

Instituto Tecnológico de Costa Rica

Escuela de Ingeniería Electrónica



TEC

Instituto Tecnológico de Costa Rica

Automatización del sistema de configuración de parámetros y monitoreo de operación para las pruebas de choque térmico de los microprocesadores Intel

Componentes Intel de Costa Rica

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

Johan Carvajal Godínez

Agosto 2004

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

Ing. Arys I. Carrasquilla Batista

Profesora Asesora

Escuela de Ingeniería en Electrónica

Instituto Tecnológico de Costa Rica

Firma: Arys I. Carrasquilla B.



Ing. Jorge Luís Rojas del Barco

Asesor por la empresa

Componentes Intel de Costa Rica S.A.

Firma: Jorge L. Rojas

COMPONENTES INTEL DE
intel. COSTA RICA, S.A.
CR3-1, Calle 129,
La Ribera de Belén, Costa Rica

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

Belén, 6 de agosto del 2004

*A Dios y mis seres queridos,
por todo el apoyo que me han dado...*

Agradecimiento

Agradezco a todas esas personas que lo largo de mi vida han contribuido para que el día de hoy yo pueda conseguir esa meta que me puse cuando era un niño.

A Dios, porque fuiste el eje que me guió hasta donde estoy.

A mi familia, gracias por haberme apoyado e impulsado a estudiar durante estos años.

A la familia Kohkemper Agüero, gracias por haber creído en mí y darme su apoyo.

A mis amigos, no tengo con que devolverles todo el apoyo que he recibido de ustedes.

A los profesores, gracias por poner de su parte todo lo posible para que yo creciera como profesional.

A Componentes Intel de Costa Rica, por darme la oportunidad de demostrar lo que soy como profesional y confiar en mí para la ejecución del proyecto.

¡A todos les debo este triunfo, por eso se los dedico a ustedes!

Gracias inmensurables,

Johan Carvajal Godínez

Resumen

La planta de ensamble de microprocesadores y chipsets para computadoras: *Componentes Intel de Costa Rica*, en su afán por cumplir con un papel clave dentro de la corporación, siempre ha buscado estar en constante mejoramiento tanto en calidad como en innovación de sus productos. Esto le ha permitido ser el líder mundial en su rama y continuar a como punta de lanza en desarrollo de tecnología.

Por ello, el control de calidad aplicado tanto al producto como al proceso que se utiliza en la fabricación de los dispositivos es muy importante, ya que permite garantizar a los clientes que el producto que reciben va a satisfacer sus necesidades tanto en desempeño como en durabilidad. Uno de los pasos para determinar la calidad consiste en tomar el producto y exponerlo a condiciones ambientales extremas; luego se prueba el desempeño del mismo y, con esto, se determina cómo será su operación a través del tiempo de funcionalidad.

Para esta actividad se utilizan hornos especializados, los cuales siempre han sido operados manualmente. Esto implica que se debe estar presente en el sitio de operación para su configuración y monitoreo, además de que se han dado casos en que se cargan parámetros de prueba erróneos, lo cual impacta el flujo de la prueba.

Los hornos son manejados por controladores lógicos programables (PLC) a través de una pantalla de tacto, ya que este es el sistema que traen de fábrica.

Para aumentar la confiabilidad sobre el funcionamiento de los hornos y eliminar la probabilidad de cometer errores a la hora de cargar las recetas se decidió desarrollar una plataforma de monitoreo y control que pudiera implementarse sobre todos los equipos que realizan este tipo de pruebas. El diseño e implementación de dicha plataforma, es el tema central que se aborda en este documento.

Al final, se obtiene una herramienta amigable con el usuario, robusta en funcionamiento, que resuelve las necesidades planteadas por la empresa, garantizando así, más calidad en el producto que se entrega al cliente final.

Palabras clave: PLC, Automatización, HMI, calidad, control, monitoreo

Abstract

Components Intel of Costa Rica, in its effort to be one of the key manufacturing sites of the corporation, has been working on the continuous improvement of quality and innovation of its products. This is the reason why Intel Corporation is the worldwide leader on CPU manufacturing.

Quality control of the product and its manufacturing process are the key to keep the company on the top because it guarantees the product reliability. This fact implies that the company has to do tests for defining the reliability life for each product. To do that, the product is stressed with extreme temperature patterns.

There are ovens for stressing the product. These ovens are manually operated. This implies that the user has to enter the test information using a touch screen interface. Also, for performance monitoring, the user has to have physical access to the equipment, because it does not exist any other mechanism for doing that. There are some issues related to wrong data configuration of equipment which impacts the test process flow.

The ovens are controlled by “programmable logic controllers” (PLC), and the user interface is a touch screen display.

To increase the equipment’s control and the process confidence level Intel Corporation has decided to design and implement a platform for the monitoring and control of them. The development of this platform is the main objective of this document.

This system improves loading, monitoring of parameters and introduces a sub-system for data logging, which makes more reliable the process’s performance.

Finally, the corporation wants, is to guarantee to final clients that the product they buy has the quality and reliability required for a good performance during the operation time.

Key words: PLC, Automation, HMI, Quality, reliability, control, monitoring

Índice general

| | | |
|-------------------|---|-----------|
| Capítulo 1 | Introducción | 13 |
| 1.1 | <i>Descripción de la empresa</i> | 13 |
| 1.2 | <i>Definición del problema y su importancia</i> | 16 |
| 1.2.1 | <i>Descripción general del problema</i> | 16 |
| 1.2.2 | <i>Descripción detallada del problema</i> | 18 |
| 1.3 | <i>Objetivos</i> | 20 |
| 1.3.1 | Objetivo general | 20 |
| 1.3.2 | Objetivos específicos | 20 |
| Capítulo 2 | Antecedentes del proyecto | 22 |
| 2.1 | <i>Estudio del problema a resolver</i> | 22 |
| 2.2 | <i>Requerimientos de la empresa</i> | 24 |
| 2.3 | <i>Solución propuesta</i> | 24 |
| 2.3.1 | <i>Red de equipos</i> | 26 |
| 2.3.2 | <i>Selección de la receta</i> | 28 |
| 2.3.3 | <i>Codificación de las recetas</i> | 29 |
| 2.3.4 | <i>Notificación de alarmas</i> | 30 |
| 2.3.5 | <i>Registro de proceso</i> | 30 |
| Capítulo 3 | Procedimiento metodológico | 32 |
| Capítulo 4 | Descripción del hardware | 36 |
| 4.1 | <i>Hardware de la cámara</i> | 36 |
| 4.1.1 | <i>Controlador lógico programable SLC 5/03</i> | 36 |
| 4.1.2 | <i>Tarjetas de expansión familia SLC 500</i> | 37 |
| 4.1.3 | Tarjeta de entradas analógicas 1746-NI16I | 37 |
| 4.1.4 | Tarjeta de entradas digitales 1746-IA16 | 39 |
| 4.1.5 | Tarjeta de salidas digitales 1746-OA16 | 40 |
| 4.2 | <i>Instrumentación</i> | 41 |
| 4.2.1 | Transductores de temperatura | 42 |
| 4.3 | Hardware de la red de equipos | 42 |

| | | |
|---|--|-----------|
| 4.3.1 | Acoplador de vínculo 1747-AIC..... | 42 |
| 4.3.2 | Tarjeta de red PKTX..... | 43 |
| Capítulo 5 Descripción del software | | 44 |
| 5.1 | <i>RSLink</i> | 44 |
| 5.2 | <i>RSLogix 500</i> | 45 |
| 5.3 | <i>Panel Builder 32</i> | 47 |
| 5.3 | <i>Rsvi32</i> | 48 |
| Capítulo 6 Análisis y resultados..... | | 50 |
| 6.1 | <i>Diseño del hardware</i> | 50 |
| 6.2 | <i>Diseño de la estructura de datos</i> | 52 |
| 6.3 | <i>Diseño del software</i> | 54 |
| 6.3.1 | <i>Módulo de carga de solicitud de prueba</i> | 55 |
| 6.3.2 | <i>Módulo de carga de recetas</i> | 59 |
| 6.3.3 | <i>Módulo de monitoreo de proceso</i> | 70 |
| 6.3.4 | <i>Módulo de control de alarmas</i> | 73 |
| 6.3.5 | <i>Módulo de monitoreo virtual del equipo</i> | 74 |
| 6.3.6 | <i>Módulo de reporte de operación de los equipos</i> | 76 |
| 6.3.7 | <i>Otras consideraciones sobre el software</i> | 77 |
| 6.3.8 | <i>Alcances y limitaciones</i> | 77 |
| Capítulo 7 Conclusiones y recomendaciones..... | | 78 |
| 7.1 | <i>Conclusiones</i> | 78 |
| 7.2 | <i>Recomendaciones</i> | 79 |
| Bibliografía | | 80 |
| Apéndices y anexos | | 81 |
| Apéndice A.1 | <i>Glosario</i> | 81 |
| Apéndice A.2 | <i>Distribución de señales del RANSCO en el PLC</i> | 83 |
| Apéndice A.3 | <i>Listado de variables de proceso</i> | 88 |
| Apéndice A.4 | <i>Listado de variables asociadas con alarmas</i> | 91 |
| Apéndice A.5 | <i>Direccionamiento de red</i> | 92 |

| | | |
|------------------|---|-----------|
| Anexo B.1 | <i>Diagramas de conexión de los dispositivos de la red</i> | 93 |
| Anexo B.2 | <i>Hoja de información del proyecto</i> | 95 |

