en Cerro San José, en el cantón de Tilarán, Guanacaste, a una distancia de 11 Km del Centro de Atención de Equipos de Transporte de la zona, encargada del mantenimiento y corrección de fallos en dicha estación.



**Figura 1.1.** Estación Cerro San José como punto de interconexión

El correcto funcionamiento del equipo en la estación, es importante para el desarrollo de las actividades cotidianas de los usuarios del ICE. Estos equipos se encuentran segmentados en módulos de operación, por lo que el funcionamiento correcto de un módulo, no esta sujeto al funcionamiento de los demás. Además, cada uno de estos módulos posee un sistema de alarma (activa una señal de alarma).



**Figura 1.2.** Esquema del equipo en la estación.

### 1.1.1 Definición del problema

La Estación Cerro San José, presenta ocasionales problemas en su funcionamiento normal (eventos), debido al fallo o avería de los distintos equipos y fluctuaciones en la amplitud del nivel CA nominal de la red de alimentación (120 Vrms). Limitando las variaciones del nivel CA, en un rango máximo de 10%, ya que valores mayores pueden ocasionar problemas de configuración en los equipos, además de afectar las interfaces de conexión.

Esta estación es vigilada remotamente desde San Pedro, del cual supervisa su correcto funcionamiento y reporta cualquier fallo (identificando el nivel de urgencia o prioridad del fallo según el equipo) al Centro de Atención de Equipo de Transporte, de donde se envía al personal a realizar una inspección en los equipos, para luego realizar su debida reparación o reemplazo. No obstante, es probable que también, sujeto al equipo que falle la comunicación CSJ – ICE San Pedro no sea posible y por consiguiente, no se pueda reportar el fallo para su eventual solución. Teniendo en estos casos, ser los usuarios los que informen sobre las averías por la falta en determinado servicio de telecomunicaciones. Esto ocasiona un gasto innecesario de tiempo. De ocurrir una eventual avería, se tendrá una respuesta lenta para volver a poner la estación en su normal funcionamiento, lo que implica una disminución en los servicios de telecomunicaciones a los usuarios del ICE.

[1] El término evento se refiere a dos situaciones:

- Fallo en el funcionamiento normal del equipo y fluctuaciones en el nivel CA

### **1.1.2 Importancia de resolver el problema**

El ICE como institución aspira al crecimiento en los mercados de servicios de telecomunicaciones. Este crecimiento implica una fuerte estructura que le de confianza a sus clientes, de manera que estos se sientan a gusto con los servicios prestados. De forma tal, pueda competir lado a lado con las empresas extranjeras venideras en los años siguientes.

Por tanto, la solución de este problema nos lleva a un mejoramiento en los servicios. Puesto que, al disminuir el tiempo de respuesta en el reporte de eventos, lleva a un deseable servicio continuo, con cortos periodos de no funcionamiento en los servicios para los usuarios.

Dentro de los beneficios que se obtienen al resolver el problema se pueden citar los siguientes:

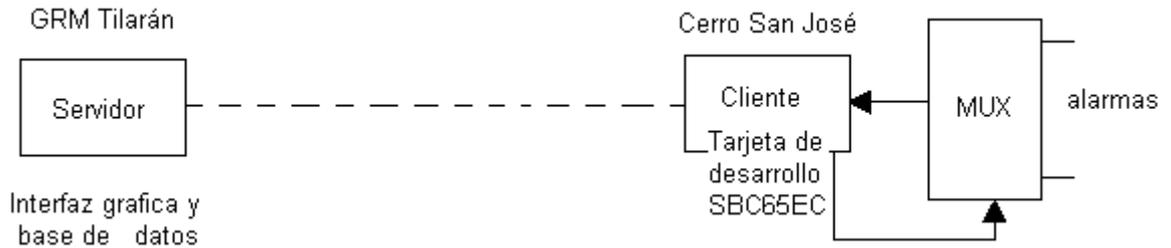
- a. Un canal auxiliar de reporte (sistema 1+1)
- b. Reporte inmediato al Centro de Atención de Equipos de Transporte
- c. Mayor velocidad en la corrección de averías
- d. Mejor servicio a los usuarios

### **1.2 Solución seleccionada**

En esta sección se presenta una breve introducción sobre la solución implementada según los requerimientos solicitados por el Instituto Costarricense de Electricidad finalizado el proyecto, los cuales se citan a continuación:

- a. Si se da una alarma en un equipo debe reportarse de manera inmediata.
- b. Todo evento debe registrarse en una base de datos.
- c. Se debe registrar la hora, fecha, equipo, tipo de evento y nivel de prioridad del evento.
- d. Reducción de costos.
- e. Sistema implementado escalable

Para obtener los resultados esperados por el ICE según los lineamientos mencionados anteriormente, se implementó un sistema de comunicación cliente/servidor utilizando el modelo de redes TCP/IP para realizar el reporte cada instante que se da un evento. En la figura siguiente se muestra el diagrama general del sistema implementado.



**Figura 1.3.** Diagrama general del sistema implementado.

Se desarrollo una aplicación gráfica, en la cual un PC actúa como servidor, además se implementó una base de datos. Para implementar el dispositivo cliente, se utilizó la tarjeta de desarrollo SBC65EC la cual utiliza el microcontrolador Microchip 18F6627. El PIC realiza un muestreo (polling) utilizando un MUX para ver cada una de las alarmas. Esta tarjeta se conecta a la red del ICE (Intranet), y comunica al servidor cuando se da un evento. El servidor es implementado en una PC mediante un programa en el que se registra los eventos enviados desde el cliente.

## **Capítulo 2: Meta y objetivos**

### **2.1 Meta**

Permitir un informe inmediato de eventos que alteren el funcionamiento normal de la Estación Cerro San José, debido al lento reporte actual de fallos.

Indicador: Reporte inmediato de eventos.

## **2.2 Objetivo general**

Implementar un sistema de reporte inmediato, de eventos que alteran el ambiente y funcionamiento normal de la Estación Cerro San José, para agilizar la corrección de fallos en la estación.

Indicador: Cuando ocurre un evento y de inmediato es notificado a la base central.

## **2.3 Objetivos específicos**

### **2.3.1 Objetivos de hardware**

1. Diseñar e implementar un módulo, que permita detectar variaciones respecto al nivel CA nominal de la red (120 Vrms) mayores de 10%.

Indicador: Al darse un cambio mayor de 10% en el nivel CA, se genera una señal indicando que ocurrió un evento.

### **2.3.2 Objetivos de software**

2. Diseñar e implementar un módulo para identificar mediante codificación binaria, cada uno de los equipos en la estación.

Indicador: Cada equipo en la estación posee un único y diferente código de identificación.

3. Establecer un sistema de comunicación entre la estación y la base principal utilizando un protocolo de comunicación, para realizar el reporte de eventos.

Indicador: Al ocurrir un evento en la estación, este se da a conocer de inmediato a la base principal, comprobando que se cumplió este objetivo.

4. Crear una interfaz gráfica en un lenguaje de programación de alto nivel, para registrar en una base de datos los eventos ocurridos en la estación.

Indicador: Luego de que ocurra un evento, se puede acceder a la base de datos y corroborar que ocurrió un evento.

### **2.3.3 Objetivos de documentación**

5. Crear el manual de usuario del sistema a implementar.

Indicador: Manual de usuario del sistema final.

6. Crear el manual de mantenimiento de hardware y software del sistema a implementar.

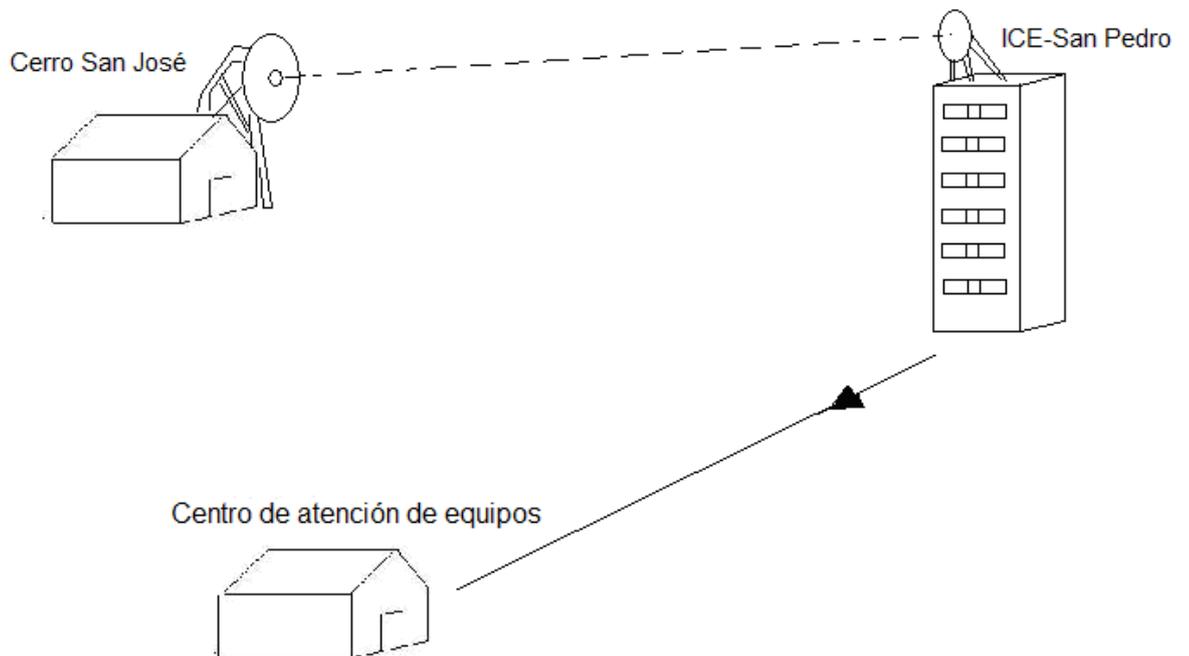
Indicador: Manual de mantenimiento de hardware y software del sistema final.

### Capítulo 3: Marco teórico

En este capítulo se resumen los principales fundamentos teóricos relacionados con física y electrónica que fueron aplicados para implementar una solución al problema planteado. Además, el lector podrá encontrar referencias bibliográficas consultadas. Las referencias presentadas son las utilizadas durante el desarrollo de la implementación final. Otras referencias consultadas durante la investigación para una selección de solución se presentan en la parte final de este documento (Bibliografía).

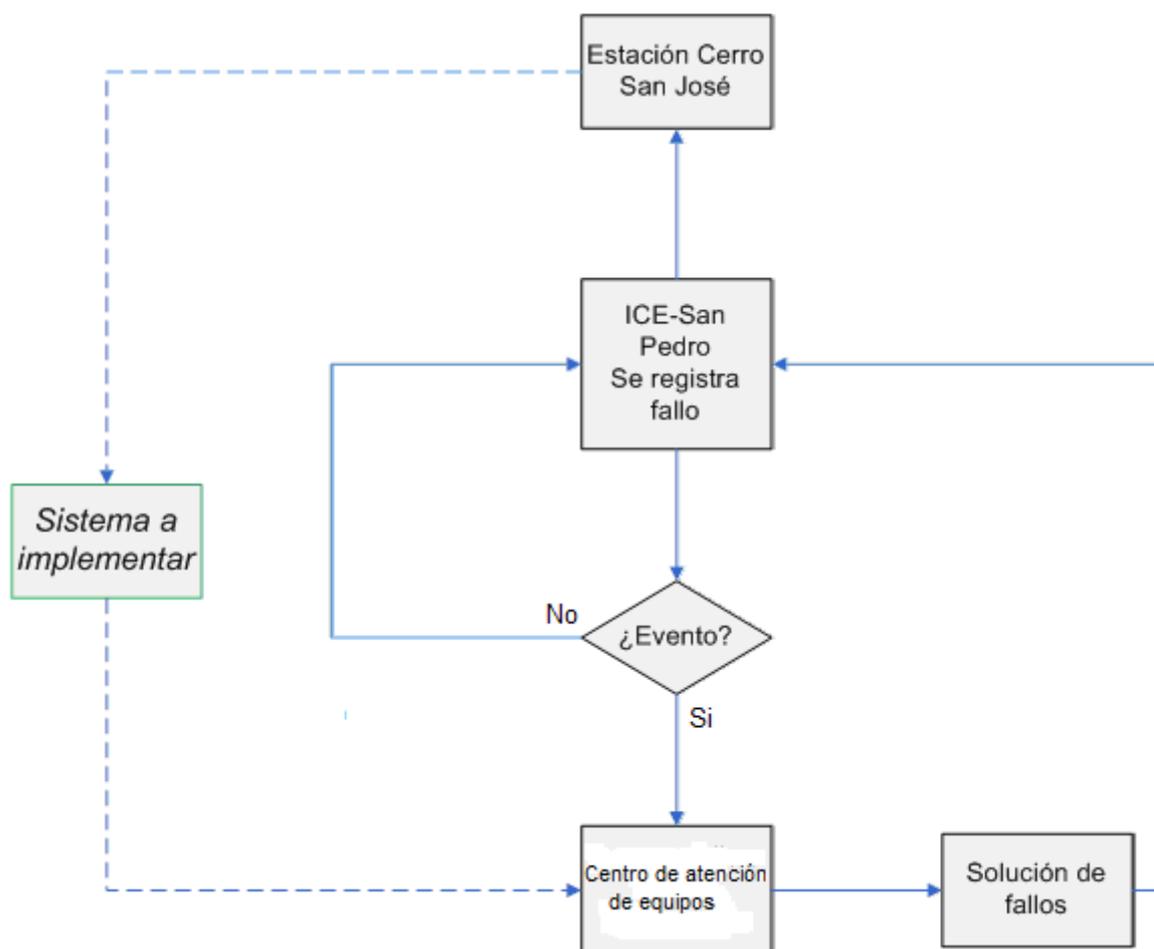
#### 3.1 Descripción del sistema a mejorar

Se quiere mejorar y darle mayor fiabilidad a la forma en que se realiza el reporte de fallos o eventos en la Estación Cerro San José. En la figura siguiente se muestra la forma en que se reportan los fallos en la estación.



**Figura 3.1.** Sistema de reporte actual de fallos en Estación Cerro San José.

Actualmente, cuando una alarma se activa en la Estación Cerro San José, esta es reportada al ICE de San Pedro, por medio de canales de conexión directa, los cuales utilizan multiplexación de frecuencia para la transmisión de información. Se registra el tipo de fallo según el nivel de prioridad, para luego ser reportada al centro de gestión de equipos. No obstante, como se mencionó en la definición del problema, si se daña el transporte de alarmas, esto dificulta el diagnóstico local en los equipos. En el diagrama de flujo de la figura siguiente se detalla el sistema actual de reporte de eventos implementado. Y en la línea punteada se marca el sistema que se desea implementar durante el desarrollo del proyecto.



**Figura 3.2.** Diagrama de flujo de reporte de fallos en la Estación Cerro San José.

## **3.2 Antecedentes Bibliográficos**

Básicamente fueron dos aspectos para la implementación del proyecto, los que tomaron mayor relevancia durante la investigación. Estos fueron la implementación del sistema remoto o unidad de control, encargado del control de alarmas; y la implementación de la interfaz gráfica a nivel del PC en el Centro de Atención de Equipo.

### **3.2.1 A nivel de programa en el PC**

Para desarrollar la aplicación en el PC, la principal investigación se hizo en dos sitios de Internet ((4) y (5)), los cuales presentan ayudas para el manejo del lenguaje Visual C# Express Edition y en general de los lenguajes de Visual Studio (MSDN). En estos sitios fue posible encontrar las bibliotecas disponibles de este lenguaje, además de ejemplos de aplicaciones básicas. No obstante, estas ayudas también se encuentran incorporadas en las opciones de instalación de Visual C# Express, por tanto, es posible utilizarlas con y sin conexión a Internet.

Para la implementación de la base de datos, principalmente se hizo de la información encontrada en (3) además, de las ayudas que ofrece Visual C# Express. Esta referencia se utilizó como guía acerca del manejo de bases de datos con SQL Server Express.

### **3.2.2 A nivel del programa en la Unidad de Control**

La información obtenida para implementar el programa a nivel de la UC, se obtuvo principalmente del artículo AN833 (9) provisto por Microchip acerca de la estructura TCP/IP a nivel de microcontroladores, de manera específica, del capítulo Transmission Control Protocol (TCP), por lo que, la estructura de conexión y envío de datos, implementada en la UC está basada en este documento.

La utilización de la tarjeta de desarrollo SBC65EC de la compañía Modtronix Engineering, facilitó el uso de aplicaciones y ayudas accesibles desde su página Web (8), sobre conexiones de redes utilizando tarjetas de desarrollo con microcontroladores Microchip, creados por Modtronix para sus clientes (6). Las cuales fueron utilizadas como estructura para el desarrollo de la comunicación a través de la red. Además, desde el sitio Web de esta compañía se obtuvo la explicación en detalle de las capacidades de la tarjeta utilizada la implementación del proyecto.

### **3.3 Descripción de los principios físicos y/o electrónicos relacionados con la solución del problema**

#### **3.3.1 Conceptos generales de redes de computadoras**

La transmisión de mensajes a través de una red, desde una fuente a un destino, conlleva una serie de componentes necesarios para alcanzar este objetivo. Dentro de estos dispositivos se pueden citar:

**Servidores:** Son host que tienen software instalados para que sean capaces de suministrar información a otros dispositivos de la red.

**Cliente:** Son host que tienen software instalados para solicitar y mostrar información proveniente de un determinado servidor.

**Dispositivos intermedios:** Estos son los encargados de proveer conectividad y dirigir el flujo de datos a través de la red. Ellos conectan host individuales y también unen diferentes redes para formar una Internetwork. Dentro de estos dispositivos se pueden mencionar: Hubs, Switches, Routers, etc.

**Medio físico:** Es el medio por el cual, el mensaje es transmitido de una fuente a un destino. El mecanismo utilizado depende del medio utilizado, por ejemplo: cable de cobre (impulsos eléctricos), fibra óptica (pulsos de luz), transmisión inalámbrica (ondas electromagnéticas), etc.

### Redes según su ubicación geográfica, fin y organización

Las redes se pueden clasificar en distintos grupos:

**Red LAN:** Es una red de computadoras a nivel de una organización. Las cuales por lo general tienen carácter privado.

**Red WAN:** Son redes de una organización que se encuentran separadas por distancias muy largas por los que requieren un TSP (Telecommunication Service Provider) para poder comunicarse entre ellas.