Índice de Figuras

| | | |
|------------|---|----|
| Figura 1.1 | Organigrama del departamento en el que se realizó el proyecto..... | 15 |
| Figura 1.2 | Vista de la cámara de choque térmico RANSCO..... | 17 |
| Figura 2.1 | Diagrama de relación entre los módulo del proyecto | 26 |
| Figura 2.2 | Arquitectura de la topología de la red de equipos RANSCO..... | 27 |
| Figura 2.3 | Flujo de información para la carga de parámetros..... | 29 |
| Figura 2.4 | Interfaz de usuario para programa codificador de recetas | 30 |
| Figura 2.5 | Ejemplo de interfaz para pantalla de historial de proceso..... | 31 |
| Figura 4.1 | Diagrama del procesador del PLC | 37 |
| Figura 4.2 | Diagrama de terminales tarjeta 1746-NI16I | 38 |
| Figura 4.3 | Diagrama de conexión tarjeta 1746-NI16I..... | 39 |
| Figura 4.4 | Diagrama de conexión tarjeta de entradas digitales 1746-IA16..... | 40 |
| Figura 4.5 | Diagrama de conexión tarjeta de salidas digitales 1746-OA16..... | 41 |
| Figura 4.6 | Diagrama de conexión para el 1747-AIC | 43 |
| Figura 5.1 | Pantalla principal RSLinx | 44 |
| Figura 5.2 | Pantalla de configuración de interfaz de red..... | 45 |
| Figura 5.3 | Pantalla de Rslogix500 utilizada para crear un proyecto | 46 |
| Figura 5.4 | Pantalla de configuración de los dispositivos externos del PLC | 46 |
| Figura 5.5 | Pantalla diseñada con Panel Builder 32..... | 47 |
| Figura 5.6 | Pantalla principal Rsvew32 Works | 49 |
| Figura 5.7 | Pantalla diseñada con RsView 32..... | 49 |
| Figura 6.1 | Diagrama del diseño de la base datos para proceso | 54 |
| Figura 6.2 | Interfaz de usuario para petición de pruebas | 55 |
| Figura 6.3 | Diagrama de flujo de la operación de agregar prueba | 57 |
| Figura 6.4 | Diagrama de flujo de la operación de agregar paso..... | 58 |
| Figura 6.5 | Pantalla de usuario del módulo de configuración de equipo | 59 |
| Figura 6.6 | Pantalla para realizar actualizaciones del equipo | 60 |
| Figura 6.7 | Diagrama de flujo para la actualización de los equipos | 61 |
| Figura 6.8 | Diagrama de flujo para la descarga de información a los equipos | 64 |

| | | |
|-------------|---|----|
| Figura 6.9 | Interfaz de usuario generada para asociar las pruebas con los equipos | 65 |
| Figura 6.10 | Diagrama de flujo del proceso de carga petición de prueba | 67 |
| Figura 6.11 | Diagrama de flujo para la carga de petición a REQ DB | 68 |
| Figura 6.12 | Interfaz diseñada para agregar un usuario a la PROC DB | 69 |
| Figura 6.13 | Interfaz para modificar identificador de prueba | 69 |
| Figura 6.14 | Pantalla para configuración del TrendX | 70 |
| Figura 6.15 | Pantalla para monitoreo de equipos en tiempo real | 71 |
| Figura 6.16 | Pantalla para configuración del TrendX histórico..... | 72 |
| Figura 6.17 | Gráfica histórica generada por el sistema..... | 73 |
| Figura 6.18 | Pantalla para notificar alarmas de los equipos..... | 74 |
| Figura 6.19 | Imagen de la pantalla de tacto emulada | 75 |
| Figura 6.20 | Sistema para generar reportes de operación de los equipos..... | 76 |
| Figura 6.21 | Pantalla para autenticación de usuarios | 77 |

Índice de Tablas

| | | |
|------------|---|----|
| Tabla A. 1 | Descripción de la función de las señales de entrada del PLC | 83 |
| Tabla A. 2 | Descripción de la función de las señales de salida del PLC | 85 |
| Tabla A. 3 | Conexiones asociadas a los módulos de termopares..... | 87 |
| Tabla A. 4 | Listado de las variables del PLC utilizadas..... | 88 |
| Tabla A. 5 | Variables asociadas con alarmas para cada equipo..... | 91 |
| Tabla A. 6 | Direccionamiento utilizado en la implementación de la red | 92 |

Capítulo 1

Introducción

1.1 Descripción de la empresa

Componentes Intel de Costa Rica es una empresa transnacional dedicada principalmente al ensamble y prueba de microprocesadores.

La planta de Costa Rica inició sus actividades en marzo de 1998 y es el único lugar en el que se practica la manufactura de microprocesadores en toda Latinoamérica.

Componentes Intel de Costa Rica S.A. se localiza en la Ribera de Belén, Heredia. La planta cuenta con aproximadamente 3 000 empleados y tiene como principal función el ensamble y prueba de microprocesadores, aunque recientemente decidió extender su operación a manufactura de chipsets.

Intel cuenta con 6 años de experiencia en Costa Rica y es importante recalcar que desde su comienzo ha formado parte de una lista de importantes empresas, en la que ocupa el puesto líder en la aplicación de los más altos estándares y políticas de medio ambiente, salud y seguridad ocupacional.

Esta empresa fue fundada en 1968 inicialmente como Noyce & Moore electronics, nombre que en 1973 fue cambiado por **Integrated Electronics (INTEL)**, para darle una visión diferente al cliente sobre los productos que se realizan.

Actualmente Intel es el líder mundial en la fabricación y desarrollo de microprocesadores. Datos recientes reportan que en esta empresa trabajan más de 80 mil empleados de 45 países alrededor del mundo.

La planta de Intel en Costa Rica funciona hoy con niveles de eficiencia y productividad iguales o mayores a otras plantas similares, en donde la calidad del recurso humano ha jugado un papel clave en el éxito.

El proyecto expuesto en el presente documento se desarrolla en el departamento de Calidad y Confiabilidad (**Quality & Reliability – Q&R**) el cual se dedica a asegurar la calidad y confiabilidad de los productos de Componentes Intel de Costa Rica, teniendo dentro de sus principales tareas asegurar los procesos adecuados para tal fin.

Su misión principal es: “proveer productos y servicios de clase mundial, por medio de soluciones sistemáticas, innovadoras y su conocimiento técnico colectivo.”

Su visión consiste en la idea de que la calidad y la confiabilidad juntas constituyen elementos clave para la obtención del éxito, estableciendo de este modo estándares en competencia técnica, flexibilidad operacional y continua expansión del potencial de sus empleados.

Los objetivos de este departamento son:

- a. Crecimiento por medio de la identificación y el manejo de las oportunidades clave para consolidar a la organización como un elemento importante en los negocios.
- b. Excelencia operacional por medio de la implementación de sistemas y servicios de calidad competitiva.
- c. Promoción del desarrollo personal mientras se define un competente sistema organizativo del departamento que permita responder a situaciones complejas.
- d. Construcción de herramientas y servicios que permitan el control de fallas.

La división de calidad y confiabilidad (Q&R), donde se realizó el proyecto se denomina Technical LAB. Dentro de esta área se encuentran los siguientes laboratorios: Stress LAB, Calibration LAB, Dimensional LAB y LAB System.

En la Figura 1.1 se presenta un organigrama que muestra la jerarquía de las divisiones mencionadas anteriormente.

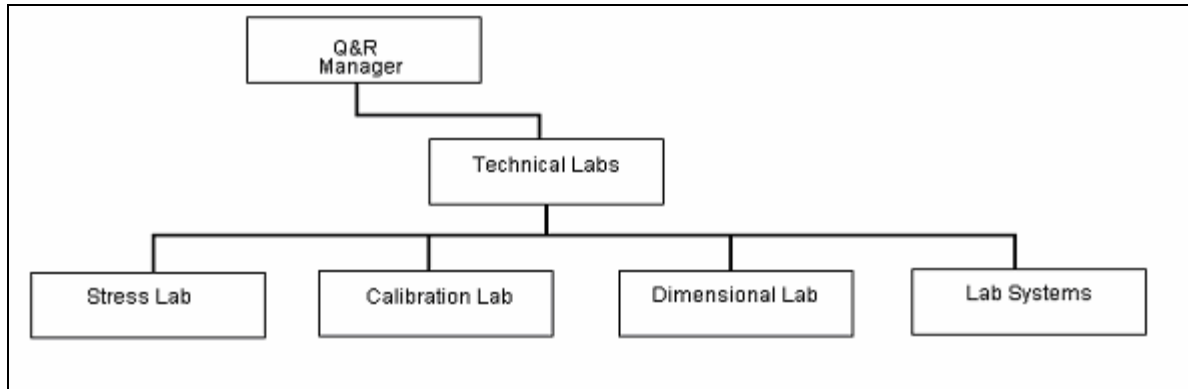


Figura 1.1 Organigrama del departamento en el que se realizó el proyecto

El proyecto en cuestión se realizó con el grupo de Stress LAB, el cual se dedica a realizar pruebas con las unidades para determinar comportamientos relacionados con ambientes extremos y así poder asegurar al cliente bajo que parámetros ambientales la unidad va ser funcional. Específicamente las pruebas que se realizan son térmicas, condiciones de humedad y presión. Para la realización de las pruebas se utilizan diferentes equipos según sea la condición que se quiere probar, es por eso que se requiere un control y monitoreo muy estricto sobre los equipos y procedimientos que se ejecuten.

1.2 Definición del problema y su importancia

1.2.1 Descripción general del problema

La unidad RANSCO es un horno constituida por tres bandejas que giran a través de dos compartimentos por medio de un ascensor controlado neumáticamente.

Esta máquina somete las unidades a cambios bruscos de temperatura, de manera que se envejece el semiconductor, lo cual permite determinar algunos comportamientos de los microprocesadores a través del tiempo. Cada uno de los tres compartimentos se configura para obtener parámetros de temperaturas dentro de un rango específico. En este caso se tienen dos compartimentos calientes (200 °C máx.) y uno frío (-70 °C min.). Para ello se programa una temperatura mínima y una temperatura máxima dentro de cada uno de los compartimentos (“chambers”).

Funcionalmente, el equipo RANSCO es un complejo sistema electromecánico, controlado por un PLC marca *Allen Bradley* de la Familia SLC500. Específicamente un SLC 5/03 que cuenta con un módulo de entradas, un módulo de salidas a relé y dos módulos de entradas con termopares que le permiten al sistema retroalimentar su condición de temperatura en varios puntos del equipo y con ello controlar la operación del horno.

Para la configuración de cada una de las cámaras se requiere realizar una secuencia manual de introducción de datos a través de una pantalla de tacto “Touch Screen”, y luego se da inicio a las pruebas por medio de un botón de arranque y paro. En la figura 1.2 se muestra una boceto de cómo luce el equipo físicamente, según los descrito en los párrafos anteriores.

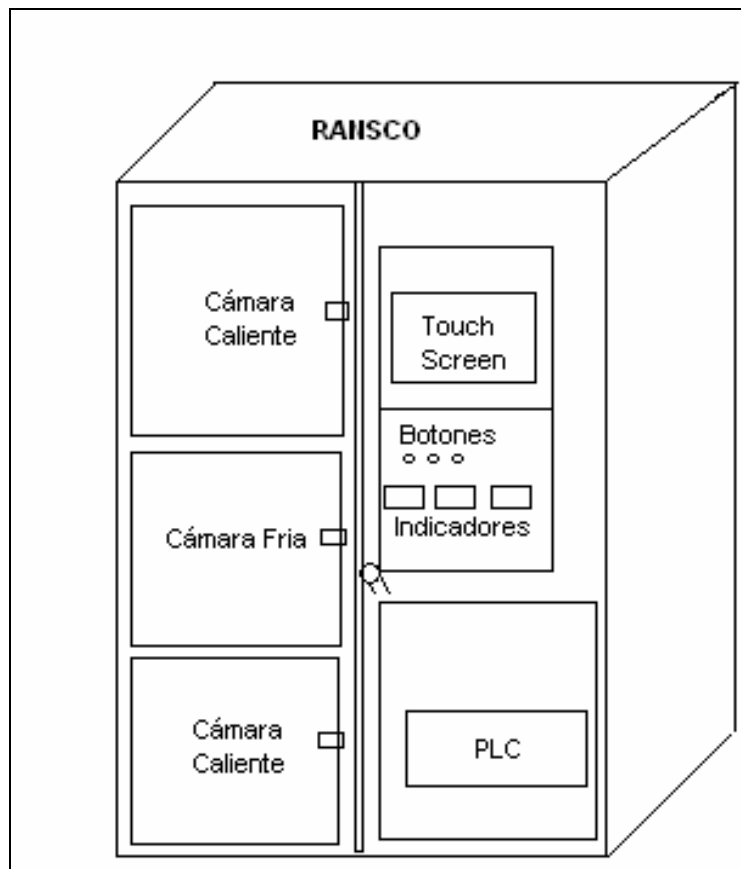


Figura 1.2 Vista de la cámara de choque térmico RANSCO

Para realizar las pruebas se tienen una serie de categorías las cuales no se pueden listar, pues son información clasificada. Las categorías se definen según los rangos de temperatura y el número de ciclos que las unidades van a estar rotando dentro de estos compartimentos.

Por el tipo y la cantidad de producto que se trata, se requiere que el procedimiento se realice sin errores, ya que los procesos de certificación dependen de la calidad en la ejecución de estas pruebas.

En general la problemática que se presenta es que no se puede tener un control estricto sobre las pruebas que se realizan en estas cámaras, lo cual podría provocar inconsistencias en el flujo del experimento, aunado a la descentralización de información que se sufre porque los equipos funcionan de forma independiente.

1.2.2 Descripción detallada del problema

Como ya se acotó en la sección 1.2.1, la necesidad surge porque no se tiene un control estricto de las condiciones bajo las cuales funcionan estos equipos (cámaras de choque térmico RANSCO), junto con una dispersión del proceso pues cada equipo funciona de manera independiente y no existe un control central. A ello se le une el inconveniente de que la carga de parámetros está sujeta a errores humanos de procedimiento. Esto implica que si las pruebas no se configuran dentro de alguna de las categorías o recetas definidas, el producto tiene que ser descartado y se tiene que iniciar la prueba nuevamente.

Cada una de las pruebas tarda como mínimo 75 horas y si llegara a darse un error en las mismas, la pérdida de material (unidades) y de tiempo es enorme (valorada en miles de dólares), además de que se levanta un proceso de investigación para determinar la causa de la falla, lo cual es muy costoso para la empresa en tiempo y dinero.

La situación descrita anteriormente se presenta porque se han detectado fallas o alta probabilidad de éstas en el proceso de configuración de parámetros de la prueba de choque térmico, debidos a que el sistema de introducción de parámetros de configuración es manual.

Otro inconveniente de la operación actual del equipo, es la poca versatilidad de su sistema de alarmas, ya que se requiere que exista una persona cerca del equipo para que lo opere. Esto produce recargo de labores para los técnicos que además tienen que atender otros equipos y actividades propias del laboratorio. Por expansión del laboratorio, la empresa necesita que los equipos sean monitoreables y controlables remotamente, por lo que se requiere una solución que satisfaga este propósito.

Con este proceso de centralización del control y monitoreo se necesita que la función de carga de parámetros previamente programada en el equipo por el

fabricante sean reprogramada, para satisfacer las nuevas necesidades de operación del equipo en el laboratorio.

Otra necesidad que se tiene es la de proveer a los ingenieros que planean los experimentos con información sobre la tendencia del equipo mientras este opera, así como mantenerlos informados de cualquier eventualidad que requiera de su participación, tal como alarmas, pruebas listas para ser trasladadas a otro procesos, entre otras.

Para resolver este problema la empresa tiene como propósito desarrollar un sistema que automatice el proceso de configuración de parámetros para la prueba, mediante la creación de un identificador único para las pruebas, el cual pueda ser cargado directamente en una PC que administre la configuración de los equipos de manera que no exista una introducción manual de la información requerida para las pruebas. El identificador se realizaría mediante un código de barras que puede ser leído e interpretado por alguna aplicación, una vez identificada la categoría por medio de una base de datos, se carga la información de la prueba y se evita la probabilidad de falla humana.

Se requería además implementar una red de equipos, para controlarlos de manera centralizada y obtener de ellos información sobre la tendencia del proceso, así como un sistema de alarmas interactivo que envíe mensajes a los responsables del equipo. Se requiere también que se generen reportes de la operación de los equipos conectados a la red antes mencionada.

En general, la solución que se plantea es un conjunto de aplicaciones que interactúen entre sí para superar los inconvenientes de centralización de la información y control de las pruebas.

1.3 Objetivos

1.3.1 Objetivo general

Aumentar la confiabilidad en el proceso de pruebas de choque térmico, mediante la automatización del proceso de carga de parámetros, monitoreo y control de los hornos de choque térmico RANSCO.

1.3.2 Objetivos específicos

- a. Implementar un mapa con las señales eléctricas que se requieren para el control del RANSCO.
- b. Diseñar una red de PLC que permita la interacción del servidor con cualquiera de los equipos RANSCO existentes en el laboratorio.
- c. Implementar una rutina que permita la comunicación entre el servidor y cualquier nodo de la red de PLC.
- d. Implementar una rutina que permita consultas a una base de datos de categoría de prueba por medio de una aplicación en el servidor.
- e. Implementar una rutina que genere una base datos del proceso por medio de los datos que le envía el PLC al sistema de monitoreo Rsvie32
- f. Implementar una rutina que permita la lectura de códigos de barra a través de un lector óptico conectado al servidor.
- g. Diseñar una interfaz de usuario que integre la lectura de los códigos de barra con la configuración de las pruebas por realizar.
- h. Implementar una rutina que reconstruya el gráfico de comportamiento de la prueba.
- i. Implementar una rutina en RsView 32 capaz de notificar alarmas.
- j. Implementar un programa que integre todas las rutinas anteriores en una sola aplicación computacional.

- k. Instalar el sistema completo
- l. Validar el funcionamiento del sistema
- m. Generar un manual de usuario el sistema.
- n. Generar un informe final del proyecto.

Capítulo 2

Antecedentes del proyecto

2.1 *Estudio del problema a resolver*

Al estudiar el problema que se quiere resolver, se deben tomar en cuenta varios aspectos relacionados con el entorno donde se realiza la actividad con los equipos. Estos aspectos van relacionados con los componentes, las variables y situaciones especiales que propician la situación descrita en la sección 1.2.2.

En primera instancia se listan los componentes del proceso en cuestión:

- a. Equipos: las cámaras de choque térmico en las cuales se cargan las pruebas.
- b. Interfaz usuario-máquina (HMI): es una pantalla de tacto que está conectada al control central del equipo, el PLC.
- c. Usuarios: los técnicos que tienen contacto con el material que se prueba y con el equipo que realiza las pruebas. Para cargar los parámetros de la prueba utilizan la pantalla de tacto, con la cual envían los parámetros al control central del equipo, en este caso un PLC.
- d. Modo de operación: es la secuencia de pasos que se siguen dentro de una ruta de prueba en un experimento.

En segundo lugar se tienen las variables relevantes del proceso de prueba, las cuales se describen a continuación:

Temperatura: en este caso existen dos valores extremos determinados por la receta que se esté aplicando, pues al ser el equipo un sistema de exposición a situaciones extremas, se requieren que existan tanto temperaturas altas (200°C máx.) como bajas (-70 °C min.) para poder generar la situación de choque térmico.

Tiempo de procesamiento: se refiere a la cantidad de tiempo en que el producto se verá sometido a situación de choque térmico. En este caso se mide en ciclos con periodos definidos por la categoría de la prueba.

El tercer punto que se trata, es el ambiente que rodea al problema. En este caso los equipos se encuentran en un laboratorio, en el cual se deben respetar normas de seguridad, por lo que la implementación tiene que estar bajo los estándares que se exigen para el funcionamiento de cualquier laboratorio en la empresa.

Una vez descritos los componentes que rodean al problema se procede a definir el mismo: en este caso lo que sucede es que no hay una herramienta que permita realizar un control y monitoreo estricto sobre las pruebas de choque térmico que ingresan al laboratorio. Eso implica que se deben hacer inspecciones visuales en el sitio para realizar el monitoreo de los equipos. Además, se deben generar peticiones de prueba por escrito y los reportes de capacidad se deben generar manualmente.

Con este proyecto lo que se persigue es implementar una herramienta que permita controlar y monitorear los equipos de manera remota, así como tener un sistema tecnológicamente adecuado para darle seguimiento a las pruebas y generar información de proceso que le de validez al mismo.