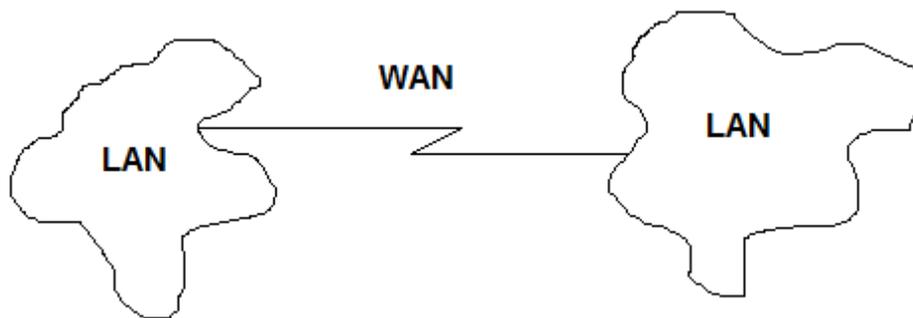


Figura 3.3. Diagrama general de una red WAN.

**Intranet:** Este termino es usado para referirse a conexiones privadas LAN y WAN propias de una organización y usados para ser accedidos solo por miembros de la organización. Desde los host internos se puede acceder a Internet y a los servicios que presta, sin embargo, desde Internet no se puede acceder a los host de la red.

Ahora bien, complementario a los conceptos anteriormente citados, los cuales tienen sus bases en dispositivos físicos principalmente, se encuentran los protocolos de comunicación.

Un Protocol Suite, es un grupo de protocolos interrelacionados necesarios para establecer una comunicación. Son implementados en hardware y software y se encuentran cargados en cada dispositivo de red. Los protocol suite más conocidos son TCP/IP y OSI. Dentro de las funciones de los protocolos se encuentran:

- a. Formato y estructura del mensaje
- b. Control de errores en la red
- c. Control de proceso de transferencia de información entre dispositivos de red.

Dos protocolos utilizados en el desarrollo de este proyecto para realizar la comunicación entre dos dispositivos de red fueron TCP e IP.

**Transport Control Protocol (TCP):** Este protocolo se encuentra en la capa 2 del modelo TCP/IP (capa de transporte). Maneja conversaciones individuales entre un cliente y un servidor. TCP se encarga de dividir los mensajes en pequeños pedazos llamados segmentos para enviarlos a su destino final. Este protocolo garantiza que los datos enviados desde una fuente, lleguen a su destino sin errores y en el mismo orden con que fueron enviados. Además, se encarga de reconocer distintas aplicaciones en un mismo equipo (por ejemplo una PC) por medio del concepto del puerto.

Por ejemplo, la aplicación servidor implementada en Visual C# Express, encargada de recibir los mensajes del cliente utiliza el puerto 1000 para las conexiones entrantes.

**Internet Protocol (IP):** Se encarga de tomar los segmentos de la capa de Transporte y encapsularlo dentro de paquetes, asignándole una apropiada dirección y seleccionando el mejor camino a su destino final. Cada paquete posee una dirección IP destino y fuente, las cuales son usadas por los routers y switches para alcanzar su destino final.

El conjunto dirección IP y puerto forman un socket, el cual es utilizado para enviar información a un host con una determinada IP y corriendo una aplicación identificada por un puerto definido.

El protocolo IP más utilizado es el IPv4. Una IP es una dirección lógica que identifica una interfaz de un dispositivo. Estas direcciones pueden ser proporcionadas estática o dinámicamente (necesario un DHCP).

**Máscara de red:** La máscara de red es utilizada junto a la dirección IP para diferenciar entre los host y la red. Con esta es posible establecer si un dispositivo se encuentra dentro de una red o no. Esto hace posible enviar y dirigir datos hacia otras redes. Si un host envía un dato a otro con determinada dirección IP, el router aplica la máscara de red para determinar si el dispositivo final se encuentra dentro de su red, de no ser así, lo envía al siguiente router hasta alcanzar su destino final.

### Tensión RMS

$$V_{rms} = \sqrt{\frac{1}{T} \int_{-\frac{T}{2}}^{+\frac{T}{2}} v(t)^2 dt} \quad (3.1)$$

Para una señal CA, se cumple que:

$$V_{rms} = \frac{V_p}{\sqrt{2}} [V] \quad (3.2)$$

donde  $V_p$  es el valor pico de la señal.

### Ley de Ohm

$$R = \frac{V}{I} [\Omega] \quad (3.3)$$

Donde  $V$  es la diferencia de potencial en la resistencia  $R$  y  $I$  es la corriente que circula por esa resistencia.

### Ley de tensión de Kirchhoff

La sumatoria de tensiones en un lazo cerrado es igual a cero:

$$\sum_{i=0}^n V_i = 0 \quad (3.4)$$

### **Periodo y frecuencia**

Periodo es el intervalo de tiempo necesario para que una señal se repita nuevamente. Se expresa comúnmente como  $T$ .

La frecuencia es la cantidad de ciclos por segundo en que se repite una señal.

Matemáticamente se expresa como el inverso del periodo:

$$f = \frac{1}{T} [Hz] \quad (3.5)$$

## **Capítulo 4: Procedimiento Metodológico**

En este capítulo se presentan el proceso seguido para llegar a la solución final implementada, luego de pasar por etapas de investigación, rediseño y verificación de funcionamiento. Además, se detalla la forma en que se dio a conocer los resultados de la implementación realizada.

### **4.1 Reconocimiento y definición del problema**

Para lograr llegar a una definición concreta del problema en este proyecto, se utilizó el método de entrevista a ingenieros del Instituto Costarricense de Electricidad del sector DTOM involucrados directamente con problema en cuestión. Preliminarmente estas entrevistas dieron una idea general al problema presente. Sin embargo, también se recurrió a consultas vía correo electrónico para evacuar las dudas eventuales y con ello llegar a la definición final del problema.

Los lineamientos y restricciones planteadas en la solución del problema, los cuales se encuentran enmarcadas en la meta y objetivos de este documento, estuvieron basados en los resultados esperados del producto final por el ICE. Estos resultados se obtuvieron de cada una de las entrevistas y exposiciones realizadas durante el desarrollo del proyecto.

### **4.2 Obtención y análisis de la información**

Los primeros datos relevantes acerca del problema en cuestión se obtuvieron durante el desarrollo de entrevistas a ingenieros. Los cuales proporcionaron una idea del esquema básico del sistema y herramientas disponibles. Complementario a esto, visitas a la Estación Cerro San José y consultas a expertos y personal en la zona (encargados del mantenimiento), dieron una idea más clara del estado del arte.

Durante las visitas a la estación, fue posible distinguir el equipo a realizar el control de alarmas, ubicación en la estación y estructura física. Además, se pudo

conocer las disponibilidades tecnológicas en dicha estación para la implementación de una solución.

### **4.3 Evaluación de las alternativas y síntesis de una solución**

Durante las visitas a la Estación Cerro San José y el desarrollo de entrevistas y consulta a los ingenieros de DTOM del ICE, fue posible ubicar al menos dos alternativas de solución, según lo establecido en los objetivos del proyecto. Las alternativas planteadas, están basadas en la disponibilidad tecnológica y sugerencia de ingenieros en el campo.

La validez de las propuestas de solución, tienen como fundamento los antecedentes de aplicación anteriormente realizados, además de la disponibilidad de implementación con las herramientas tecnológicas según la investigación realizada. A la luz de esto, la solución seleccionada final, se escogió debido a la flexibilidad que presentaba, además de la factibilidad y rapidez de implementación (detallado en el capítulo 5).

### **4.4 Implementación de la solución**

Las primeras etapas del proyecto, fueron dedicadas a investigación en distintos sitios de Internet principalmente (citados en la bibliografía). Esto con el fin de seleccionar las herramientas tecnológicas disponibles para alcanzar los objetivos del proyecto. Apuntando esta selección a la disminución de costos de implementación tanto en software como en hardware.

Luego de la implementación del mismo, se procedió a realizar la verificación de funcionamiento individual de cada uno de los módulos implementados. La verificación se basó en pruebas de laboratorio controladas para poder depurar el sistema hasta llegar a la versión final.

Inicialmente se planteó una solución, la cual fue tomada como base para agregarle pequeñas modificaciones a solicitud de ingenieros del ICE.

Los resultados obtenidos durante el desarrollo del proyecto hasta alcanzar la solución final, según la solución implementada, fueron divulgados en exposiciones de avance del proyecto. Estas reuniones fueron organizadas a por funcionarios del ICE. Otro mecanismo para dar a conocer los resultados de la implementación fue por medio del manual de usuario desarrollado y el cual era parte de los objetivos del proyecto.

#### **4.5 Reevaluación y rediseño**

Una de las alternativas respecto a mejoras en el diseño, podría ser aplicar un sistema de búsqueda en las bases de datos implementada en el PC. En cuanto a la Unidad de Control, es posible ubicar nuevos problemas en la estación los cuales pueden ser solucionados utilizando la tarjeta de desarrollo utilizada en el proyecto, puesto que, esta tarjeta de desarrollo aún dispone puertos, los cuales pueden ser utilizados como convertidores A/D, E/S, entre otras aplicaciones.

Otra de las alternativas de rediseño es la implementación de un control de nivel CA realizado mediante el método de muestreo, esto es, tomar cierto número de muestras durante cada semi-ciclo de la señal CA y calcular de esta forma el valor RMS. Lo cual es posible por medio de un convertidor A/D.

## Capítulo 5: Descripción detallada de la solución

En este capítulo se detalla el diseño del sistema implementado a nivel de hardware y software. Además, se explica los criterios de diseño y la selección de la solución implementada. En este capítulo se presentan los cálculos y procedimientos correspondientes al diseño.

### 5.1 Análisis de soluciones y selección final

#### 5.1.1 Establecimiento de la comunicación

Dada la disponibilidad de equipos e interfaces de comunicación en Cerro San José como en la Base de Tilarán, era posible implementar una solución para realizar la comunicación entre ambos puntos, de dos maneras distintas. A continuación se presentan estas dos opciones de implementación:

##### a. Comunicación vía 64 kbits/s

Existe una conexión directa entre la Base de Tilarán y la Estación Cerro San José, por medio de la cual es posible realizar la comunicación entre estos dos puntos. Se dispone de 30 canales con una capacidad de 64 kbits/s cada uno, teniendo así un máximo de 2 Mbits/s ( $30 \times 64 \text{ kbits/s} = 1920 \text{ kbits}$ ). El tipo de información transmitida a través de cada canal, debe encontrarse en los rangos de frecuencias de voz (300-3400 Hz).

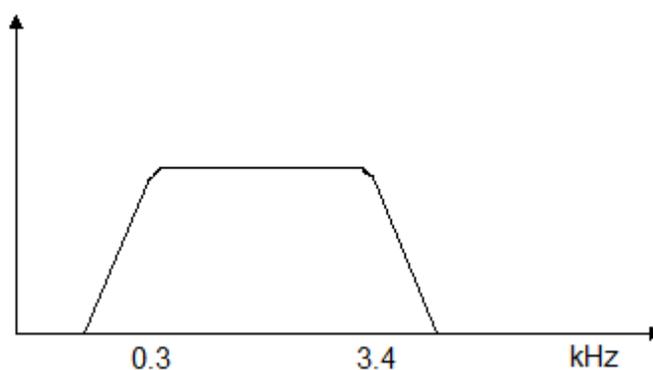
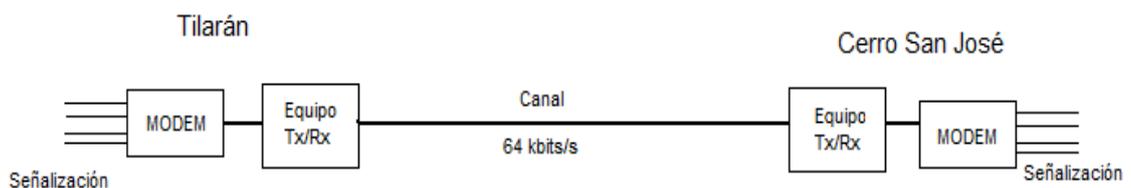


Figura 5.1. Espectro de frecuencia de voz

Cada canal posee cuatro terminales, uno para recepción, uno para transmisión, además dos para señalización, los cuales permiten el establecimiento de la comunicación.

Para la implementación de la comunicación, se requiere equipo adicional (MODEM), el cual permita realizar una interfaz entre las terminales entrada/salida del canal y el equipo de transmisión/recepción, como se muestra a continuación:



**Figura 5.2.** Acople entre canal de 64 kbits/s y equipo de transmisión-recepción.

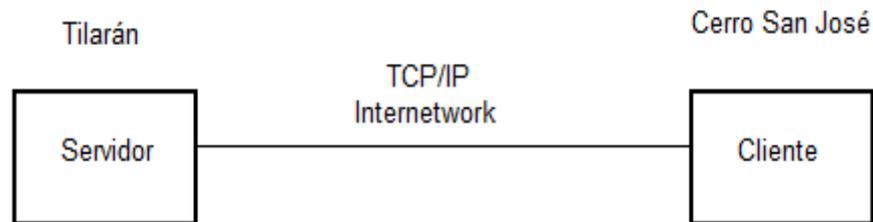
Sin embargo, la implementación de la comunicación utilizando estos canales, hacen al sistema rígido, puesto que, solo es posible realizar un monitoreo del equipo desde la Base de Tilarán. Para poder hacer un monitoreo de igual forma, de la Estación Cerro San José desde otro punto, es necesario otro canal en conexión directa de 64 kbits/s que comunique los puntos en cuestión, además del hardware adicional.

## **b. Comunicación vía TCP/IP**

La otra opción e implementada finalmente, fue por medio de IP (Internetwork Protocol), el cual hace posible establecer un sistema de comunicación cliente-servidor, entre dos lugares (estación a monitorear y la base de control), a través de la red del ICE (Intranet), la cual es un dominio privado.

IP hace el sistema más flexible en comparación con el mencionado anteriormente, puesto que, ahora lo que se requiere cambiar o configurar son direcciones IP únicamente para acceder el equipo remoto (estación remota como servidor) o por

medio del sistema de control en la estación remota acceder a la base de control (estación remota como cliente).



**Figura 5.3.** Sistema de comunicación cliente-servidor

Respecto a la selección del protocolo de transmisión de paquetes de información entre el cliente y el servidor. Se seleccionó TCP, puesto que proporciona una mayor confiabilidad en la transmisión de información que otros protocolos como UDP, el cual no posee un sistema de chequeo de errores ni recepción. No obstante, TCP verifica si el paquete llegó o no a su destino final, de esta forma asegurándonos el reporte de errores en la estación remota.

### **5.1.2 Selección de lenguajes de programación**

El establecimiento de un sistema de comunicación cliente-servidor utilizando TCP/IP en este proyecto, requiere la implementación de una interfaz gráfica en un software de alto nivel en la Base de Tilarán, para poder recibir y registrar en una base de datos la información proveniente de la estación remota (CSJ).

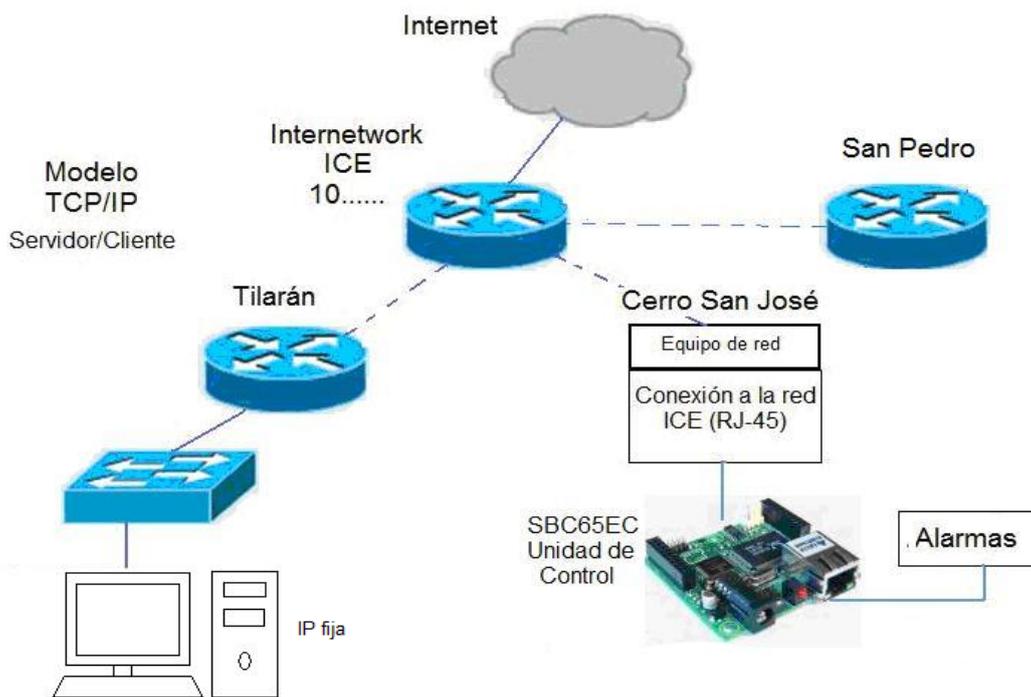
La selección adecuada de un lenguaje de programación, está sujeta, a si este lenguaje permite el desarrollo de aplicaciones con sockets (dirección IP y puerto, ejemplo: 192.152.25.112/546).

Dentro de las opciones disponibles, se utilizó Visual C# Express Edition 2005 de Microsoft para implementar la interfaz gráfica en el PC. De igual forma, se utilizó SQL Server Express Edition 2005 como complemento para la generación de una base de datos donde registrar los eventos reportados.

De igual forma que en el PC, se investigó de forma tal, para implementar un sistema económico pero eficiente en la unidad de control (microcontrolador de Microchip 18F6627) del sistema remoto. Para ello se utilizaron las herramientas: MPLAB IDE y el compilador MPLAB C-18 versión estudiante. Con estas herramientas es posible el manejo de sockets gracias a las bibliotecas libres proporcionadas por Microchip (TCP/IP Stack) y por tanto, es posible el acceso a una red LAN o WAN como cliente o como servidor.

### 5.1.3 Implementación y diseño

En la figura siguiente, se muestra el diagrama del sistema general implementado en el proyecto, el cual se explica a continuación.



**Figura 5.4.** Diagrama del sistema cliente/servidor implementado en la red ICE.

Se implementó un sistema cliente/servidor en el dominio de redes privadas del ICE mediante el modelo TCP/IP y el estándar Ethernet, en el cual se estableció el servidor en el Centro de Atención de Equipos ubicada en Tilarán y el sistema en la Estación Cerro San José como cliente.