Ya que se tiene un panorama más amplio sobre la situación, se plantean los requerimientos que la empresa tiene, con respecto a lo que la ingeniería puede ofrecer para mejorar el proceso que se ejecuta.

2.2 *Requerimientos de la empresa*

En la actualidad la empresa cuenta con tres máquinas que realizan las pruebas, y está en sus planes instalar un cuarto horno. Estos equipos se llaman RANSCO OVEN. Cada uno de ellos opera independientemente, de manera que los técnicos llevan el material y cargan los parámetros manualmente. Después de esto inician la prueba y para llevar los indicadores del proceso efectúan inspecciones donde solamente se puede visualizar la cantidad de ciclos que se han ejecutado, así como la posición de las bandejas. Toda la información recopilada es utilizada para generar los reportes de capacidad y operación, los cuales se escriben por los operarios del sistema.

El principal requerimiento es que se integren todas las operaciones carga de parámetros, monitoreo de la prueba, control de alarmas y generación de reportes de operación, de manera que se automatice el proceso, además que se pueda hacer monitoreo remoto en tiempo real. Esto quiere decir que se sustituiría la operación manual por una automatizada.

En la sección 2.3 se plantea el esquema de solución que se siguió para resolver el problema, basado en los requerimientos de la empresa.

2.3 *Solución propuesta*

La solución que se planteó al problema se hizo siguiendo la secuencia de pasos que se describen a continuación:

- a. Análisis del problema e implicaciones técnicas de la solución.
- b. Análisis de la arquitectura eléctrica de los hornos para determinar los parámetros que se debían controlar para automatizar su configuración. Para ello se utilizaron los manuales y planos del equipo.
- c. Estudio de las distintas tecnologías de comunicación aplicadas en redes de PLC, para determinar que la topología de red utilizada fuera compatible con todos los equipos involucrados.

- d. Selección del hardware y software requeridos para realizar un diseño de acuerdo a las necesidades de la empresa.
- e. Diseño e implementación de una red de hornos conectados a un servidor mediante una red industrial basada en el protocolo DH-485.
- f. Diseño e implementación de un programa que convierta secuencias de texto en códigos de barras estándar, esto para generar las peticiones de prueba.
- g. Diseño e implementación de una estructura de datos para el manejo de la información relacionada con la prueba.
- h. Diseño e implementación de una interfaz de usuario para el sistema.
- i. Configuración del sistema de comunicación para lograr acceder al programa contenido por el PLC de los hornos.
- j. Estudio de la estructura del programa del PLC y generación de la base de datos de parámetros por controlar o monitorear.
- k. Implementación de un sistema de inclusión de pruebas, en cual el usuario ingrese los datos por medio de lectura de los mismos a través de un lector de código de barras. Los datos son adjuntos al material por medio de una hoja de petición
- l. Diseño e implementación de un sistema de carga de pruebas, basado en la lista de pruebas en espera, que se generó con el sistema de petición de pruebas.
- m. Diseño e implementación de módulo que permita el monitoreo en tiempo real de la prueba de manera que genere un gráfico para el comportamiento observado.
- n. Diseño e implementación de un módulo de notificación y control de alarmas presentadas en los equipos.
- o. Diseño e implementación de un sistema de emulación de la pantalla de tacto, para monitorear la posición de las bandejas dentro de la cámara.

- p. Diseño e Implementación de un sistema de reportes que pudiera monitorear la capacidad de los hornos así como el estado de las pruebas en curso.
- q. Definición de la estrategia de acceso remoto al sistema, siguiendo las políticas de privacidad que establece la empresa.
- r. Integración de la operación de los distintos módulos y su validación.
- s. Entrenamiento a los usuarios y administradores del sistema.
- t. Documentación de la solución

En la figura 2.1 se muestra un diagrama con los módulos más relevantes del sistema.

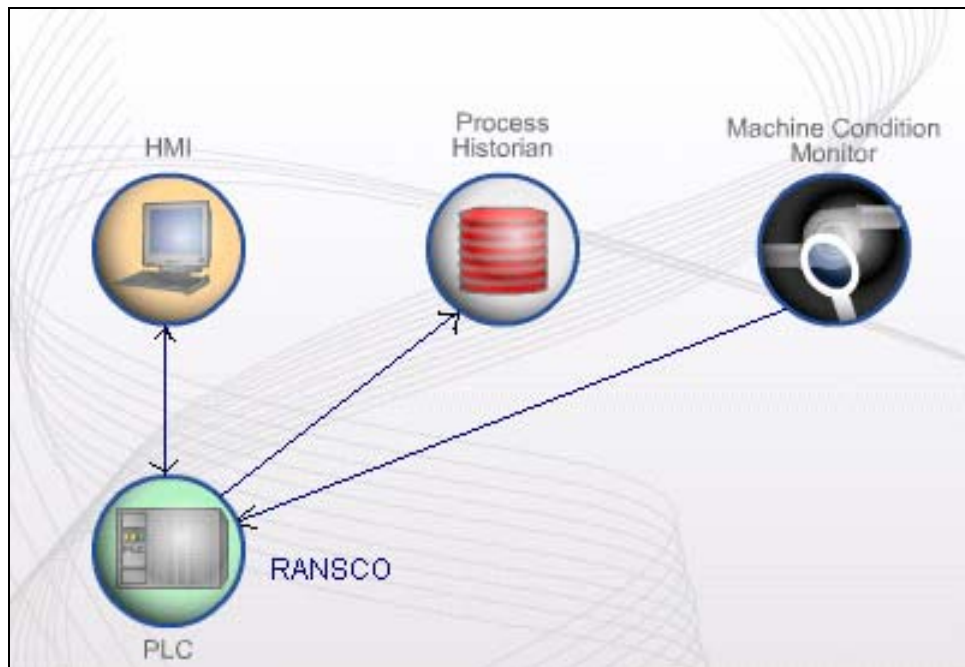


Figura 2.1 Diagrama de relación entre los módulos del proyecto

2.3.1 Red de equipos

La implementación de esta red de equipos responde a la necesidad de centralización de la información tanto para control como para monitoreo. En la figura 2.2, se muestra como se interconectaron los equipos RANSCO con el computador servidor en el cual se procesan las pruebas.

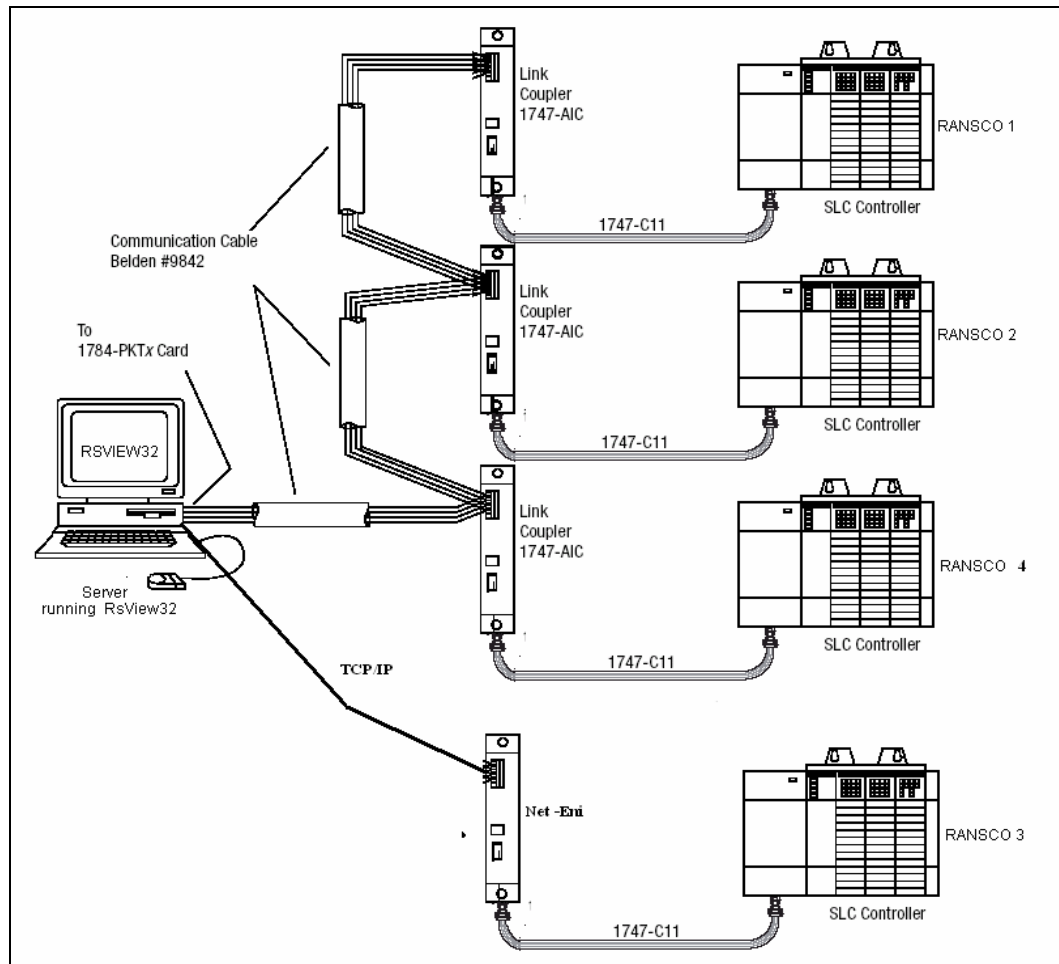


Figura 2.2 Arquitectura de la topología de la red de equipos RANSCO

Esta red está basada en el estándar industrial DH-485 y por las políticas de calidad de la empresa se requiere que la instalación del cableado se ajuste a los parámetros que se exigen en el estándar. Para la instalación de esta red, se requiere equipar la computadora servidor con una tarjeta de interfaz denominada PKTX, la cual permite que exista la compatibilidad con el protocolo de comunicación DH-485 y el software de control y monitoreo Rsvie32. Para la conexión de cada uno de los equipos a la red se requería la instalación de un dispositivo llamado acoplador de vínculo (AIC), que permite agregar nodos a la red.

En el anexo B.1 se muestra un diagrama de cableado tanto para la PKTX, como para el acoplador de vínculo (AIC)

En la implementación de esta red se hizo una modificación en la conexión de módulos que existía, pues anteriormente el módulo de interfaz con pantalla de tacto del RANSCO estaba conectado directamente al PLC de la máquina. En este diseño se decidió conectarlo al acoplador de vínculo, esto con el propósito de utilizar el canal de comunicación DH-485 que posee el PLC para comunicación con la red. La decisión se tomó porque el PLC que se estaba utilizando (SLC5/03) solo posee una interfaz DH-485 y se requería conexión tanto para la red como para la terminal de usuario.

Un aspecto importante de este diseño es que solo se requería un lector de código de barras para cargar las recetas de todos los equipos conectados a la red, ya que el proceso de recetas se encontraría centralizado en el servidor.

2.3.2 Selección de la receta

Para la selección de la receta se debió crear una base de datos en donde se guardara la información relacionada con la prueba. Para solicitar la prueba se creó un formulario, con el cual se presentaba la información, dicho formulario contenía la información codificada con código de barras. Esta información se procesaba con el lector de código de barras instalado y ésta sería guardada en una base de datos junto con el resto de las pruebas solicitadas. En la figura 2.3, se muestra como fue implementado el flujo de información entre las aplicaciones involucradas.

En la figura 2.3 se pueden destacar tres elementos principales, el primero es el administrador de datos (“driver”), el cual tiene la función de enlazar la información leída en el equipo con el software de control. El segundo elemento es la plataforma de desarrollo (“RsView 32”), que en este caso incluye a la base de datos de variables (Tag DB), la interfaz de usuario (“user Interface”) y el entorno de desarrollo en Visual Basic. El tercer elemento es la base de datos de proceso (“test tracking DB”) en la cual se guarda la información del proceso.

Cabe recalcar la compatibilidad que debía existir entre las aplicaciones, pues se requería que se pudiera realizar una interacción apropiada para poder enlazar la información entre los módulos de manera exitosa.

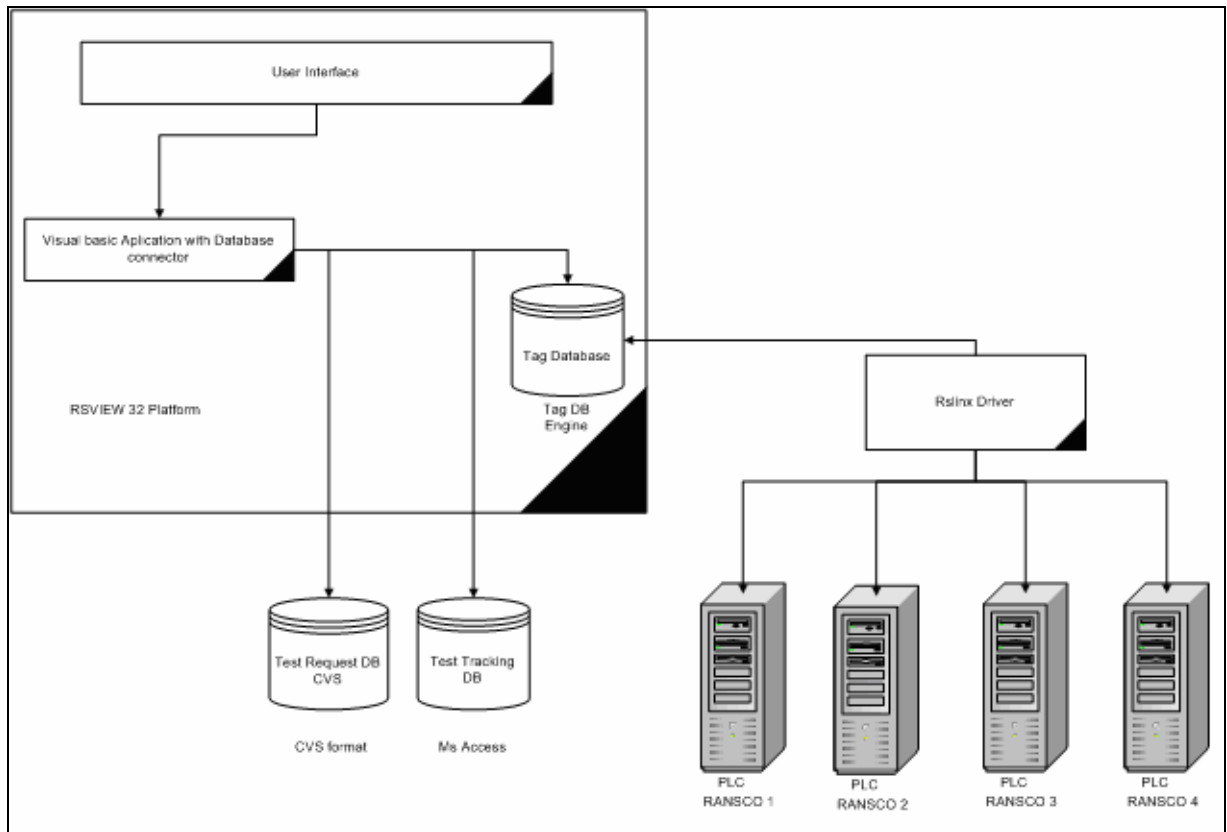


Figura 2.3 Flujo de información para la carga de parámetros

2.3.3 Codificación de las recetas

En la sección 2.3.2, se hablaba de un formulario para la solicitud de las pruebas por realizar en el equipo. Para generar dicho formulario se debió implementar una aplicación capaz de convertir cadenas de caracteres en códigos de barra estándar.

Se requería que la interfaz fuera muy amigable con el usuario que generaba las peticiones de prueba, para reducir al máximo las posibilidades de una equivocación en la generación de las mismas.

La información requerida para generar la formula con la codificación de la prueba fue tomada del sistema de manejo de pruebas de confiabilidad R2D2.

En la figura 2.4 se muestra cómo era la interfaz de usuario que se debía implementar para poder realizar la codificación de las cadenas de caracteres. Este programa debía ser capaz de poder exportar la codificación hasta un procesador de texto como MS Word, donde se llenaría el formulario con la información requerida.

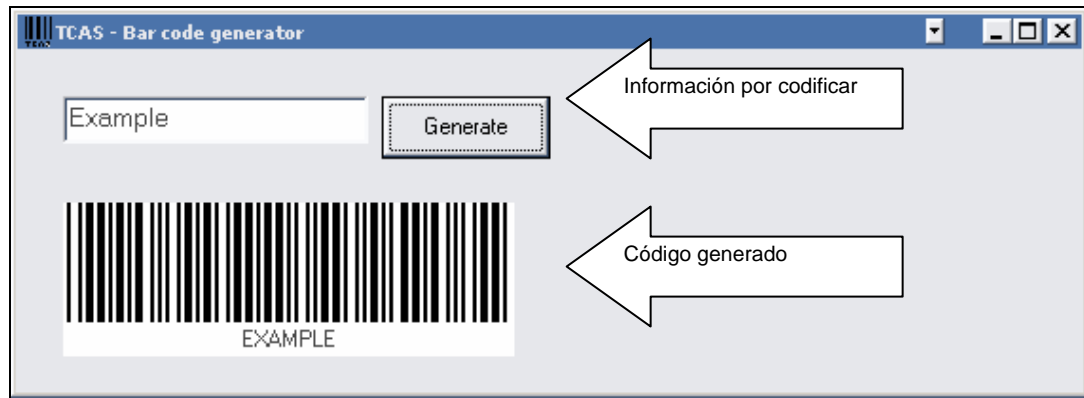


Figura 2.4 Interfaz de usuario para programa codificador de recetas

2.3.4 Notificación de alarmas

Otro de los bloques que constituye esta propuesta de solución es el sistema de notificación de alarmas, que debía tener la función de enviar mensajes cuando se presentara una anomalía en la operación normal del equipo. Además se debía tener la posibilidad de controlar el estado de las alarmas remotamente. Este módulo se implementaría por medio del software Rsviiew 32.

2.3.5 Registro de proceso

Uno de los aspectos relevantes que debía contener la solución implementada, era un módulo de registro de proceso en el cual se pudieran tener datos sobre la tendencia del proceso.

De esta manera se implementó una rutina para recolectar datos sobre el proceso, así como un módulo para generar gráficos de operación para los distintos equipos. Esto permitiría tener una visión comparativa del funcionamiento de los equipos, y además detectar problemas en la operación, mediante el análisis de los datos de operación recopilados por la herramienta.

Este objetivo se logró por medio del Software RsView, en éste se diseñó e implementó un módulo para monitorear las distintas cámaras que componen los equipos RANSCO.

En la figura 2.5 se muestra una pantalla de prototipo diseñada para visualizar la tendencia en un proceso dado.

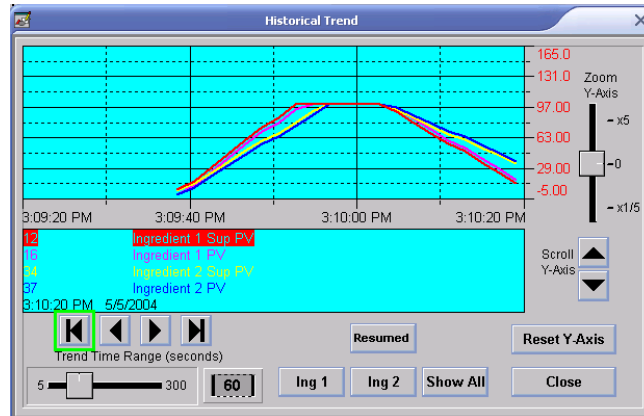


Figura 2.5 Ejemplo de interfaz para pantalla de historial de proceso

Según lo descrito en el capítulo 2, se requiere mucho desarrollo de software, especialmente aplicaciones con conexión a base de datos y ambientes gráficos. Al final lo que se logró fue un diseño que integrara un conjunto de opciones las cuales cumplen con lo que la empresa requería para satisfacer la necesidad planteada.