Utilizando la herramienta Visual C# Express Edition se creó una aplicación que hiciera actuar a un PC común conectado a la red como un servidor, esto aprovechando la disponibilidad de direcciones IP fijas en la red (ICE). Ahora bien, de no contar con una dirección IP fija en el servidor, cada vez que este es reconectado adquiere una nueva dirección IP (utilizando un servidor DHCP), lo que implica que la dirección destino al que el cliente envía un mensaje debe ser reconfigurado con la misma dirección que el servidor. Esta aplicación al recibir un mensaje del cliente muestra en pantalla el evento dado, identificando además, el tipo de prioridad del evento (si se debe al fallo en un equipo), de forma tal que le indique al usuario el nivel de urgencia en la solución del problema. Luego de recibido el evento, este queda almacenado en una base de datos.

La implementación del cliente se hizo utilizando la tarjeta de desarrollo SBC65EC de Modtronix, la cual actúa como un host. Ésta utiliza microcontroladores Microchip (para esta aplicación el 18F6627), el cual maneja el puerto RJ-45 y por tanto, permite conexión a una red. Esta tarjeta es la unidad de control (UC) del sistema en la Estación Cerro San José y es la encargada de monitorear los equipos para determinar en cual de ellos se dio una alarma y enviarlo al servidor ubicado en Tilarán.

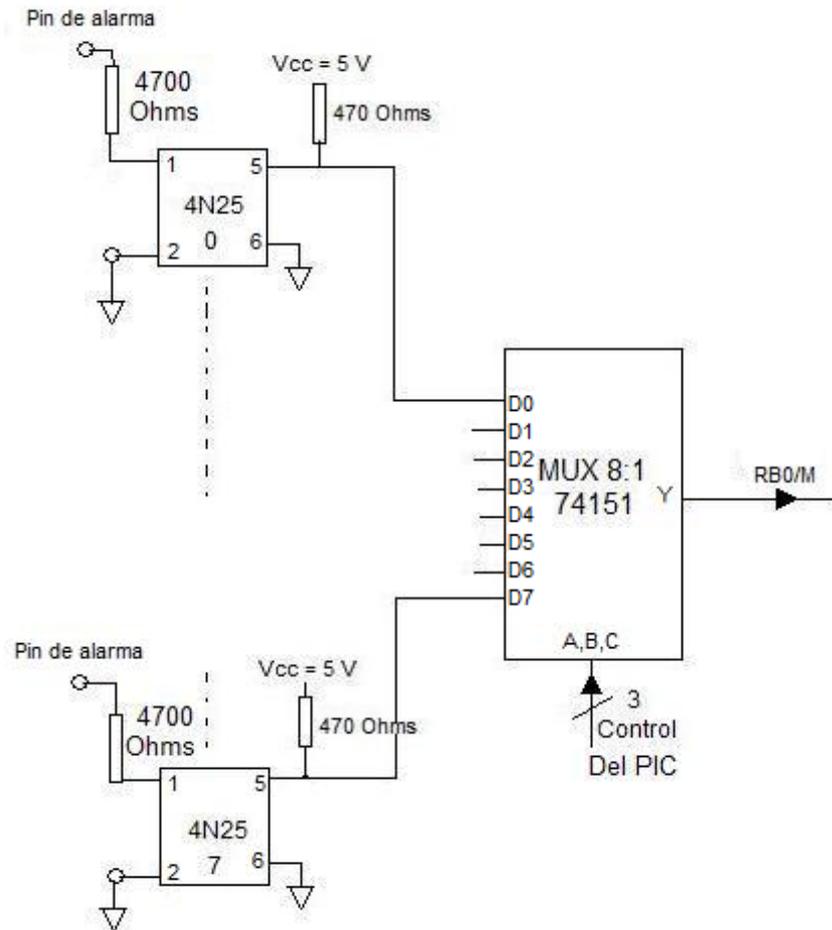
También, se implementaron dos etapas adicionales. Una etapa de acople entre las tensiones de las alarmas de los equipos (+48 V) y la tarjeta utilizada (tecnología TTL) y otra etapa para detectar cambios en el nivel CA mayores o iguales al 10% del valor RMS de la red de alimentación.

## **5.2 Descripción del hardware**

### **5.2.1 Acople de tensiones**

Los equipos en la Estación Cerro San José normalmente tienen las alarmas a un nivel de tensión de 0 V. Cuando un equipo falla, ponen el punto de alarma a una diferencia de potencial aproximado de +48 V respecto a su referencia. Por tanto, no es posible realizar un acople directo entre la UC y las terminales de alarmas, ya que la UC opera a niveles de tensión máximas de 5.5 V. Por ello, se

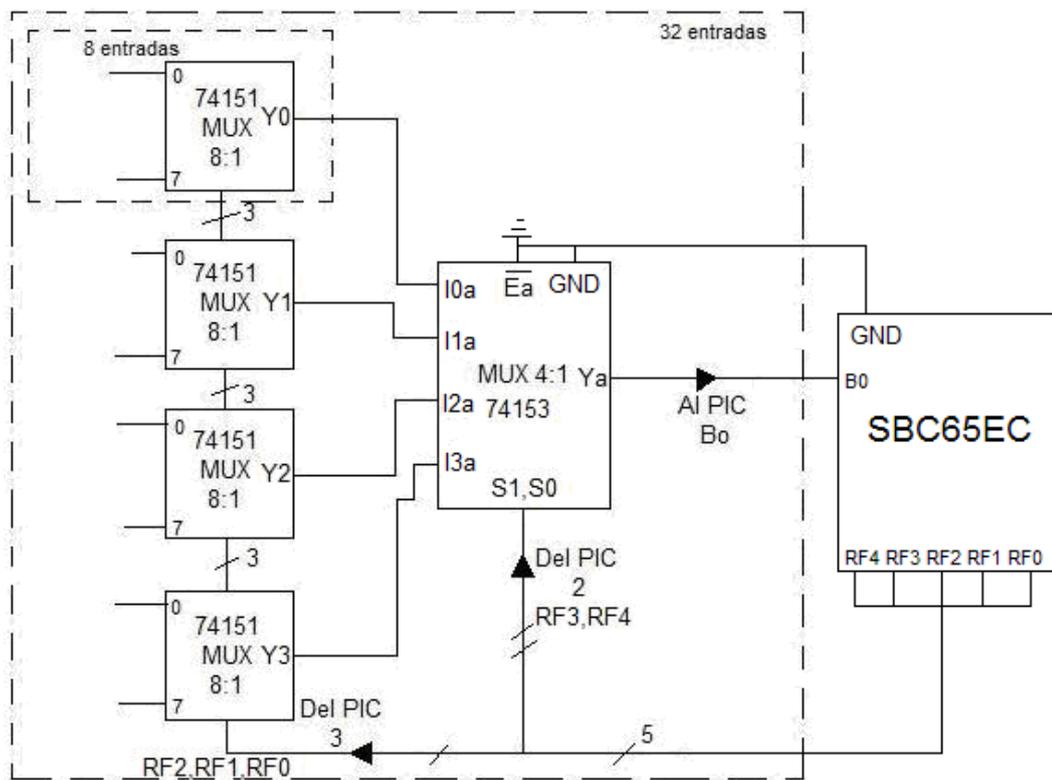
implementó una etapa de potencia para realizar este acople, en el cual se hizo un aislamiento de referencias por medio de opto acopladores como se muestra a continuación:



**Figura 5.5.** Etapa de acople entre las tensiones de alarmas y la UC

Para ello se utilizó, el opto acoplador 4N25 o un equivalente a éste. En la configuración del circuito anterior, los pines de entrada del multiplexor 74151 se encuentran normalmente en alto, puesto que el LED del 4N25 se encuentra apagado y por tanto su transistor interno está en modo abierto. Cuando se da una alarma, el LED entra en conducción y por tanto satura al transistor, poniendo con ello a 0 V la entrada correspondiente del MUX (nivel normal de alarma). Cuando el microcontrolador selecciona dicha entrada del MUX detecta el nivel de alarma y realiza la conexión con el servidor remoto para reportar el evento.

Con el objetivo de generar un sistema flexible, el circuito de la figura anterior se montó en una placa independiente con sus respectivas entradas y salidas. Esta placa tiene la capacidad de acoplar 8 equipos en la estación (MUX 8:1). Sin embargo, es posible adherir nuevas placas idénticas agregando otros multiplexores y señales de control provenientes del PIC, para expandir el número de equipos a monitorear como se muestra en la figura 5.6.



**Figura 5.6.** Placa de acople para 32 entradas de alarmas. Cuatro módulos de 8 entradas cada uno.

En este circuito cada equipo posee un respectivo código de identificación según la posición en el módulo que se encuentre, tomando en cuenta que esta posición ya conocida, es utilizada en el programa del PC y de la Unidad de Control (UC) para saber cual equipo falló si se da una alarma (se retoma con mayor detalle en la explicación de la etapa de software). Cada módulo es seleccionado de manera independiente por medio de los pines RF3 y RF4 (controlados por el PIC).

**Tabla 5.1.** Selección de módulos mediante polling

S1/RF4	S0/RF3	Módulo
0	0	I0a
0	1	I1a
1	0	I2a
1	1	I3a

La selección de cada una de las entradas de los multiplexores es hecha también por el microcontrolador, mediante los puertos RF2, RF1 y RF0, los cuales se conectan a cada uno de los módulos.

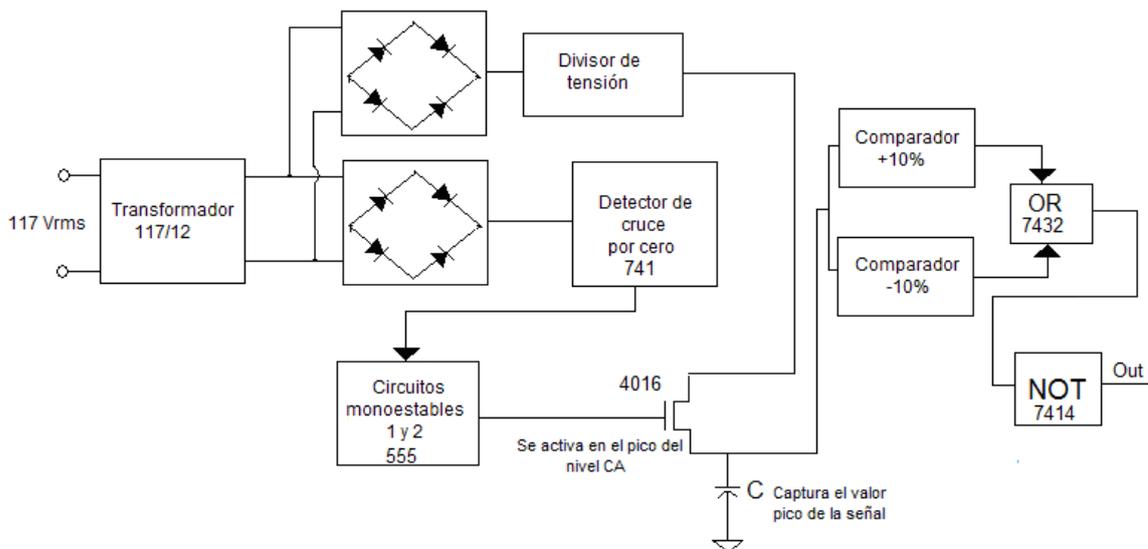
La retardo de muestreo entre cada una de las alarmas es de 1 ms y la latencia depende de la cantidad de módulos implementados en la UC. Si se realiza el control de 25 tipos de alarmas o eventos, la latencia de muestreo es de 25 ms si durante ese muestreo no se da ninguna alarma y es en el peor de los casos de aproximadamente de 100 ms si se da una alarma durante la ejecución.

También, es posible utilizar solamente un módulo básico de 8 entradas. En este caso, la salida del multiplexor 74151 debe conectarse a la entrada RB0 del microcontrolador, la cual conmuta cada 1 ms hasta ver la salida de cada una de las alarmas (polling), teniendo cada equipo una latencia de chequeo de 8 ms en caso de no haber alarmas en los equipos.

Si bien es cierto utilizando el método del muestreo no es posible obtener un reporte instantáneo de una alarma, la latencia de tiempo de muestreo es despreciable a nivel de percepción humana. Además, la utilización del hardware adicional para realizar el polling (uso de los multiplexores) presenta beneficios en cuanto a expansión del sistema, puesto que, el número de puertos utilizados del microcontrolador no supera el 25 % de los disponibles.

## 5.2.2 Detección de cambios en el nivel CA

En el diagrama de la figura 5.7 se muestra el circuito implementado para detectar cambios mayores o iguales a 10% del valor rms de la red CA de alimentación. Este módulo tiene como entrada 117 Vrms (red de alimentación).



**Figura 5.7.** Diagrama de bloques del circuito implementado para detectar cambios de +/- 10% en el nivel CA de la señal de la red de alimentación.

Para detectar cambios en el nivel CA, se diseñó un circuito que tomara una muestra a escala del valor pico de la señal CA de la red, para comparar este valor pico (pico actual) con los posibles valores picos de la señal CA ( $\pm 10\%$ ), a partir de los cuales se genera una bandera de error o fallo, el cual se representa por medio de una diferencia de potencial de 0 V. El diseño de este circuito está basado en la igualdad:

$$V_{rms} = \frac{V_p}{\sqrt{2}} [V]; \quad \text{para una onda senoidal.}$$

De la ecuación anterior se observa que  $V_{rms}$  es directamente proporcional a  $V_p$  y por tanto, teniendo  $V_p$  se tiene también  $V_{rms}$ .

Puesto que, no es posible trabajar directamente con la tensión nominal de la red de alimentación para implementar un circuito de control con tecnología TTL y amplificadores operacionales, se utilizó un transformador de  $117V/24V$  con devanado central ( $\pm 12V$ ) para reducir la tensión a un nivel manejable sin alterar la señal CA. La salida de este transformador, es la entrada de dos rectificadores de onda completa lo cual implica que la nueva señal tiene el doble de la frecuencia que la original.

$$f_{señal} = 60[Hz]$$

Luego del rectificador:

$$f' = 120[Hz]$$

La salida de uno de los rectificadores es la entrada de un detector de cruce por cero, este es el encargado de sincronizar la señal de la red con el circuito de control. Este detector de cruce por cero, dispara un monoestable el cual mantiene su tiempo en alto aproximadamente hasta alcanzar el valor pico de la señal CA:

$$T_{señal} = \frac{1}{60Hz} = 16.66[ms]$$

$$\Rightarrow T'_{señal} = \frac{1}{120Hz} = 8.333[ms]$$

$$\Rightarrow V_{pico} = \frac{8.333ms}{2} = 4.166[ms] \quad ; \text{ se da valor pico de la señal CA.}$$

Durante la transición de alto a bajo (flaco negativo) de la salida del monoestable, este dispara un segundo monoestable que es el encargado de obtener el valor pico de la señal CA, disparando el conmutador CMOS 4016B, el cual permite que el capacitor en su salida almacene el valor pico y sea comparado con los valores máximos permitidos antes de activar una alarma.

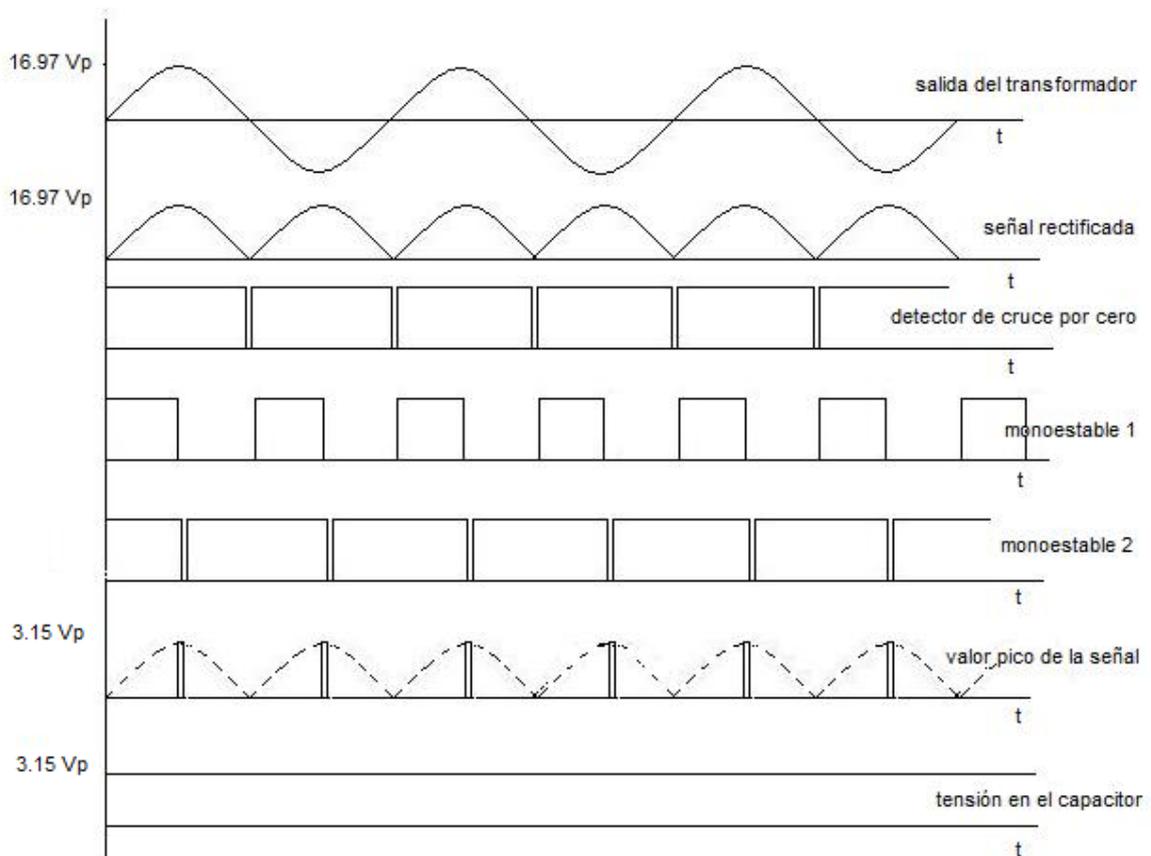
Tiempo en alto del monoestable 1:

$$t_{alto-1} = 1.1(900\Omega)(4.7\mu F) = 4.65ms$$

Tiempo en alto del monoestable 2:

$$t_{alto-2} = 1.1(64K\Omega)(0.1\mu F) = 7.37ms$$

En la figura siguiente se muestra el diagrama de tiempos del circuito:



**Figura 5.8.** Diagrama de tiempos del circuito implementado para detectar cambios de +/- 10% en el nivel CA de la señal de la red de alimentación.

La tensión pico normal en la salida del transformador es de 16.97 V como lo muestra la figura anterior. No obstante, es necesario reducir esta tensión a un valor de forma tal que pueda ser comparado con valores de tensión entre 0 y 3.5 voltios, esto con el fin de tener una referencia fiable y constante. El valor máximo

de 3.5 V es seleccionado, puesto que, es el valor máximo de entrada del IC 4016 utilizado para tensiones de alimentación de 5 V.