Capítulo 3

Procedimiento metodológico

Para alcanzar la solución planteada se debió realizar una secuencia metodológica que permitiera lograr cada uno de los objetivos planteados en la sección 1.3.

En primera instancia, se realizó un estudio detallado de los hornos, ya que se requería conocer de manera amplia el funcionamiento de los mismos para poder controlar los parámetros de configuración involucrados en la pruebas. De esta manera, se estudió del sistema eléctrico, el sistema térmico y el sistema de control (PLC). Esta actividad lo que preemitió sería obtener un plano de señales importantes para el control del horno.

Una vez que se tenía conocimiento de la arquitectura del horno, se procedió a estudiar las diferentes tecnologías utilizadas para implementación de redes de PLC. Basándose en un cuadro del manual del PLC, en el que se presentan las características propias para trabajo en entorno de red, se tomó la decisión sobre la tecnología a implementar. Luego de ello se cotizó el hardware y software respectivo para comenzar con la implementación.

Ya con los componentes, se realizó la construcción de la red. Primero se instalaría el hardware y, seguidamente, luego de probar la conexión, a nivel de enlace de datos, se instalaría el software requerido para obtener la interacción deseada con el servidor.

Para lograr una comunicación efectiva con cada uno de los nodos se debió realizar un esquema jerárquico de direccionamiento, así como el establecimiento de una metodología de configuración de la red. Una vez realizados estos pasos, ya era posible obtener un mapa de los dispositivos conectados a la red por medio del software *Rslinx*. Sin embargo en este momento, no existía la posibilidad de leer o escribir parámetros dentro del equipo, porque aun no se tenía una definición clara de

las estructura de datos a seguir. Por esta razón, se procedió a diseñar e implementar la estructura de datos que daría soporte al sistema. Para ello se desarrollaron dos etapas, la primera cubría el diseño a nivel de hardware, lo cual implicaba generar una base de datos donde se asociaran variables de software a parámetros físicos en el horno. Para lograr esta asociación se implementó lo que en la plataforma de desarrollo utilizada (*RsView*), se llama Tag Database. Una vez implementada esta base de datos (Tag Database), se procedió a diseñar la estructura de datos que manejaría la estructura de la prueba. Para ello se utilizó Microsoft Access, obteniendo de esta forma una base de datos del proceso, la cual administraría todos los datos relacionados con las pruebas realizadas en los hornos.

Ya con una estructura de datos sólida, se podían realizar rutinas de software capaces de manipular datos físicos del horno, como datos relacionados con la administración de las pruebas que se realizaban en los hornos. Con este concepto, se realizaron rutinas para escribir como leer información de los distintos equipos que conformaban la red.

Para cumplir con el requerimiento de minimizar la intervención del los usuarios del sistema en el ingreso de datos se diseñó una estrategia en la que los clientes generarían sus solicitudes de prueba con un software capaz de generar códigos de barras. Esto implicaba la instalación y configuración de un escáner de este tipo de códigos en el servidor para poder procesar las solicitudes giradas por los clientes del sistema.

Para conseguir el cumplimiento de este objetivo se diseñó un programa que generaba códigos de barras a partir de secuencias de caracteres dadas.

Ya con una interfaz de códigos de barras, se requería que la misma fuera compatible con las rutinas de manipulación de datos, para ello se instaló una biblioteca para que el escáner funcionara como emulador de teclado y así se integrara efectivamente al sistema.

Ya con la conclusión esta actividad, se tenía todo lo requerido para desarrollar rutinas de software que podían interactuar tanto con el usuario, como con los equipos. De allí en adelante se comenzó con el diseño e implementación de rutinas especificadas por los usuarios.

Primeramente se diseñó un módulo que permitía agregar pruebas a una lista de espera. En esta subrutina era donde se procesaban las solicitudes del cliente y se agregaban a la tabla de pruebas en espera ubicada en la base de datos de proceso.

El segundo módulo diseñado fue el de configuración del equipo. En éste, lo que se hacía era leer el identificador de la prueba por medio del lector de códigos de barra y la rutina lo asociaba a un equipo específico, luego cargaba los parámetros al equipo según la selección del tipo de prueba a realizar.

El tercer módulo lo que realizaba era un monitoreo de la operación del equipo. Para ello se presentaba una pantalla donde se reconstruía el proceso basándose en los datos recopilados por el sistema. Para este propósito se configuró una rutina de recolección de datos que trae embebida la plataforma de desarrollo utilizada.

El cuarto módulo permitía el control y notificación de las alarmas de los equipos a través de una interfaz de usuario que mostrara un sumario de la mismas y el estado actual. Para ello hubo que configurar una rutina que generara un historial dentro de un archivo. Este archivo se tendría que generar día a día.

El quinto módulo lo que mostraría sería un gráfico de cómo está constituida la red y permitiría observar una emulación de la pantalla de tacto que tienen los hornos. En ella se muestran parámetros relacionados con la configuración, así como la posición de las bandejas dentro del horno.

El sexto y último de los módulos que se implementó en esta versión del programa, fue el sistema de reportes. Este sistema generaría reportes conteniendo información sobre los experimentos que se ejecutan en los hornos.

Una vez que estos módulos estaban integrados en una sola aplicación, se procedió a instalarlos en el servidor y probar su funcionamiento por medio de la

emulación de solicitudes de prueba. Después de ello se comenzó con la validación, lo cual requería dejar el equipo funcionando y monitorear el estado de los mismos.

Una vez corregidos los errores de funcionamiento detectados y, finalizada la validación de la operación del sistema se procedió a mostrarlo a los usuarios y a entrenarlos en el funcionamiento y mantenimiento del mismo.

Al final se generó la documentación del sistema y se culminó haciendo una presentación del sistema a los clientes.

Capítulo 4

Descripción del hardware

En este capítulo se describen los componentes que tienen que ver con el control de la cámara, así como los que utilizaron para la implementación de la red de equipos.

4.1 Hardware de la cámara

Esta sección del capítulo se dedica a describir los distintos módulos de hardware que componen el control de la cámara de choque térmico RANSCO.

4.1.1 Controlador lógico programable SLC 5/03

El controlador lógico programable (PLC) que se encarga de controlar el funcionamiento de la cámara RANSCO, es uno de la familia SLC 500, específicamente el SLC 5/03, que incluye las siguientes características:

- Memoria para programa de 8K ó 16K
- Control para 4096 puertos de entrada o salida
- Programación en línea(incluye edición y pruebas del programa)
- 1 canal DH-485
- Un canal RS232 que soporta:
 - DF1 Full-Duplex para comunicación punto a punto, remotamente vía MODEM, o conexión directa a dispositivos de operación como programadores o terminales de usuario.
 - DF1 Half-Duplex Maestro / esclavo para SCADA (punto a multipunto)
 - DH-485 (sirve como un canal secundario). Use a 1761-NET-AIC con un 1747
 - CP3, 1761-CBL-AC00, o 1761-CBL-AP00 para conectar a la red DH-485

En la figura 4.1 se muestra un diagrama del controlador lógico programable, junto con una vista frontal del mismo.

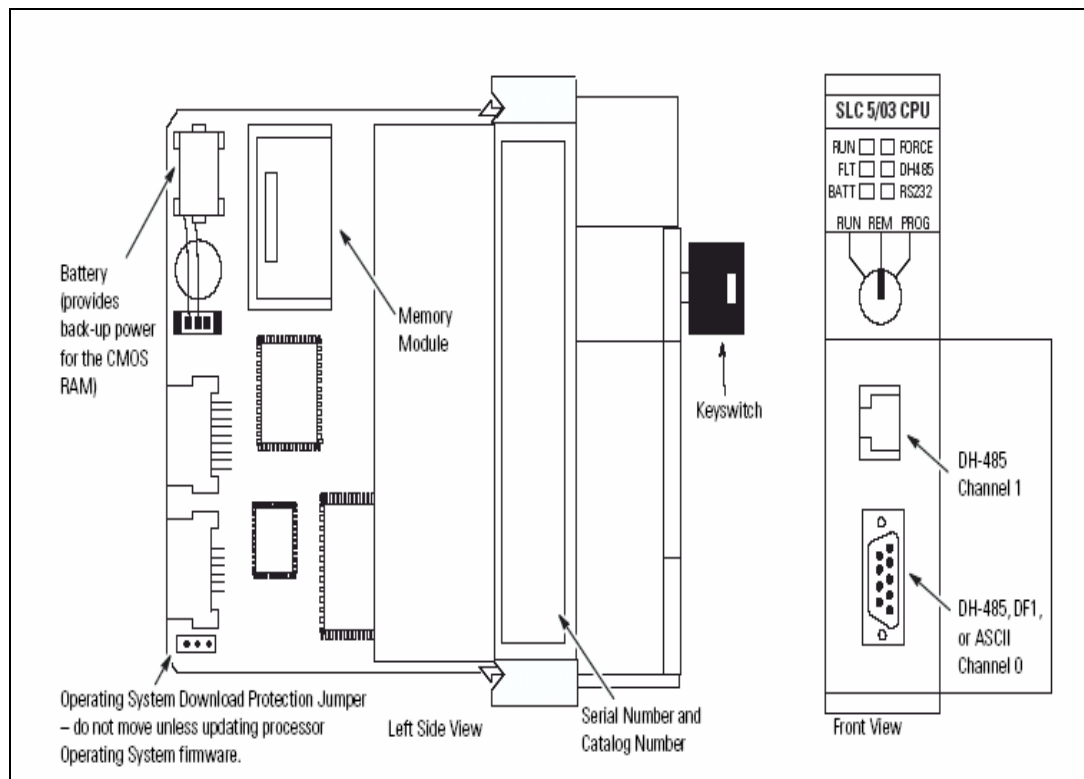


Figura 4.1 Diagrama del procesador del PLC

4.1.2 Tarjetas de expansión familia SLC 500

El control de la cámara de choque térmico RANSCO cuenta con tarjetas para interfaz con los sistemas eléctricos y mecánicos que controla, para ello se instalaron las siguientes tarjetas de interfaz: entradas analógicas (1746-NI16I), entradas digitales (1746-IA16) y salidas digitales (1746-OA16).

4.1.3 Tarjeta de entradas analógicas 1746-NI16I

Esta tarjeta tiene capacidad para 16 señales analógicas, las cuales deben variar entre los ± 20 mA. Además cuenta con una palabra de configuración para cada canal, por medio del cual se elige el rango de variación, filtro de ruido y formato del dato. Las palabras de configuración se encuentran en las direcciones O:e.0 a la O:e.15, donde e es el slot donde se ubique la tarjeta, los datos se encuentran en I:e.0

a l:e.15; cada palabra es de 16 bits, lo que permite tener una resolución de 1 μ A/paso, es decir, si aumenta un microamperio el dato aumentará en uno.

La tarjeta cuenta con un terminal para cada entrada y dos comunes, como se muestra en la figura 4.2.

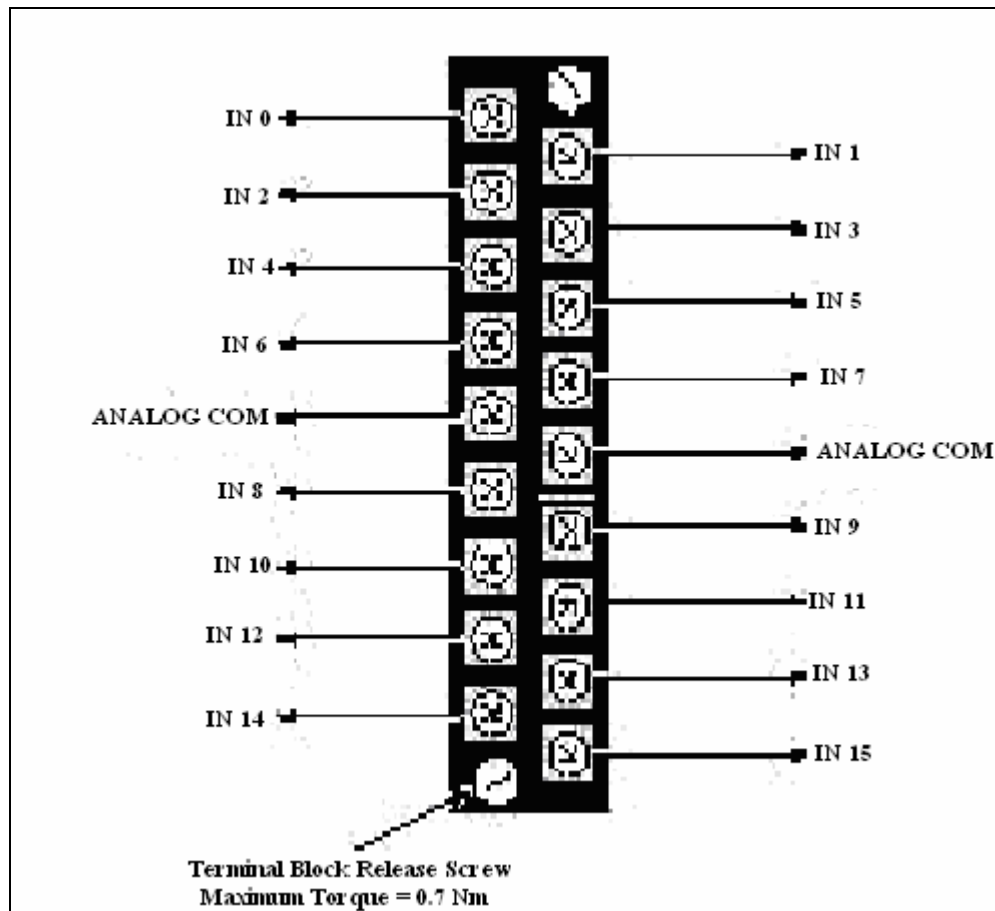


Figura 4.2 Diagrama de terminales tarjeta 1746-NI16I

Esta tarjeta se conecta poniendo los terminales comunes a 0 Vdc de la fuente, mientras que las entradas van a los sensores que le correspondan, como se muestra en la figura 4.3.

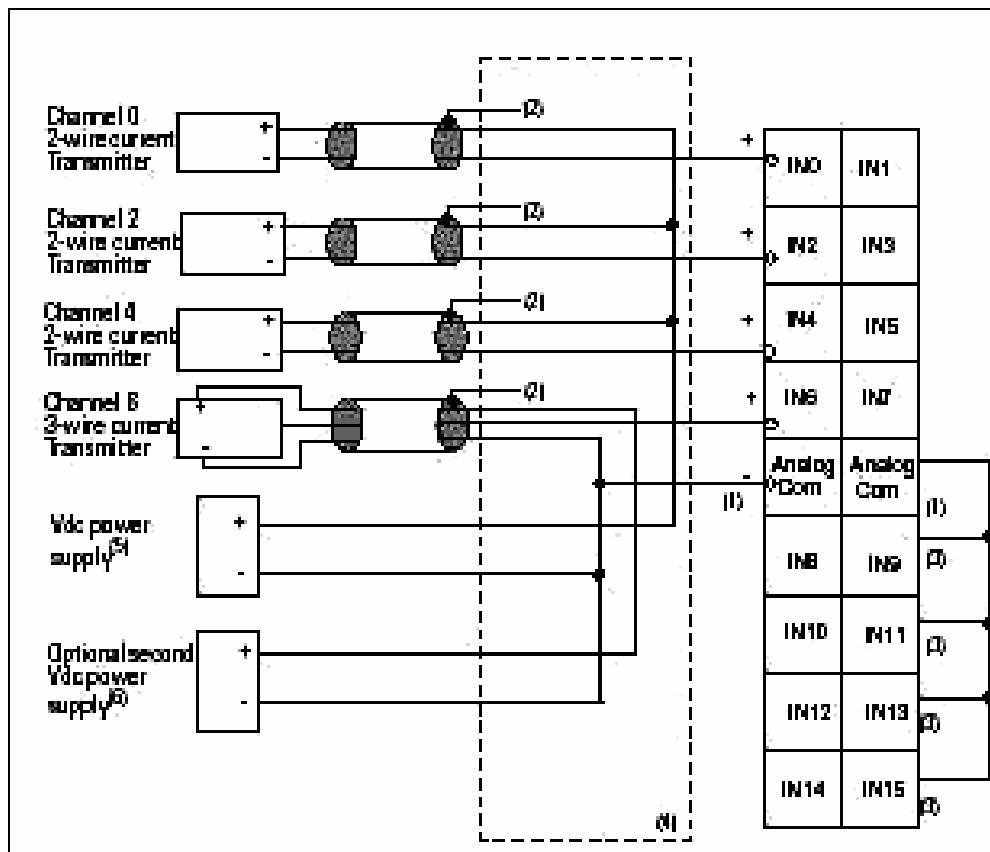


Figura 4.3 Diagrama de conexión tarjeta 1746-NI161

4.1.4 Tarjeta de entradas digitales 1746-IA16

Esta tarjeta permite tener 16 entradas digitales; estas entradas digitales pueden presentar dos estados: activado, que sería una señal de 110 VAC y desactivado, que sería 0 VAC.

La tarjeta no requiere ninguna programación adicional para su configuración y se pueden acceder los datos de entradas en las direcciones de la I:e/0 a la I:e/15, donde “e” es la ranura donde se ubica la tarjeta.

Para la conexión de la tarjeta la misma cuenta con un terminal llamado “AC COM”, el cual se conecta al neutro de la red, mientras que la línea de 110 VAC es llevada a la entrada por el dispositivo que se le conecta a la misma, y de esta forma se cierra el circuito, como se muestra en la figura 4.4.

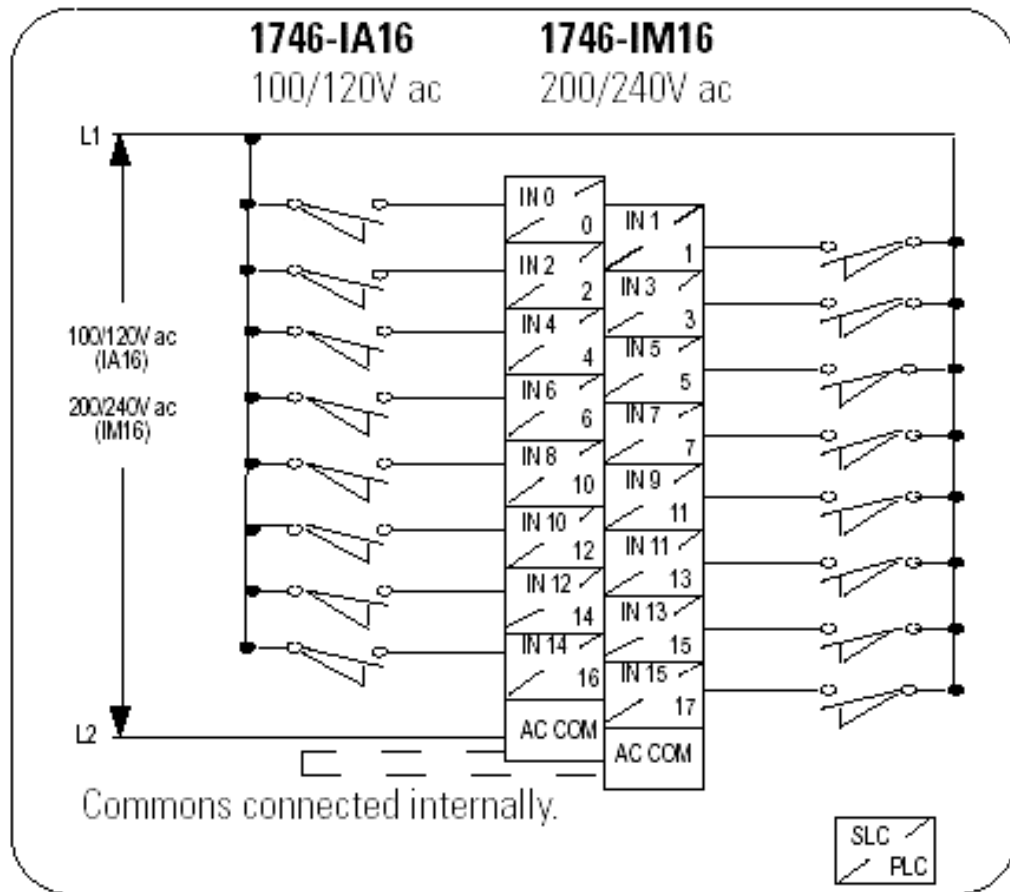


Figura 4.4 Diagrama de conexión tarjeta de entradas digitales 1746-IA16

4.1.5 Tarjeta de salidas digitales 1746-OA16

Esta tarjeta se encarga de proporcionar 16 salidas digitales, las cuales pueden utilizarse para accionar diferentes tipos de relé; la señal que genera cada salida es de 110 VAC cuando se encuentra energizada.