La salida del otro rectificador es conectado a un divisor de tensión para obtener una tensión pico de 3.15 V. Este valor se obtuvo, tomando como límite superior (+10%) 3.5 V. A continuación se muestran los cálculos realizados:

$$V_{rms (+10\%)} = V_{rms_N} + 0.1V_{rms_N} [V]$$

$$V_{rms (-10\%)} = V_{rms_N} - 0.1V_{rms_N} [V]$$

$$\text{Se quiere que: } V_{rms (+10\%)} = \frac{3.5}{\sqrt{2}} [V]$$

$$\Rightarrow \frac{3.5}{\sqrt{2}} [V] = V_{rms_N} + 0.1V_{rms_N} [V] = 1.1V_{rms_N}$$

$$\Rightarrow V_{rms_N} = \frac{3.5}{1.1\sqrt{2}} = 2.25 [V_{rms}]$$

$$\Rightarrow V_{p_N} = 2.25\sqrt{2} = 3.15 [V]$$

$$\text{Valor pico del máximo: } V_{p_{+10\%}} = 3.5 [V]$$

De igual forma calculamos el otro extremo de tensión rms:

$$V_{rms (-10\%)} = V_{rms_N} - 0.1V_{rms_N} [V]$$

$$V_{rms (-10\%)} = 2.25 - 0.1(2.25) = 2.02 [V]$$

$$\Rightarrow V_{p_{-10\%}} = 2.83 [V]$$

Calculo de valores de resistencias para el divisor de tensión a la salida del rectificador (atenuador de valor pico):

$$V_x = \frac{(12\sqrt{2})R_{out}}{R + R_{out}} = 3.15 [V]$$

Si  $R = 1K\Omega$ :

$$\Rightarrow R_{out} = 228\Omega$$

Calculo de valores de resistencias para el divisor de tensión a la entrada de los comparadores:

$$V_{(-10\%)} = 2.86[V]$$

$$V_{(-10\%)} = \frac{(5)R_{out}}{R_{out} + R}[V] = 2.83[V]$$

Si  $R_{out} = 1K\Omega$ :

$$\Rightarrow R = 770\Omega$$

$$V_{(+10\%)} = 3.5[V]$$

$$V_{(+10\%)} = \frac{(5)R_{out}}{R_{out} + R}[V] = 3.5[V]$$

Si  $R_{out} = 2.3K\Omega$ :

$$\Rightarrow R = 1k\Omega$$

Por tanto se tiene:

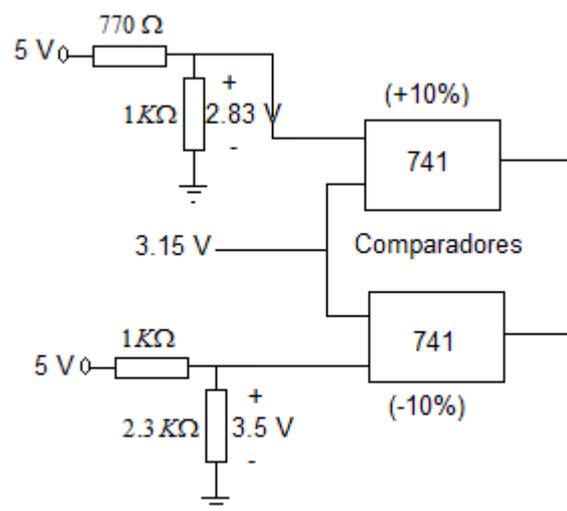


Figura 5.9. Diagrama de circuitos comparadores.

### 5.3 Descripción del software

La descripción del software se encuentra dividida en dos secciones, una a nivel de la Unidad de Control (cliente: Microchip C18) y la otra a nivel del PC (servidor: Visual C# Express), las cuales se detallan a continuación.

#### 5.3.1 Descripción del programa en la Unidad de Control

Para el desarrollo del programa a nivel de la UC, se utilizó el lenguaje de programación C18 de Microchip (versión estudiante).

El programa implementado se encuentra dividido en tres secciones, los cuales son invocados en el programa:

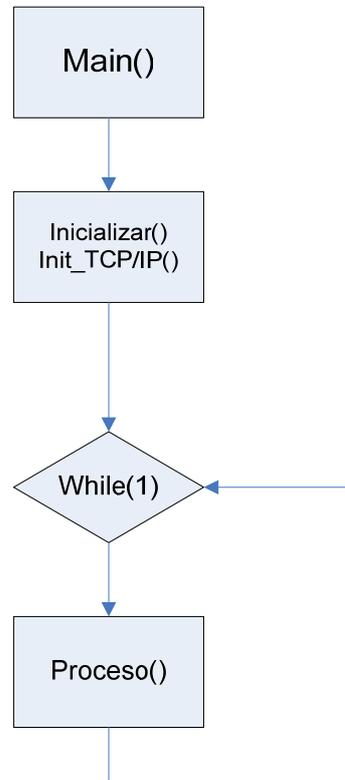
- a. Proceso ();
- b. Inicializar ();
- c. Main();

#### **Main():**

En el diagrama de flujo de la figura 5.10, se muestra la estructura del programa principal. Al ejecutarse este programa, dentro del método main() se ejecutan tres procesos:

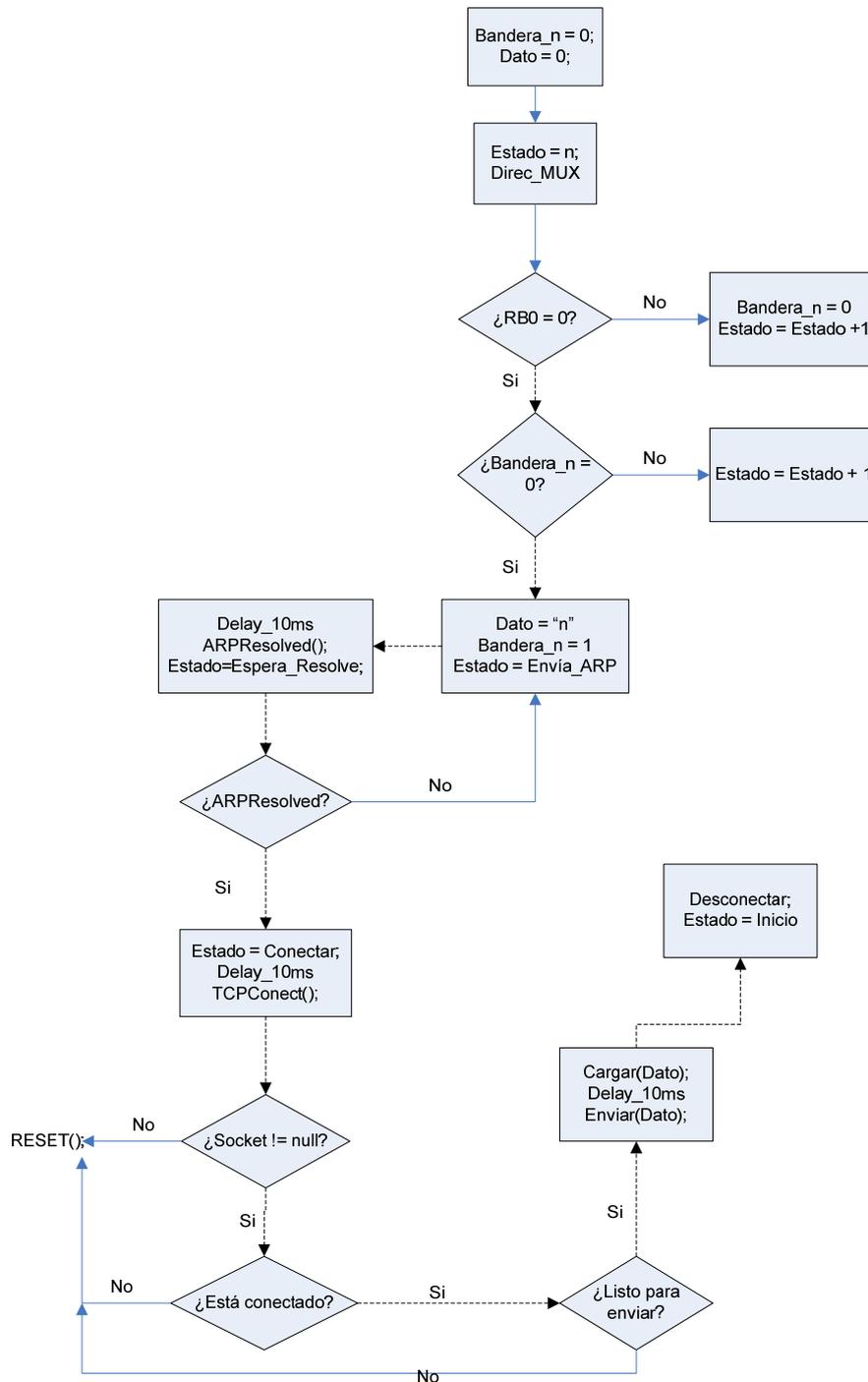
1. Inicializar(): Se inicializa el microcontrolador, y por tanto, de igual manera se configura la tarjeta de desarrollo. Se definen los puertos de entrada y salida. Además, se inicializan: las direcciones IP, máscara de red, dirección MAC y puerta de enlace predeterminada. También, dentro de la estructura main() se el socket (dirección IP y puerto) del servidor remoto al cual el cliente se conecta.

2. `Init_TCP/IP()`: Se inicializa los componentes de las bibliotecas de Microchip para la utilización del modelo de redes TCP/IP.



**Figura 5.10.** Diagrama de flujo del programa principal

3. `Proceso()`: Este método es el encargado de realizar el monitoreo de cada una de las alarmas en el sistema, además, de realizar la conexión con el servidor al darse una alarma y controlar el proceso de envío de información. En el siguiente diagrama de flujo se detalla el algoritmo implementado:



**Figura 5.11.** Algoritmo implementado para el monitoreo de alarmas y envío de datos al servidor

En este método se inicializan cada una de las variables de control a utilizar. Se declaran variables *bandera*, una para cada equipo o evento en monitoreo. Al darse una alarma en un equipo (puerto RBO en bajo) se corrobora si la bandera se encuentra en estado bajo (0). Este caso indica que el servidor no ha sido

informado de la alarma existente en la estación. Si el estado de la bandera es alto (1) aunque se encuentre la alarma del equipo activa ( $RB0 = 0$ ), implica que el servidor ya está enterado de la alarma. Cada bandera es puesta en bajo únicamente cuando cada equipo al ser seleccionado presenta un estado de  $RB0$  alto ( $RB0 = 1$ ).

Cada evento monitoreado es seleccionado por el microcontrolador, verificando el nivel alto o bajo del puerto  $RB0$ . Si  $RB0$  es alto el microcontrolador selecciona el siguiente equipo hasta verificar el estado de cada uno de los equipos. Este es un ciclo que se repite hasta que se da una alarma.

Solo en el caso que se de un evento en la estación, la tarjeta de desarrollo realiza la conexión con el servidor. La línea punteada del diagrama de flujo de la figura 5.11 muestra el proceso seguido por el microcontrolador para establecer comunicación con el PC remoto e informar del evento específico. Para poder hacer esto, la variable *Dato* se actualiza (un código por evento) cada vez que se selecciona un equipo para verificar su estado de alarma. En la tabla 5.2 se muestra el código correspondiente a cada evento en la estación, código idéntico utilizado en el programa a nivel del PC.

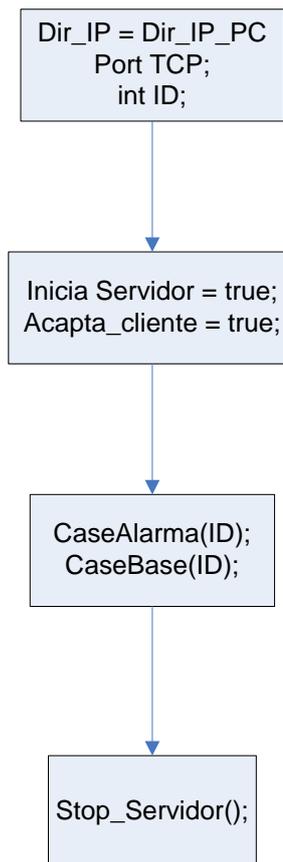
**Tabla 5.2.** Identificación del equipo en la Estación Cerro San José

Identificador	Equipo de monitoreo
"1"	Radio Enlace OKI 5 Hda. Tenorio
"2"	Microonda Sat 2/8 Bebedero
"3"	Microonda Telettra 2/34, 2+1Jicaral
"4"	Microonda HT6 Internacional
"5"	Deshidratador Nec 1
"6"	Deshidratador Nec 2
"7"	Deshidratador Nec 3
"8"	Microonda Nec 2/34 Ruta Guayabo
"9"	Microonda Nec 155 Mb 3+1 Ruta Liberia
"A"	Microonda Nec 155 Mb 3+0 Ruta Liberia
"B"	Microonda Nec 155 Mb 3+1 Ruta Guaria
"C"	Microonda Nec 155 Mb 4+0 Ruta Puntarenas
"D"	Microonda Nec 155 Mb 3+1 Ruta Puntarenas
"E"	Microonda Nec 155 Mb 2+1 Ruta Tilarán
"F"	Microonda Nec 155 Mb 1+1 Ruta Tierras Morenas
"G"	Mux Alcatel 2/34 # 2 Jicaral
"H"	Mux Nec 2/34 # 9 Guayabo
"I"	Mux Nec 2/34 # 8 Jicaral
"J"	Nec Pasolink STM 0 # 1, 1+1 Pilas de Canjel
"K"	Nec Pasolink STM 0 # 2, 1+1 Puerto Humo
"L"	DXR 100 – Los Angeles
"M"	DXR 100 – Pijije
"N"	TNE Alcatel cuarta generación11 a 18
"O"	Nivel CA
"P"	Puerta de acceso

### 5.3.2 Descripción del programa en el PC

A nivel del PC son básicamente tres métodos los que cumplen con las funciones principales de la aplicación.

#### 1. Servidor():



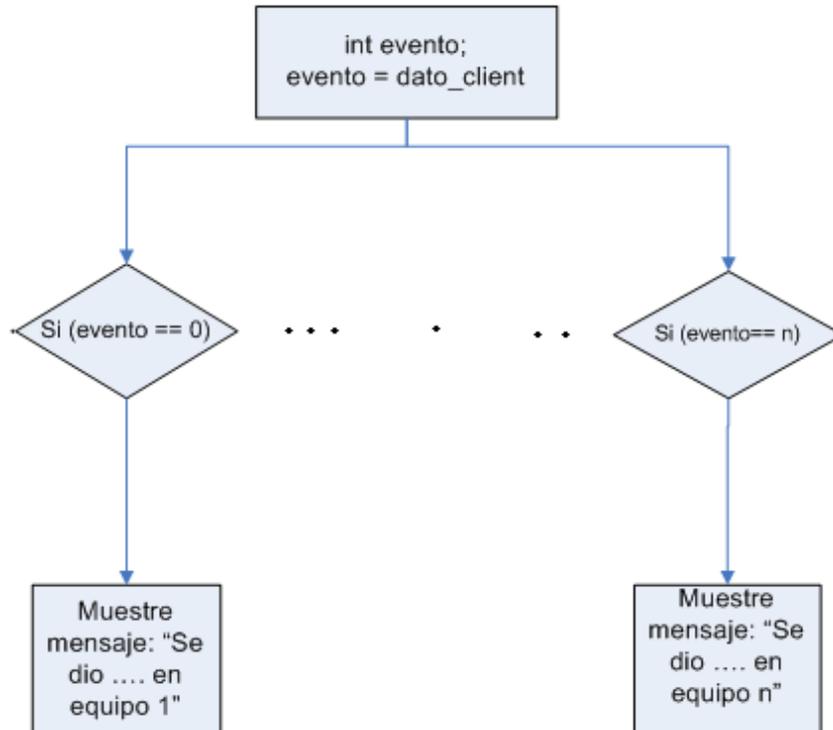
**Figura 5.12.** Diagrama de flujo del método Servidor(), implementado a nivel del PC

Este método es el encargado de aceptar las conexiones del cliente o tarjeta de desarrollo remota, con destino al puerto 1000 y la dirección IP de la PC en la cual se corre la aplicación. Esto es posible por medio de la utilización de la clase Sockets de Visual C# Express Edition. Al iniciar la aplicación, el sistema entra en espera de conexiones. Luego de una conexión entrante el sistema servidor se desconecta automáticamente, siendo el usuario el encargado de volver a correr el método *Servidor* si hace lo desea.

Como se puede ver en el diagrama de la figura 5.13, durante la ejecución del método *Servidor()* se invocan los métodos *CaseAlarma()* y *CaseBase()* los cuales se explican a continuación.

## 2. CaseAlarma():

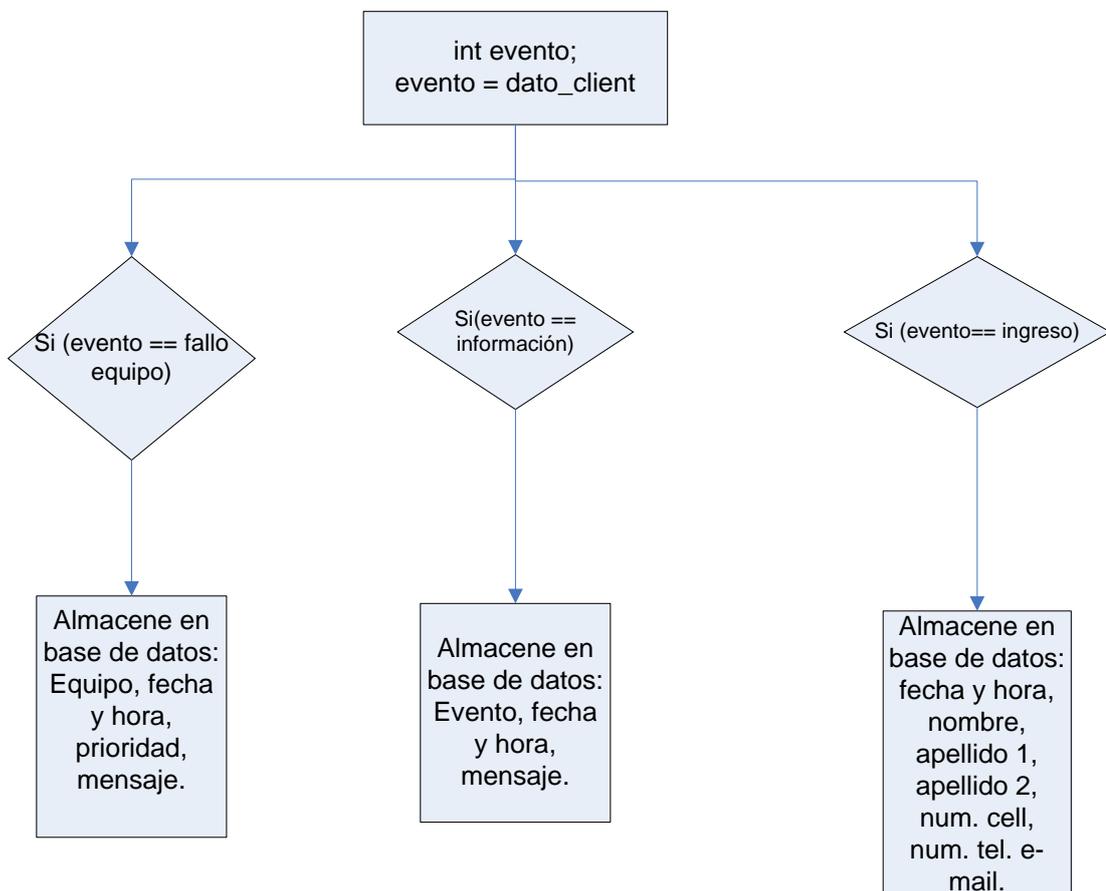
Este método tiene como parámetro de entrada un identificador, el cual es enviado por la tarjeta de desarrollo según sea el equipo fallido o evento dado. Este método recibe el parámetro y lo codifica, mostrando en la pantalla del PC en el que se corre la aplicación un mensaje del evento ocurrido en la Estación Cerro San José según el código recibido (Tabla 5.2)



**Figura 5.13.** Diagrama de flujo del método *CaseAlarma()*, implementado a nivel del PC

### 3. CaseBase():

Este método al igual que el método anterior recibe un parámetro de entrada, asignado por el cliente. De igual manera es codificado según la información enviada por la tarjeta de desarrollo. La función de este método es registrar en la base de datos, implementada en SQL Server el evento según su tipo. Teniendo tres categorías posibles de eventos: Alarma en equipo, evento de información y evento de ingreso de una persona. Así pues, según su categoría, será registrado en una de las tres bases de datos como se muestra en el diagrama siguiente.



**Figura 5.14.** Diagrama de flujo del método CaseBase(), implementado a nivel del PC

Luego de recibir un reporte de evento desde la Estación Cerro San José, éste queda registrado de forma inmediata en la base de datos, pudiendo luego ser corroborado por parte del usuario de aplicación.

En la base de datos *Alarma en equipo* se registra: equipo que presenta la alarma, fecha y hora que se dio la alarma, prioridad de alarma y un mensaje de información que le indique al usuario del software como proceder ante la alarma.

En la base de datos *Eventos de información* se registra el tipo de evento, fecha y hora en que se dio el evento y un mensaje de información para el usuario.

Por ultimo, la base de datos *Ingreso de personal* registra datos personales acerca de quien ingresa a la estación. No obstante, aunque esta opción se encuentra implementada en software a nivel del PC, aún no se encuentra implementada a nivel de la UC.

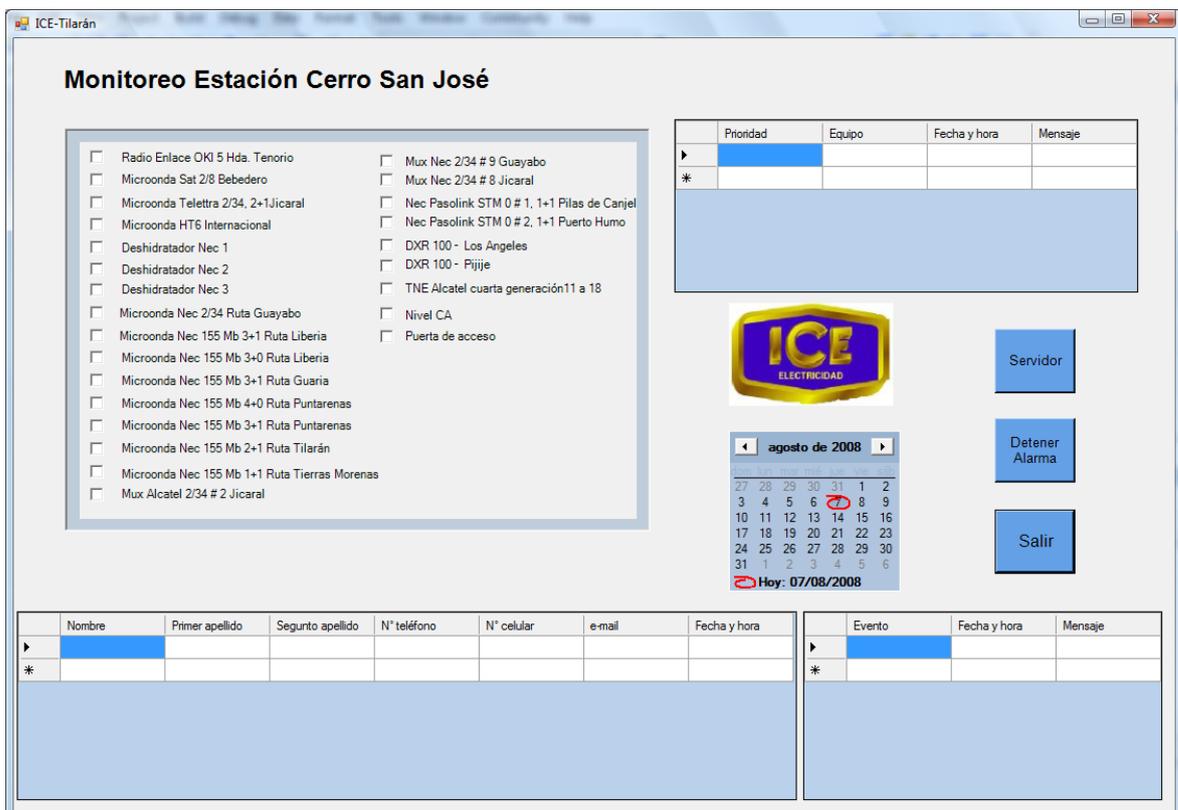
## Capítulo 6: Análisis de Resultados

En esta sección se presentan los resultados experimentales obtenidos de la realización de pruebas a cada una de las etapas diseñadas en el proyecto. Los resultados en este capítulo se encuentran divididos en dos secciones: A nivel de software y a nivel de hardware. Cada una de las pruebas y resultados obtenidos se explican y analizan a continuación. Además, los resultados obtenidos son comparados con los objetivos planteados en el proyecto.

### 6.1 Resultados Experimentales y Análisis

#### 6.1.1 A nivel de software:

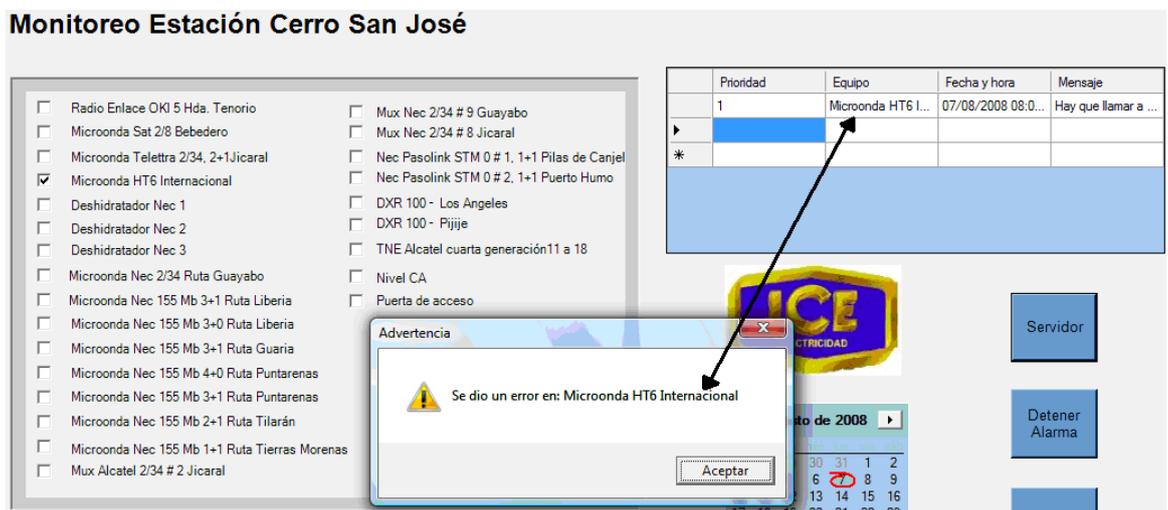
Los resultados aquí presentados, son los realizados a nivel de la interfaz en el PC y de la Unidad de Control en conjunto. Puesto que, se requiere de ambos host para la implementación de la comunicación cliente/servidor del sistema.



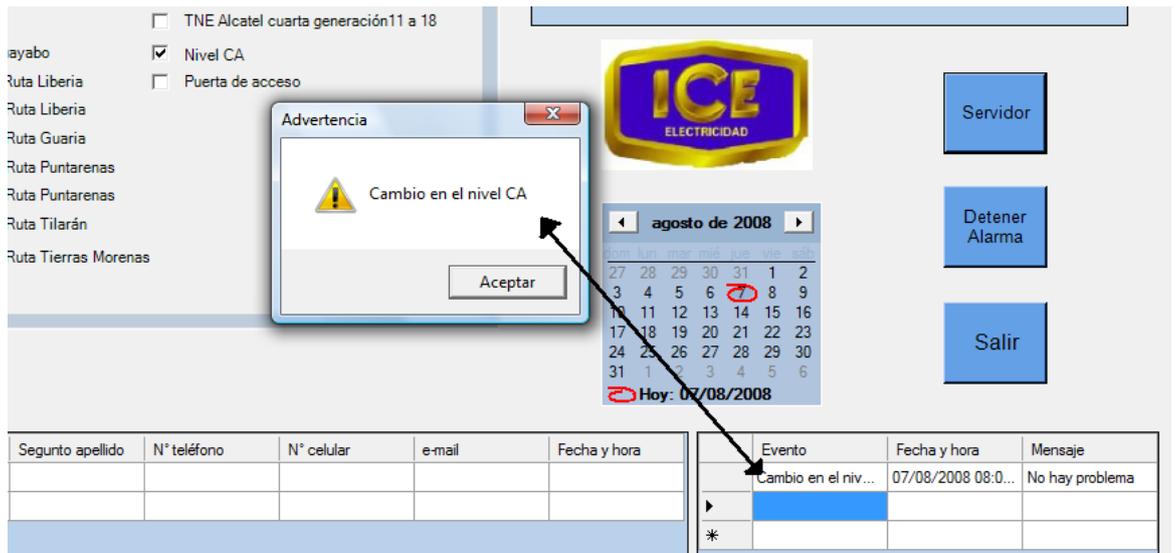
**Figura 6.1.** Interfaz gráfica implementada en el PC, utilizando el lenguaje Visual C# Express Edition

En la figura 6.1 se puede apreciar la interfaz gráfica implementada en Visual C# Express, en la cual se conjuntan la aplicación del sistema servidor y la base de datos. Esta interfaz tiene como función principal indicar al usuario de la aplicación que se dio un evento en la Estación Cerro San José, además, de darle la posibilidad de verificar dicho evento al ejecutar la aplicación (referido a los objetivos específicos 3 y 4).

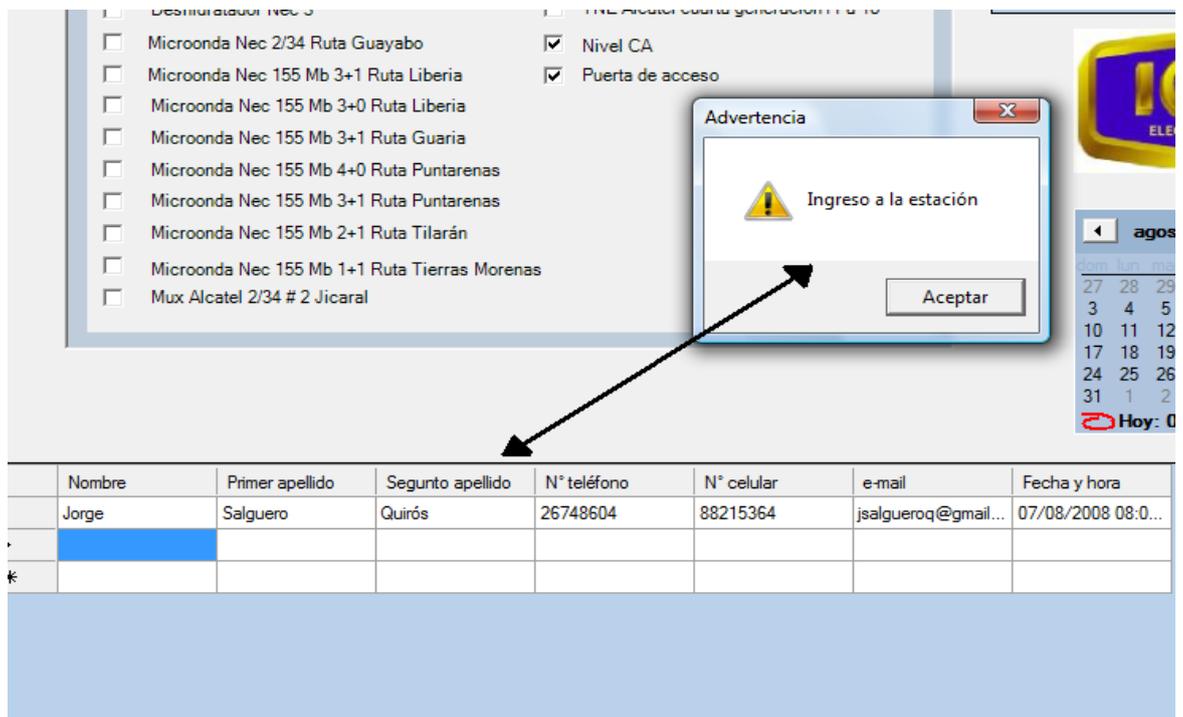
Para la visualización del reporte de alarmas en la interfaz gráfica se simularon alarmas remotamente utilizando la Unidad de Control, la cual es la encargada de la comunicación con el servidor.



**Figura 6.2.** Mensaje de error en equipo remoto de la Estación Cerro San José



**Figura 6.3.** Mensaje de evento en la Estación Cerro San José



**Figura 6.4.** Mensaje de ingreso de persona a la Estación Cerro San José

En las figuras 6.2, 6.3, y 6.4 se muestran los mensajes presentados por la aplicación luego de la simulación de un evento remoto. Para esto, se hizo la conexión del recolector de alarmas a la tarjeta de desarrollo y se puso a funcionar la aplicación en el PC (servidor esperando conexión del cliente). Al poner en bajo cada una de las entradas del recolector en la que se deseaba una alarma

(referido al objetivo específico 2), esta fue reportada de forma casi inmediata al servidor, almacenándola según el tipo de evento. Sin embargo, el sistema reporta eventos de manera inmediato si la aplicación servidor se encuentra activo. De no ser así, si se da una alarma la tarjeta de desarrollo (cliente) tratará de conectarse al servidor mientras la alarma se mantenga, logrando tal objetivo hasta que el servidor entra en funcionamiento (en el apéndice A.2 se detallan estos aspectos).

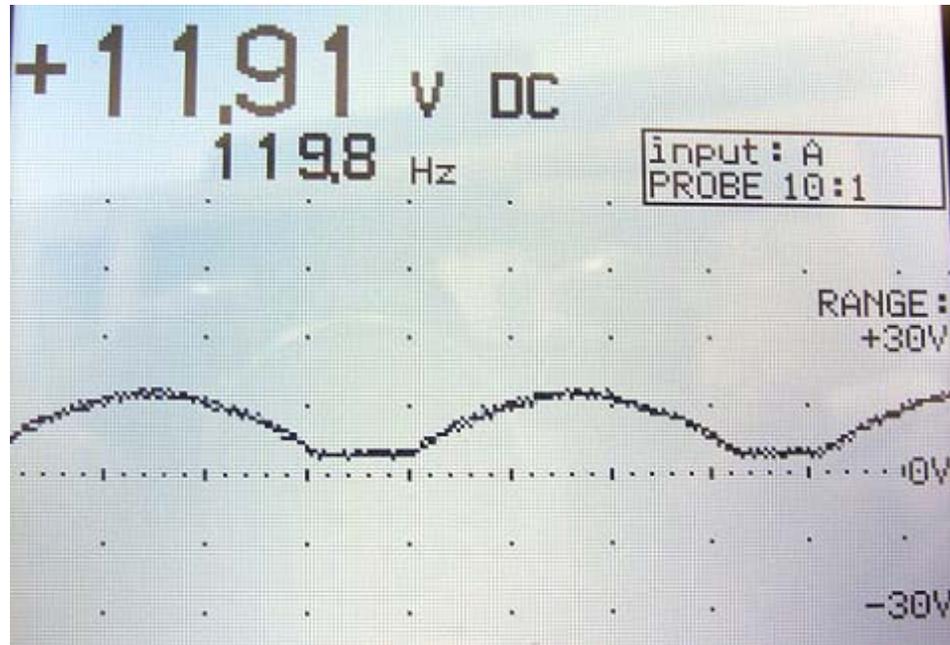
Los tiempos de retardo imperceptibles para el hombre debido a las frecuencias de operación de la tarjeta de desarrollo (40 MHz) y la tasa de transferencia de datos (10 Mb/s), entre el momento que se da una alarma y el momento en que es reportado en la interfaz gráfica, se debe al uso de polling y no de interrupciones (mayor eficiencia a nivel de microprocesadores) para realizar el control de alarmas. Como se explicó en el capítulo 5, la latencia de un punto de alarma, si no se reporta ningún evento es de 25 ms y es de aproximadamente 100 ms en el peor de los casos si se da un evento.

$$T_{\text{Latencia}} = 65 \text{ ms (conexión y envío de datos)} + 25 \text{ ms (muestreo de equipos)} = 100 \text{ ms.}$$

Durante el proceso de conexión entre el cliente y el servidor, se utilizan retardos de manera intencional, con el objetivo de estabilizar el sistema luego de cada etapa de conexión. No obstante, toma importancia el retardo establecido entre el momento en el que se carga el buffer de salida de datos de la tarjeta de desarrollo y el instante en que son enviados a su destino final. Puesto que, como lo indica la guía TCP/IP de Microchip ([5] AN833), es necesario establecer un tiempo de espera entre los pasos mencionados.

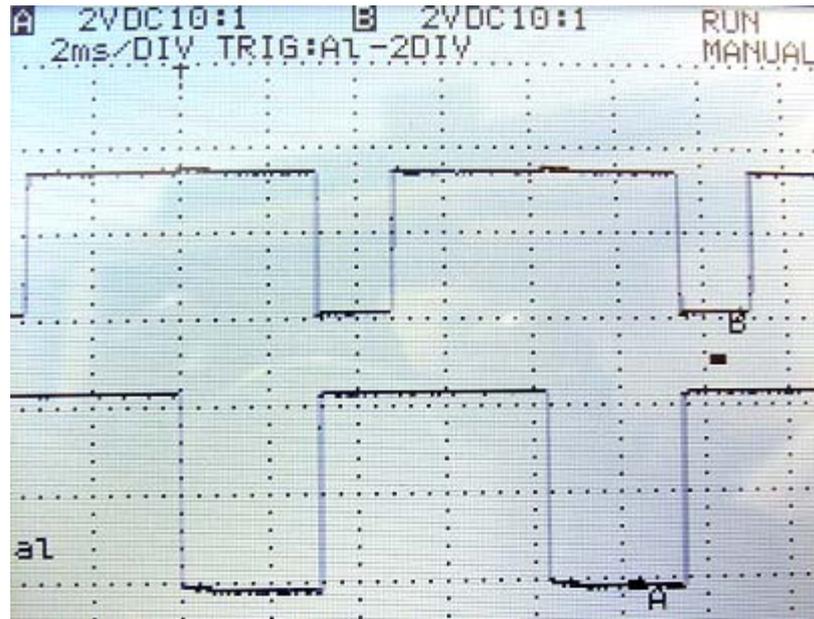
### 6.1.2 A nivel de hardware:

En esta sección se presentan los resultados obtenidos durante las pruebas a los circuitos complementarios desarrollados para alcanzar los objetivos del proyecto.

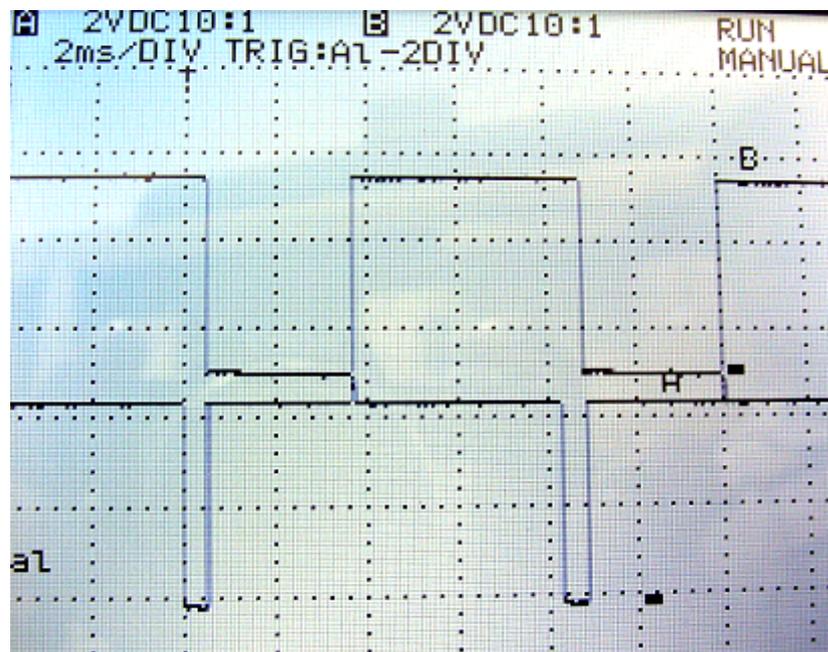


**Figura 6.5.** Forma de onda de la señal CA rectificada. Fotografía tomada de un ORC y procesadas con Paint de Windows

En la figura 6.5 se aprecia la señal a la salida del rectificador. Esta señal es utilizada con dos objetivos: detección de cruce por cero y como patrón de referencia de la tensión en la red de alimentación luego de pasarla por un divisor de tensión. De esta señal se toma el valor pico, la cual es utilizada como punto de comparación para percibir cambios en el nivel CA. La zona de la señal con valor constante de tensión, consiste en la zona de no conducción de los diodos rectificadores, alcanzando el punto de conducción aproximadamente a los 0.7 V,

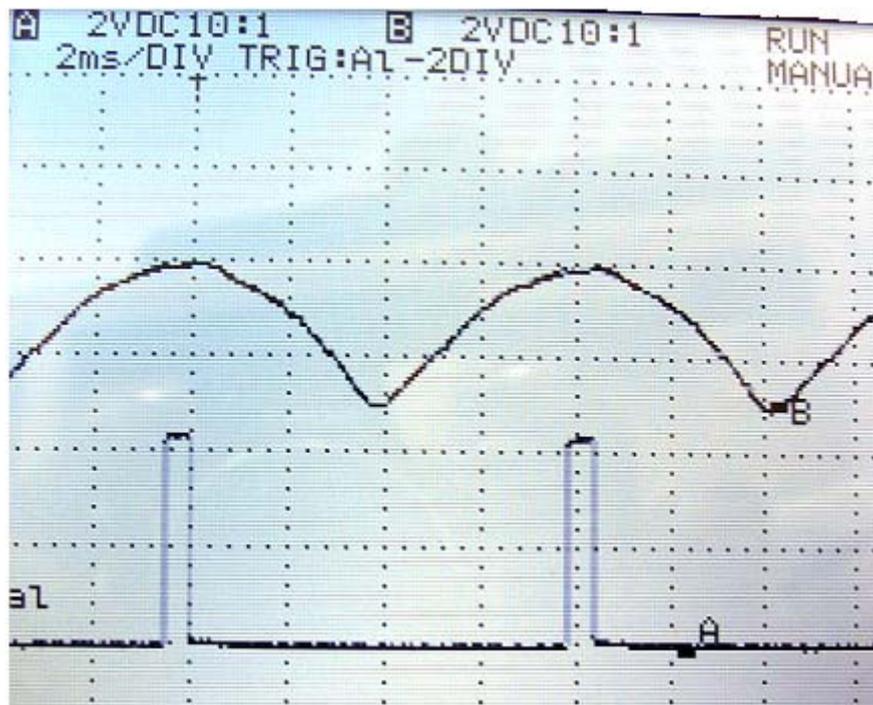


**Figura 6.6.** Señal de salida del detector de cruce por cero y salida de monoestable 1. Fotografía tomadas de un ORC y procesadas con Paint de Windows



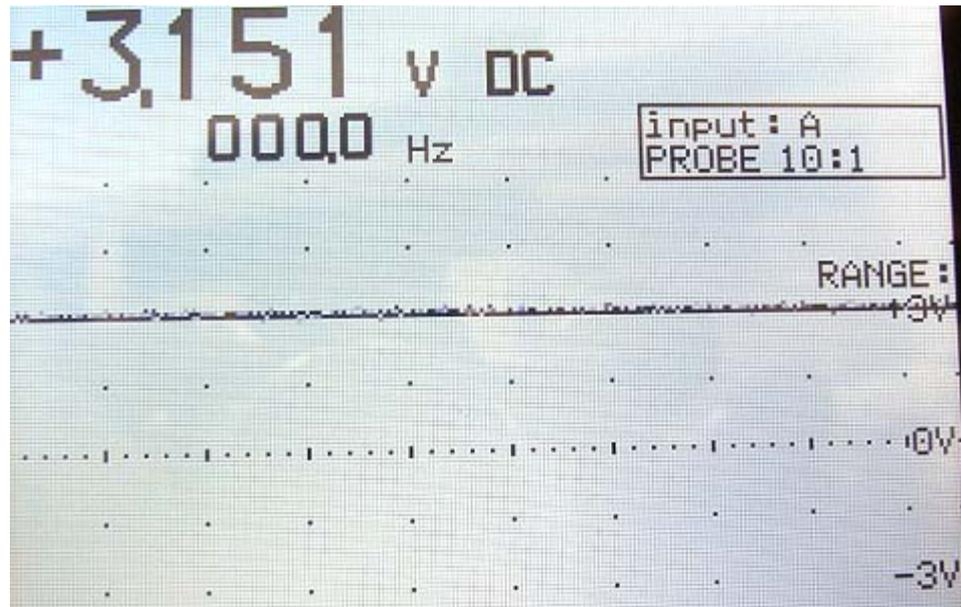
**Figura 6.7.** Señales de salida de circuitos monoestables. Monoestable 1 (arriba), monoestable 2 (abajo). Fotografía tomadas de un ORC y procesadas con Paint de Windows

Las figuras 6.6 y 6.7 muestran la sincronía en cada una de las etapas del circuito para poder tomar la muestra del valor pico de la señal CA y realizar la comparación con los valores máximos permitidos. Como se puede apreciar en la escala de tiempo en la cual fue tomada la figura (2 ms/DIV), cada una de estas señales posee un periodo de operación de 8.333 ms (120 Hz), lo cual comprueba la frecuencia de operación del sistema, definida por el puente rectificador.



**Figura 6.8.** Sincronización de módulo para obtener el valor pico de la señal CA. Fotografía tomadas de un ORC y procesadas con Paint de Windows

La figura 6.8 muestra el instante en el que se toma la muestra de la señal CA, la cual es lograda por medio de los circuitos de sincronía ya mencionados. Ahora bien, el impulso obtenido no es un impulso unitario ideal, para que deje pasar solo el valor pico de la señal CA, ya que este impulso se encuentra definido por el tiempo de conducción del cada diodo en el puente rectificador. No obstante, los cálculos fueron realizados para que este impulso pase a nivel bajo aproximadamente durante el valor pico de la señal CA.



**Figura 6.9.** Tensión almacenada en el capacitor. Fotografía tomadas de un ORC y procesadas con Paint de Windows

Finalmente, el valor pico de la señal CA a comparar con los valores máximos permitidos, es almacenada en un capacitor como se muestra en la figura 6.9. La tabla 6.1 resume los valores teóricos y experimentales de estos niveles máximos. El porcentaje de error entre ambos se debe a la incertidumbre de los componentes utilizados como resistencias y las pérdidas de tensión en los amplificadores operacionales.

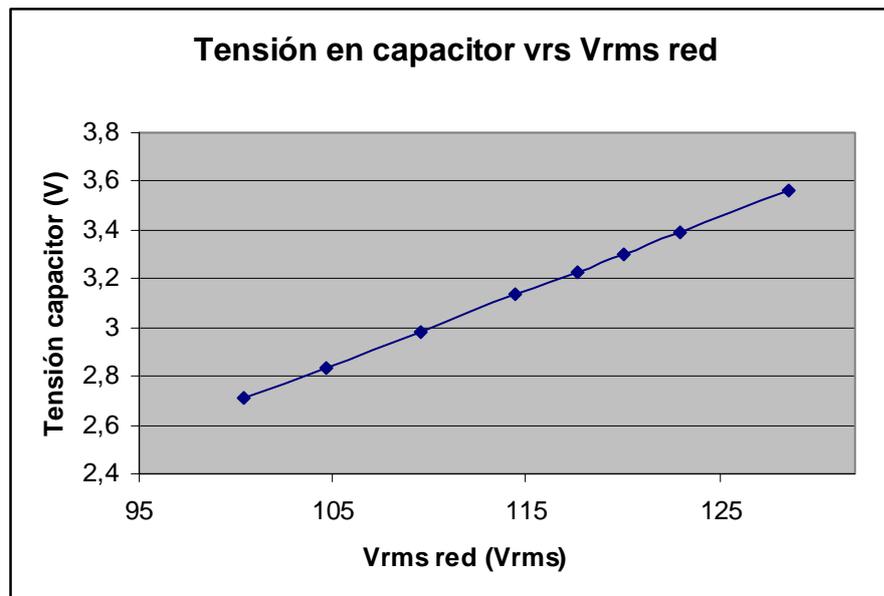
**Tabla 6.1.** Tensiones de límites de alarmas al darse cambios en el nivel RMS de la red. Para una fuente CD de 5 V

Cambio en nivel RMS	Teórico (V)	Experimental (V)	% error
-10%	2.83	2.831	0.035
+10%	3.5	3.58	2.28

**Tabla 6.2.** Valor pico de la señal de la red durante variaciones de la tensión en el valor  $V_{rms}$  de la red de alimentación

Tensión en la red ( $V_{rms}$ )	Valor pico (V)
128,6	3,56
123	3,39
120	3,3
117,7	3,23
114,4	3,139
109,6	2,98
104,7	2,83
100,4	2,71

En la tabla 6.2 se muestran los valores experimentales de tensión (valor pico de la señal CA luego del transformador y divisor de tensión) al aplicarle variaciones en el valor  $V_{rms}$  de la red. Estas pruebas fueron realizadas utilizando un generador trifásico, del cual se hizo uso de una de las fases para realizar la prueba. La figura 6.10 resume gráficamente los datos obtenidos. En esta figura se observa la tendencia de los datos, lo cual comprueba la operación en la zona lineal del transformador utilizado en el circuito durante variaciones de  $\pm 10\%$  de tensión en la red, aún sin alcanzar su valor de saturación. Los resultados, obtenidos garantizan la linealidad del sistema y por tanto, la fidelidad en los resultados del circuito.



**Figura 6.10.** Línea de tendencia del valor pico en la señal CA al variar el valor  $V_{rms}$  de la red de alimentación. Figura hecha con Excel de Microsof.

## Capítulo 7: Conclusiones y recomendaciones

### 7.1 Conclusiones

1. El reporte inmediato de eventos entre la Estación Cerro San José y el Centro de Atención de Equipos es inmediato solo si la aplicación *servidor* esta habilitada.
2. De darse una alarma en la estación sin estar el servidor remoto habilitado, la tarjeta de desarrollo intentará conectarse mientras la alarma se encuentre activa. Logrando conectarse solo cuando el sistema servidor entre en funcionamiento.
3. Los periodos de latencia debido a la utilización de polling para realizar el control de estado de alarmas, son despreciables a la percepción humana debido a las frecuencias de operación del sistema implementado.
4. La implementación de un sistema de comunicación cliente/servidor utilizando el modelo TCP/IP, permite que el sistema sea flexible geográficamente.
5. La estructura del diseño del sistema permite que se hagan expansiones o reducciones en la cantidad de eventos a monitorear, según sea la necesidad.
6. Los periodos de latencia son directamente proporcional a la cantidad de eventos a monitorear en la estación por la Unidad de Control.
7. El hardware desarrollado es óptimo para el control de eventos en la estación tomando en cuenta las capacidades de la tarjeta de desarrollo utilizada.

8. El sistema implementado en el proyecto garantiza un reporte inmediato de eventos, el cual en comparación con el sistema de reporte actual implementado en el ICE, mejora los tiempos de respuesta para llegar a una solución de fallos.

## **7.2 Recomendaciones**

1. Implementar la interfaz gráfica del PC en distintos lugares dentro de la red ICE, haría posible ver los eventos en la estación desde dichos lugares.
2. Implementar distintas interfaces gráficas en una PC de manera que cada una espere conexión de un cliente, haría posible el monitoreo de distintas estaciones en una misma PC.
3. Para realizar expansiones o reducciones del sistema es necesario tener presente la estructura y orden en que se ubica cada punto de evento. Puesto que, tanto a nivel del programa en el PC como a nivel de la Unidad de Control, cada punto de evento posee una ubicación específica.
4. El tipo de sistema implementado, propicia la mensajería automática a teléfonos celulares
5. Diseñar soluciones alternativas para sitios que requieran de mayor control

## Bibliografía

1. Barajas, Saulo. Protocolos TCP/IP. Diciembre 8, 2001. <<http://www.saulo.net/pub/tcpip/a.htm>>. [Consulta: Agosto 4, 2008]. Información acerca de TCP/IP.
2. Estivill, A y Urbano, C. Como citar recursos electrónicos. Escuela Universitaria Jordi Rubí. Information World en Español, Septiembre 2007. [Consulta: Agosto 11, 2008]. Estructura de citas bibliográficas.
3. Mansfield, Richard. Visual Basic Express Edition 2005 for Dummies. Wiley Publishing, Inc. Canada and Indiana. Chapter 13: Managing Dataset. [Consulta: Mayo 25, 2008]. Información acerca del manejo de base de datos con SQL.
4. Microsoft Corporation. Microsoft Developer Network. 2008. <<http://msdn.microsoft.com/es-es/default.aspx>>. [Consulta. Mayo 20, 2008]. Información para la elaboración de la interfaz gráfica desarrollada en el PC.
5. Microsoft, Corparation. MSDN: Media Center. 2008. <<http://www.microsoft.com/spanish/msdn/latam/mediacenter/>>. [Consulta: Mayo 22, 2008].
6. Modtronix Engineering. Example TCP/IP projects for the SBC65EC. July 30, 2007. [examples\\_v248/tcp/src/](http://www.modtronix.com/examples_v248/tcp/src/). [Julio 15, 2008]. Códigos fuentes de proyectos para la tarjeta SBC65EC.
7. Modtronix Engineering. SBC65EC: Ethernet enabled Single Board Computer (Product datasheet). <<http://www.modtronix.com/products/sbc65ec>>. [Consulta Mayo 15, 2008]. Información sobre la tarjeta de desarrollo SBC65EC.
8. Modtronix Engineering. SBC65EC. August, 2008. <<http://www.modtronix.com/products/sbc65ec>>. [Consulta Mayo 15, 2008]. Información sobre la tarjeta de desarrollo SBC65EC.
9. Rajbharti, Nilesh. AN833: The Microchip TCP/IP Stack. Microchip Technology Inc. 2002. [Consulta: Junio 23, 2008]. Información sobre TCP/IP de Microchip.
10. Textos científicos. Fast Ethernet. Febrero 25. 2008. <<http://www.textoscientificos.com/redes/ethernet/fast-ethernet>>. [Consulta: Agosto 4, 2008]. Información sobre Fast Ethernet.

11. Hayt, W; Kemmerly, J; Durban, S (2003). *Análisis de circuitos en ingeniería*. 6ta edición. McGraw-Hill Interamericana. México.  
Conceptos básicos de circuitos eléctricos.
12. Coughlin, R; Driscoll, F (1999). *Amplificadores operacionales y circuitos integrados lineales*. 5ta edición. Pearson Prentice Hall. México.  
Conceptos básicos de amplificadores operacionales.
13. Hugo, Jaime. Montaje- 2 Circuitos de muestreo de señal. Saber Electrónica, Editorial Quark, Web Electrónica: Club SE, Saber. <<http://www.webelectronica.com.ar/news07/nota02.htm>> [Consulta: Julio 28, 2008].  
Muestreo de señales analógicas.

## Apéndices y anexos

### A1. Glosario

- **Eventos mediante polling:** Este principio es aplicado generalmente en computación. Es utilizado para realizar una consulta constante a distintos puntos de similares características de un determinado objeto. Es de menor eficiencia que las interrupciones, puesto que, la latencia de cada punto de consulta depende del número de puntos totales.
- **10BASE-T:** Este protocolo es una variedad de protocolo de red Ethernet, desarrollado por la IEEE 802.3. De su nombre se derivan algunas características de funcionamiento. El número 10 indica la tasa de transferencia, la cual se encuentra en unidades de Mbits/s. BASE implica que la información transmitida se encuentra en banda base y T indica que se utiliza cable par trenzado o UTP. El tipo de conector utilizado es el RJ-45 para las conexiones a la red.
- **Multiplexor (MUX):** Dispositivo con mas de una entrada y al menos una salida, en la cual se muestra las entradas mediante la selección individual de cada una por medio de variables de control.
- **ORC:** Osciloscopio de rayos catódicos.
- **PIC:** Microcontrolador distribuido por la casa de fabricación Microchip.

## **A2. Manual de usuario**

### **Introducción**

En el siguiente documento se presenta una explicación detallada acerca del uso de las aplicaciones realizadas durante el desarrollo del proyecto “Sistema de reporte de eventos de la Estación Cerro San José”. Estas se encuentran divididas en dos secciones principales:

- A nivel de hardware
- A nivel de software

El funcionamiento de las dos secciones anteriores en conjunto nos trae como resultado una solución al problema planteado.

## 1. Interfaz gráfica en el PC

Previo a correr la aplicación en el PC de la interfaz gráfica, es necesario ubicar los siguientes archivos .xml en (C:) del disco duro de la PC donde se va a usar la interfaz gráfica: *PrioridadEsquema.xml*, *PrioridadDatos.xml*, *EventoEsquema.xml*, *EventoDatos.xml*, *EmpleadoEsquema.xml* y *EmpleadoDatos.xml*. Estos archivos se encuentran en el CD-ROM de la aplicación, en la carpeta: “Base de datos”.

De igual forma a lo anterior, antes de poner en funcionamiento la aplicación como servidor es necesario ubicar en (C:) del disco duro, la carpeta “Sonidos”, ubicada en el CD-ROM. En esta se encuentra el archivo de audio de la alarma, el cual se carga desde la aplicación.

Para poner en funcionamiento la interfaz gráfica, únicamente hay que ejecutar *AplicacionCSJ*. Al ejecutar la aplicación aparece en pantalla lo siguiente:

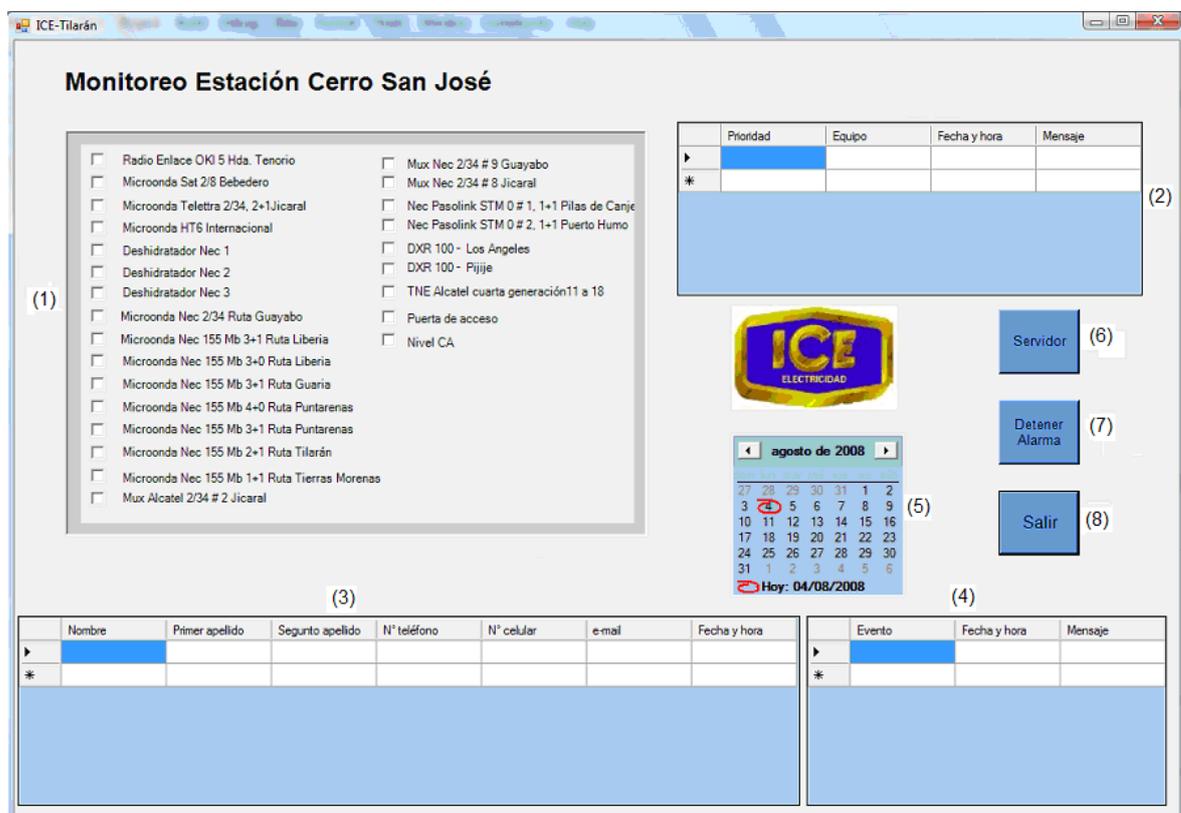


Figura 1. Interfaz gráfica en el PC.

### (1) Panel de monitoreo de equipo

En este panel se muestran los equipos y tipo de eventos monitoreados por el sistema. Al darse una conexión del cliente al servidor para reportar un evento, un check aparece en el recuadro izquierdo, señalando el equipo o evento que provocó la alarma. Ejemplo:

<input checked="" type="checkbox"/>	Mux Alcatel	<input type="checkbox"/>	TNE Alcatel	<input type="checkbox"/>	Telettra	<input type="checkbox"/>	E13
<input type="checkbox"/>	Mux Nec	<input type="checkbox"/>	Sist de Rect	<input type="checkbox"/>	HT6	<input type="checkbox"/>	E14
<input type="checkbox"/>	Nec Pasolink	<input checked="" type="checkbox"/>	OKI 5 Hda	<input type="checkbox"/>	Deshidratador	<input type="checkbox"/>	Puerta
<input type="checkbox"/>	DXR 100	<input type="checkbox"/>	Microonda	<input type="checkbox"/>	E12	<input type="checkbox"/>	Nivel CA

**Figura 2.** Equipo de monitoreo en la estación.

Según lo anterior el equipo Mux Alcatel y OKI 5 Hda reportan una alarma.

### (2) Interfaz de base de datos: Fallo de equipo

En esta sección se registra en la base de datos y se muestra al usuario del software los reportes de fallos en equipos luego de que estos ocurran. Esta base de datos está estructurada en 4 partes: nivel de prioridad de la alarma, equipo alarmado, fecha y hora del fallo y un cuadro de mensaje de procedimiento a seguir.

### (3) Interfaz de base de datos: Ingreso de personal a la estación

De igual manera que la sección anterior, en esta se registra los eventos de personas a la estación. No obstante, esta sección está implementada en software pero no en hardware, por lo que es solo previsto para futuras ampliaciones y mejoras. En su estructura se registra: nombre y apellidos de la persona que ingresó, número de teléfono fijo, número de teléfono celular, dirección de correo electrónico y la fecha y hora de ingreso.

### (4) Interfaz de base de datos: Eventos en la estación

En esta sección se registra en la base de datos y muestra al usuario del programa los eventos que se dan dentro de la estación y que no son parte del ambiente normal de la misma. La base de datos se encuentra

estructurada en tres partes: El evento, la fecha y hora en que se dio este evento y un cuadro de mensaje del procedimiento a seguir.

**(5) Calendario**

Al correr la aplicación se muestra en la interfaz un calendario del mes en curso.

**(6) Botón Servidor**

Al pulsar el botón Servidor se activa el programa para que funcione como un servidor. El cual se encuentra pendiente de los datos entrantes desde host remotos (clientes).

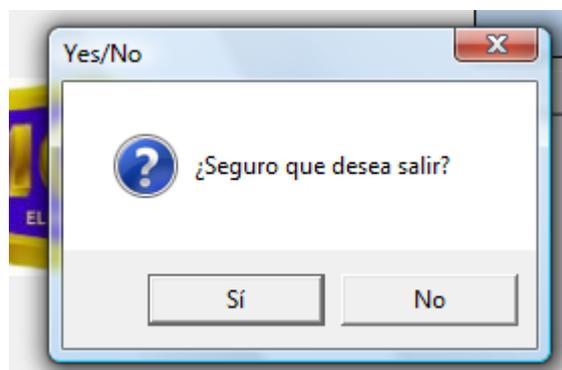
Para poder recibir reporte de eventos o alarmas, es necesario pulsar este botón al iniciar la aplicación y luego de darse una reporte.

**(7) Botón Detener Alarma**

Luego de darse una alarma reportada desde el cliente, se genera una alarma audible en el PC. Ésta es posible detenerla pulsando el botón *Detener Alarma*.

**(8) Botón Salir**

Al pulsar el botón Salir, se muestra en pantalla el siguiente mensaje:



**Figura 3.** Opciones de salida de la interfaz gráfica.

En el cual se toma la decisión de salir o no de la ejecución del programa.

## Restricciones

Al darse un evento o alarma durante la ejecución de la aplicación, el día y hora es tomada de la PC donde se corre el programa. Por lo que es necesario verificar que estos datos en el PC.

Ejemplo:

Paso 1. Ejecute la aplicación.

Paso 2. Pulse el botón *Servidor*

En este momento el programa se encuentra esperando que se de una alarma enviada desde un cliente.

Al darse una alarma se muestra lo siguiente:

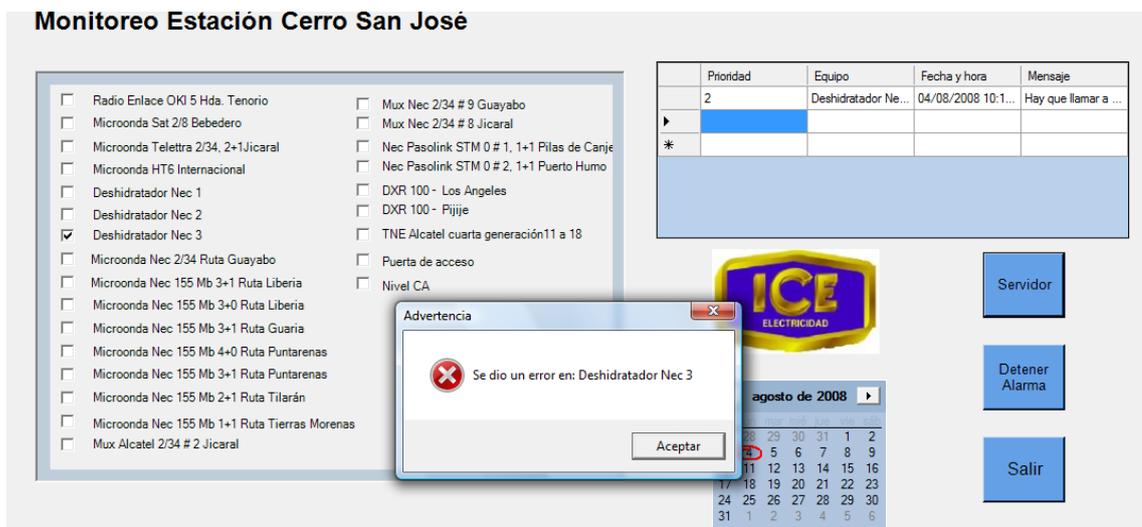


Figura 4. Reporte de alarmas en el PC.

En este caso se puede ver, que el equipo que reporta una alarma es el Deshidratador Nec 3.

Paso 3. Pulse el botón *Detener Alarma* para detener el sonido de alarma.

Se puede corroborar el equipo reportado y la prioridad en la base de datos

s de Canje  
 rto Humo  
 a 18

	Prioridad	Equipo	Fecha y hora	Mensaje
	2	Deshidratador Ne...	04/08/2008 10:1...	Hay que llamar a ...
	1	Deshidratador Ne...	04/08/2008 10:2...	Hay que llamar a ...
	0	Microonda Sat 2/...	04/08/2008 10:2...	Hay que llamar a ...
	2	Deshidratador Ne...	04/08/2008 10:2...	Hay que llamar a ...
▶				
*				

**Ultima alarma reportada**

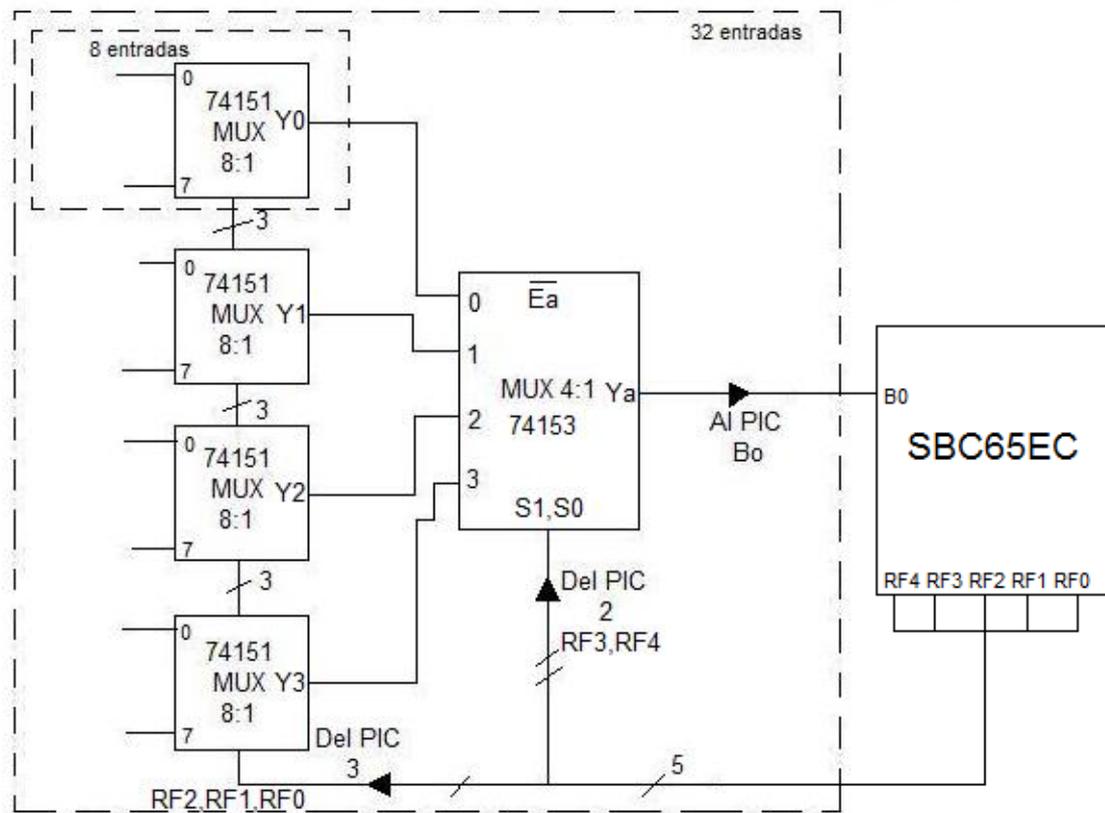
**Figura 5.** Almacenamiento en base de datos.

Paso 4. Pulse nuevamente el botón Servidor para continuar recibiendo datos desde el cliente.

## 2. A nivel de hardware

A continuación se presentan los módulos implementados en el proyecto, así como los pines de conexión entre ellos.

Para realizar el monitoreo de las alarmas en cada uno de los equipos, se utiliza el método de polling, para lo cual se hace uso de los multiplexores 74151 y 74153 de ocho y cuatro entradas respectivamente, los cuales se colocan en cascada como se muestra a continuación.



**Figura 6.** Diagrama de bloques del sistema implementado.

Los bits más significativos de selección (RF4 y RF3) de la tarjeta de desarrollo se conectan al multiplexor 74153. Estos son los encargados de la selección de los cuatro módulos individuales con capacidad de ocho alarmas cada uno.

**Tabla 1.** Selección de módulos mediante polling.

S1/RF4	S0/RF3	Módulo
0	0	0
0	1	1
1	0	2
1	1	3

Los bits menos significativos provenientes de la tarjeta de desarrollo (RF2, RF1 y RF0), se conectan a los pines de selección del multiplexor 74151 (C, B y A respectivamente) para el control de selección de cada una de las alarmas.

En el caso de que se utilice solo un módulo para realizar el monitoreo de alarmas, la conexión entre el módulo y la tarjeta de desarrollo se puede realizar de igual forma. Conectando la salida de dicho módulo (OUT) a la entrada RB0 de la tarjeta de desarrollo y los pines de selección como se indica a continuación: RF0/P0, RF1/P1 y RF2/P2. Sin embargo, el orden de conexión en cada módulo debe ser considerado durante la conexión (ver tabla 1)

## **2.1 Tarjeta de desarrollo SBC65EC**

Esta tarjeta de desarrollo se encuentra debidamente programada para realizar el monitoreo de 25 alarmas. No obstante, realizando las debidas modificaciones al programa original es posible expandir o disminuir esta cantidad de alarmas.

Pasos a seguir para realizar el monitoreo de alarmas en el lugar remoto:

1. Alimentar con las tensiones CD indicadas la tarjeta SBC65EC.
2. Conectar la tarjeta a la red (Intranet) del ICE.
3. Realizar las conexiones entre la tarjeta SBC65EC y el recolector de alarmas.

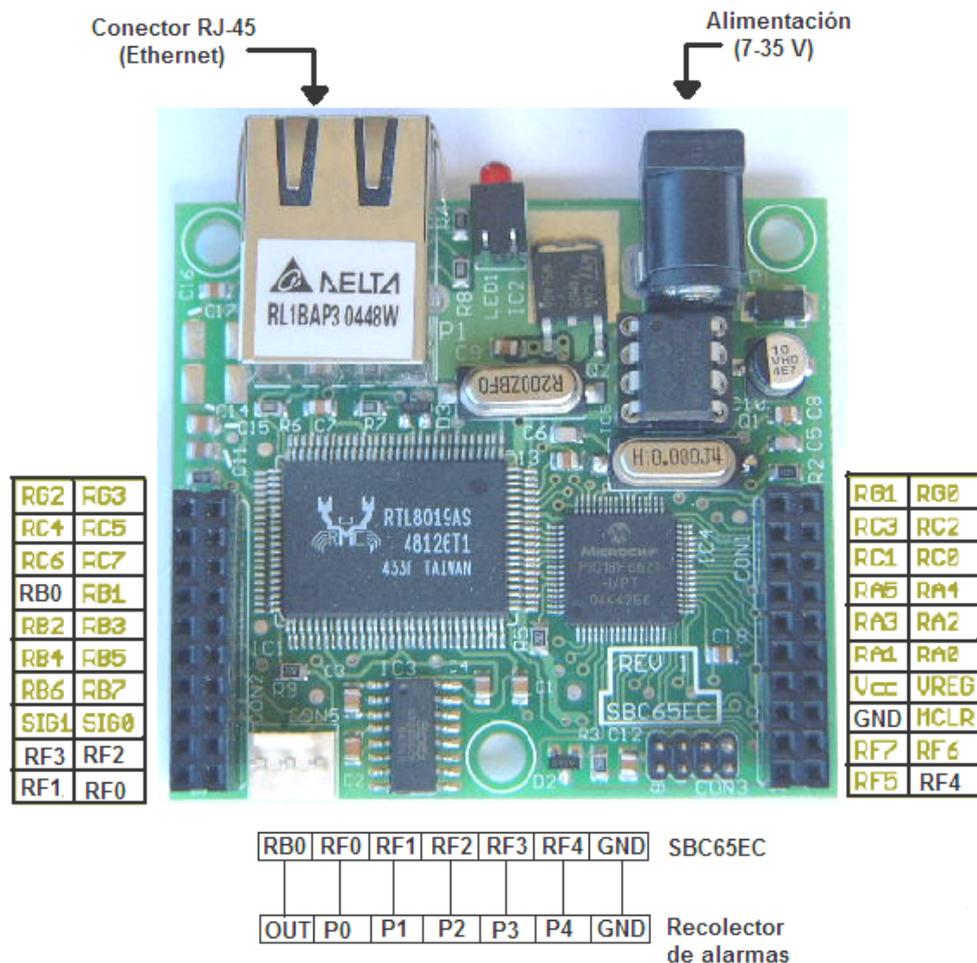
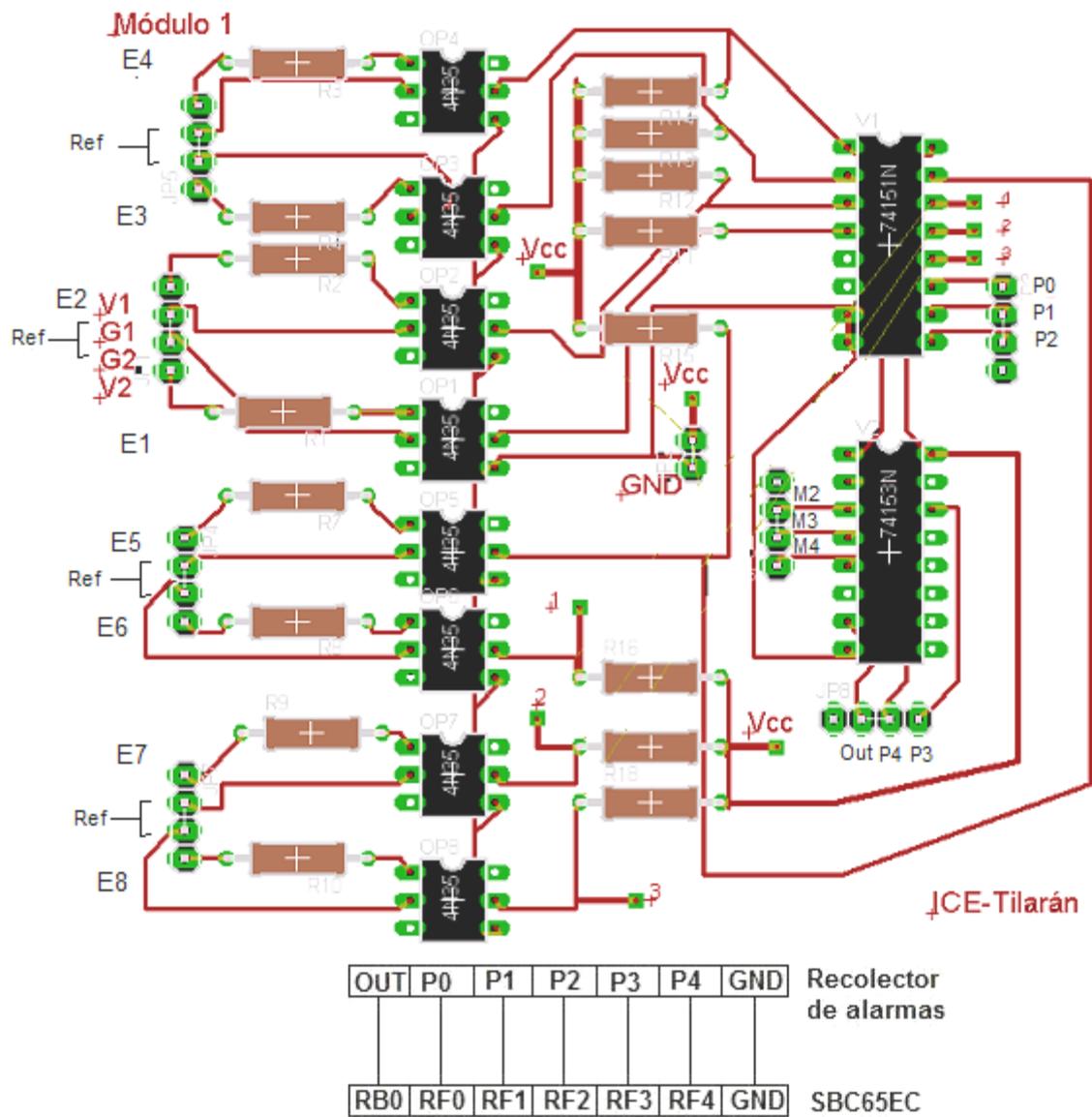


Figura 7. Puertos de conexión en la tarjeta SBC65EC. Figura tomada de [8].

## 2.2 Recolector de alarmas

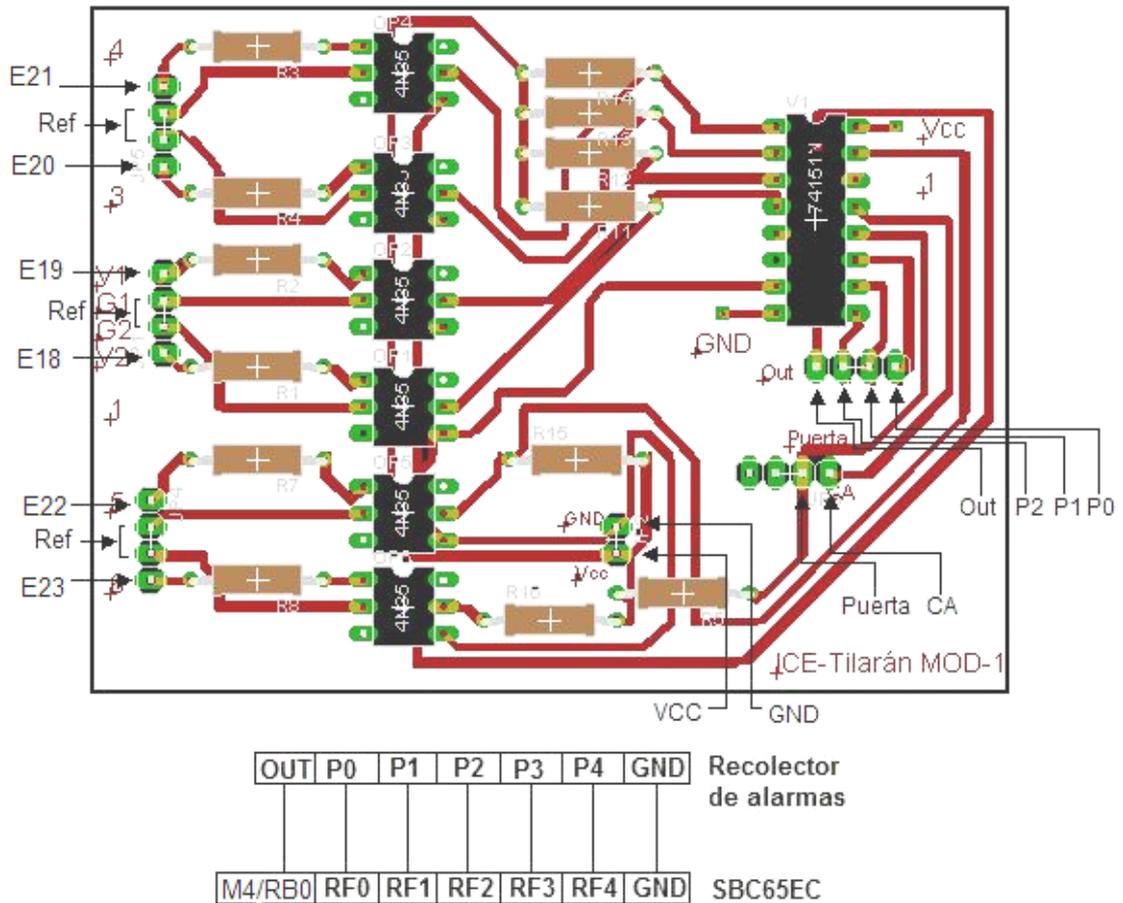
El siguiente circuito implementado en circuito impreso, está diseñado para monitorear equipos, los cuales poseen una señal de alarma con una diferencia de potencial de 48 V.



**Figura 8.** Circuito recolector de alarmas. Módulo 1.

Las etiquetas, E1, E2,...E8 son para conectar debidamente las alarmas del equipo que corresponde en cada puerto de la placa. En los puertos REF, se debe conectar la referencia (Ref(-)) del sistema.

Módulo 4



**Figura 9.** Circuito recolector de alarmas. Módulo 4.

Pines de conexión del recolector de alarmas: En estos pines se conectan las salidas de alarmas tomadas de los equipos mencionados.

E18: Mux Nec 2/34 # 8 Jicaral

E19: Nec Pasolink STM 0 # 1, 1+1 Pilas de Canjel

E20: Nec Pasolink STM 0 # 2, 1+1 Puerto Humo

E21: DXR 100 – Los Angeles

E22: DXR 100 – Pijije

E23: TNE Alcatel cuarta generación 11 a 18

Ref: Nivel de tensión de referencia de alarmas.

**Tabla 2.** Codificación de alarmas en la tarjeta SBC65EC.

RF4	RF3	RF2	RF1	RF0	Equipo de monitoreo
0	0	0	0	0	E1: Radio Enlace OKI 5 Hda. Tenorio
0	0	0	0	1	E2: Microonda Sat 2/8 Bebedero
0	0	0	1	0	E3: Microonda Telettra 2/34, 2+1Jicaral
0	0	0	1	1	E4: Microonda HT6 Internacional
0	0	1	0	0	E5: Deshidratador Nec 1
0	0	1	0	1	E6: Deshidratador Nec 2
0	0	1	1	0	E7: Deshidratador Nec 3
0	0	1	1	1	E8: Microonda Nec 2/34 Ruta Guayabo
0	1	0	0	0	E9: Microonda Nec 155 Mb 3+1 Ruta Liberia
0	1	0	0	1	E10: Microonda Nec 155 Mb 3+0 Ruta Liberia
0	1	0	1	0	E11: Microonda Nec 155 Mb 3+1 Ruta Guaria
0	1	0	1	1	E12: Microonda Nec 155 Mb 4+0 Ruta Puntarenas
0	1	1	0	0	E13: Microonda Nec 155 Mb 3+1 Ruta Puntarenas
0	1	1	0	1	E14: Microonda Nec 155 Mb 2+1 Ruta Tilarán
0	1	1	1	0	E15: Microonda Nec 155 Mb 1+1 Ruta Tierras Morenas
0	1	1	1	1	E16: Mux Alcatel 2/34 # 2 Jicaral
1	0	0	0	0	E17: Mux Nec 2/34 # 9 Guayabo
1	0	0	0	1	E18: Mux Nec 2/34 # 8 Jicaral
1	0	0	1	0	E19: Nec Pasolink STM 0 # 1, 1+1 Pilas de Canjel
1	0	0	1	1	E20: Nec Pasolink STM 0 # 2, 1+1 Puerto Humo
1	0	1	0	0	E21: DXR 100 – Los Angeles
1	0	1	0	1	E22: DXR 100 – Pijije
1	0	1	1	0	E23: TNE Alcatel cuarta generación11 a 18
1	0	1	1	1	Nivel CA
1	1	0	0	0	Puerta de acceso
1	1	0	0	1	No implementado
1	1	0	1	0	No implementado
1	1	0	1	1	No implementado
1	1	1	0	0	No implementado
1	1	1	0	1	No implementado
1	1	1	1	0	No implementado
1	1	1	1	1	No implementado

### **A3. Manual de mantenimiento**

#### **Introducción**

En este documento se presentan especificaciones de funcionamiento de los dispositivos y módulos implementados. La intención es proporcionar una guía que facilite una corrección o reparación en caso que uno de los módulos implementados falle.

## Señalización en la tarjeta de desarrollo

Es la unidad fundamental en la estación remota, puesto que es la que realiza la conexión con el servidor de darse un fallo o evento.

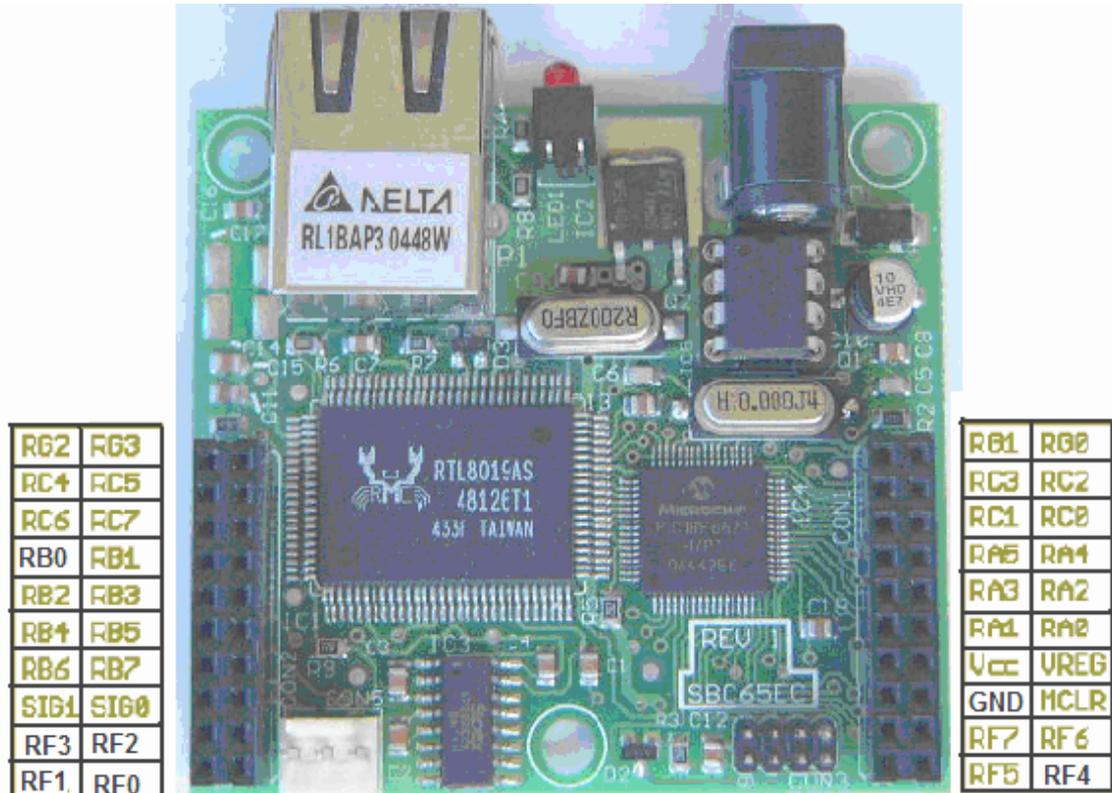


Figura 1. Tarjeta de desarrollo SBC65EC. Figura tomada de [8].

### 1. Verificación en caso de fallos en la tarjeta de desarrollo

**Alimentación:** Verificar que la tarjeta de desarrollo es alimentada con una fuente CD igual o superior a 8 V. Por lo que el puerto Vcc indicada en la figura anterior debe permanecer a 5 V y GND (referencia) a 0 V.

**Conexión:** Al alimentar la tarjeta de desarrollo y conectarla a la red del ICE, en la tarjeta se encenderá un LED amarillo constante y un LED naranja se enciende y se apaga en lapsos aproximados de 0.5 y 1 s.

**Puertos de salida/entrada:** El puerto RBO normalmente sin condición de alarma debe permanecer en un nivel de tensión ALTO (5 V). En caso de alarma este puerto pasa a BAJO (0 V).

## 2. Verificación de fallos en módulos recolectores de alarmas

La patilla 1 de los opto acopladores (4N25) debe encontrarse a una diferencia de potencial de 10 V aproximadamente respecto a su referencia (ref) en caso de no presentar alarma y una diferencia de potencial de 48 V cuando se presente una alarma.

La patilla 5 de los opto acopladores normalmente (sin alarma) debe estar a una diferencia de potencial de 5 V respecto a la patilla 4 del mismo circuito integrado y a una diferencia de potencial de 0 V en caso de presentarse una alarma.

## 3. Verificación de fallos en circuito detector de cambios de nivel CA

Ya se ha mostrado y explicado en el capítulo de Resultados Experimentales y Análisis, las principales señales que debe haber en distintos puntos del circuito detector de cambios en el nivel CA para que este funcione de manera correcta. No obstante, en esta sección se explicaran otras señales de relevancia en el funcionamiento del circuito que pueden ser corroboradas por el usuario utilizando equipo de medición básico.

### - Circuito integrado 4016B:

**Tabla 1.** Señalización CI 4016B.

Patilla	Señal
1	Señal CA rectificadora de onda completa, con un valor pico de 3.15 V.
2	3.15 V
13	Señal pulsante, con un periodo de 8.33 ms.

**- Circuitos integrados LM741**

**Tabla 2.** Niveles de tensión en las patillas de CI 741. Comparador de cambio -10% en el nivel CA.

Patilla	Condición normal (V)	Sobrepasando límites (V)
6	1.2	1.7
2	> 2.83	< 2.83
3	2.83	2.83

**Tabla 3.** Niveles de tensión en las patillas de CI 741. Comparador de cambio +10% en el nivel CA.

Patilla	Condición normal (V)	Sobrepasando límites (V)
6	1.2	1.7
2	< 3.58	> 3.58
3	3.58	3.58