Para acceder cada salida en el programa, estas se ubican en una sola palabra de 16 bits, que se direcciona como O:e/bit, donde “e” es la ranura donde se ubica la tarjeta y “bit” corresponde al número de la salida que va desde 0 al 15.

En la figura 4.5 se muestra el diagrama de conexión de este módulo, donde los terminales llamados “VAC 1” y “VAC 2” deben conectarse a la línea del 110 Vac. Las demás terminales corresponden a las señales de salida del módulo.

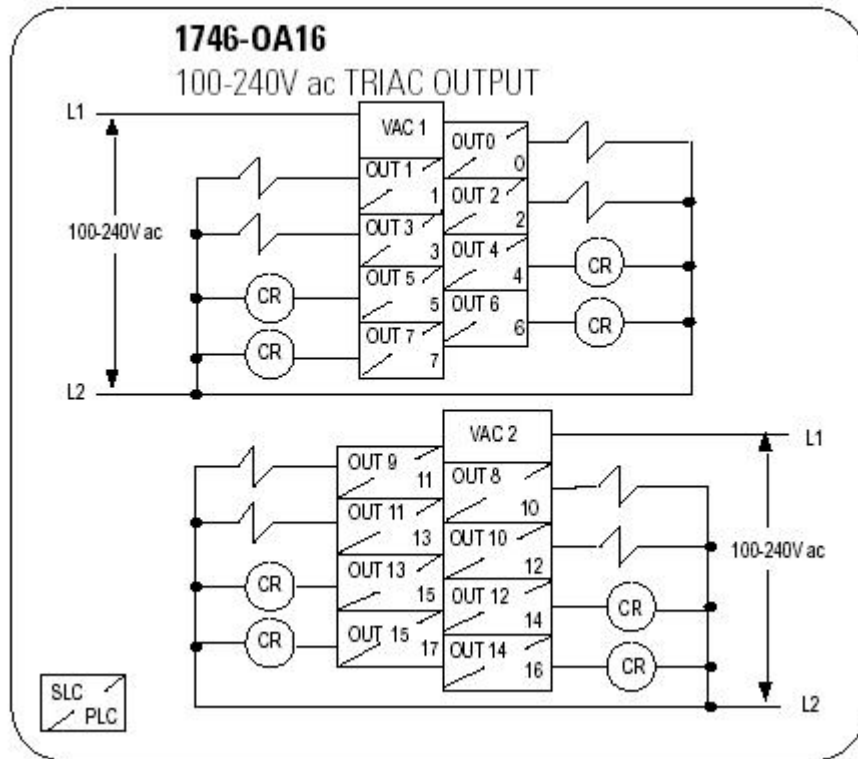


Figura 4.5 Diagrama de conexión tarjeta de salidas digitales 1746-OA16

4.2 Instrumentación

En el funcionamiento de la máquina RANSCO se ven involucradas varios tipos de variables, entre las cuales están temperatura, presión, posición y tiempo. La principal de ellas es temperatura, por lo cual se ha dedicado una sección a describir la manera en que la misma será interpretada por el sistema de control.

4.2.1 Transductores de temperatura

Para la medición de temperatura se utilizan termopares que ya se encuentran instaladas dentro del equipo, en los alimentadores, recolectores y puntos estratégicos de la cámara. Para ello se usan termopares tipo K.

Los termopares generan señales que varían de 0 a 100 mV, por lo que usa un acondicionador de señales para convertirla en una señal de 4 a 20 mA.

En el apéndice A.2 se describe su distribución dentro del equipo.

4.3 Hardware de la red de equipos

Para implementar la red en cuestión, se utilizaron dos elementos de hardware, el primero: un acoplador de vínculo (1747-AIC), y luego una tarjeta de interfaz de red (PKTX) para conectar el servidor a la red.

4.3.1 Acoplador de vínculo 1747-AIC

El acoplador de vínculo 1747-AIC, es una unidad que se utiliza para agregar nodos a una red DH-485, para ello se conecta con el PLC. El AIC posee tres interfaces, una de ellas se usa para conexión con el PLC, la segunda para la troncal de la red y la tercera para conectar a una interfaz humano-máquina. En la figura 4.6 se muestra un diagrama de conexión del AIC.

En condiciones normales de funcionamiento con el controlador programable conectado al acoplador de vínculo (1747-AIC), el procesador proporciona alimentación eléctrica tanto al acoplador de vínculo como al dispositivo periférico (DTAM, PIC, HHT), si está conectado, a través del cable 1747-C11. No es necesaria una fuente de alimentación eléctrica externa.

Si no se conecta el procesador al acoplador de vínculo, hay que utilizar una fuente de alimentación de 24 VCC de clase 2 para la alimentación del acoplador de vínculo y del dispositivo periférico.

Conecte siempre el terminal CHS GND (tierra de chasis) a la toma de tierra más cercana. Esta conexión es necesaria independientemente de que se utilice o no una fuente de alimentación eléctrica de 24 VCC.

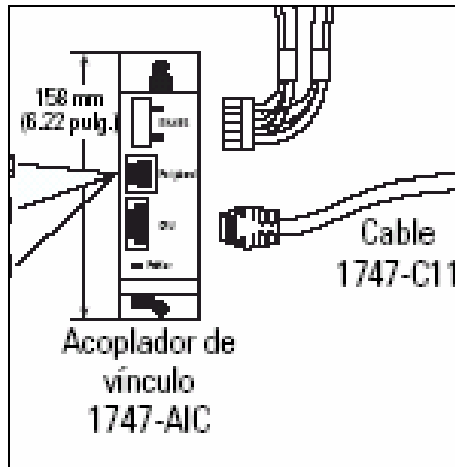


Figura 4.6 Diagrama de conexión para el 1747-AIC

4.3.2 Tarjeta de red PKTX

La tarjeta 1784-PKTX se utiliza para conectar computadoras personales (PC) a controladores lógicos programables (PLC) mediante los protocolos DH+ y DH-485. Esta herramienta facilita la creación de redes de PLC con computadoras, lo cual permite fácil programación y adquisición de datos. Esta tarjeta posee una interfaz PCI (Peripheral Component Interconnect) para adaptarla a la PC. Además, de una interfaz para redes DH+ y una para DH-485. Para activarla se requiere instalar un driver en el sistema operativo, y para administrarla se utiliza el software RsLinx, el cual permite configurar la velocidad, protocolo, dirección de nodo, entre otras.

Entre las características relevantes de este tipo de protocolo se encuentra:

- a. La longitud máxima de la línea troncal es de 4000 pies.
- b. El tipo de medio es cable trenzado tipo Belden 9842.
- c. Tasa de transmisión máxima de 19200 bps
- d. Método de acceso al medio basado en un “token pass”

Capítulo 5

Descripción del software

Para el desarrollo del proyecto se utilizaron distintos paquetes de software de Rockwell Automation. El programa del PLC que controla el RANSCO se obtuvo con RSLogix 500 v5.50. Para las redes de comunicación entre PLC y el servidor se usó el RSLinx v40.1 y se programó la interfaz con los usuarios por medio del Rsvi32 v6.50. Para la obtención de los parámetros de configuración del PLC se utilizó el panelBuilder 32.

5.1 RSLinx

Este programa se encarga de la comunicación de los PLC con la computadora. En este caso se utiliza una red DH-485; para configurar la red, se debe establecer el dispositivo que utilizará la computadora para la comunicación; una vez se haya configurada este dispositivo, el RSLinx establece la comunicación con el PLC, la cual se mostrará en la pantalla principal del programa, representada por icono del PLC con su nombre respectivo, como se muestra en la figura 5.1.

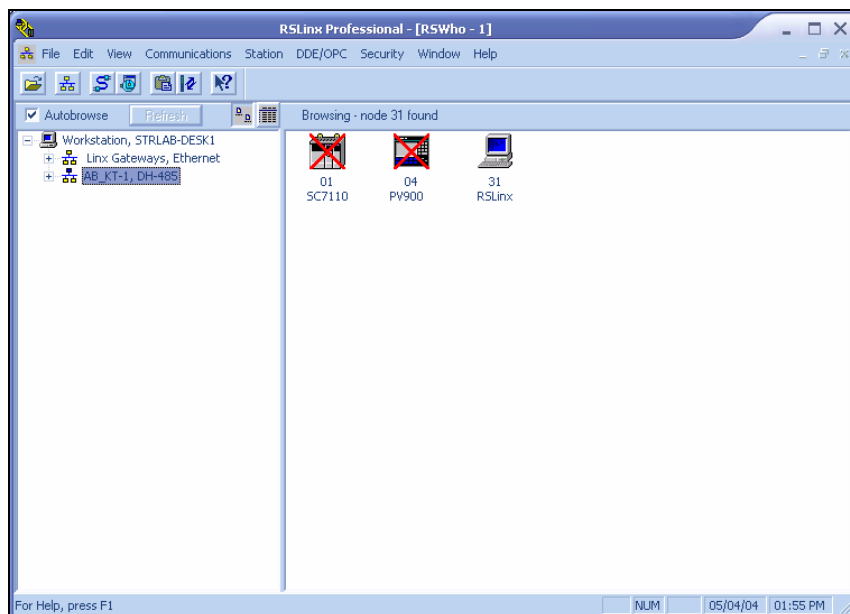


Figura 5.1 Pantalla principal RSLinx

Este software permite conectar la computadora a toda clase de redes de PLC, todo depende del dispositivo con que cuente la computadora para conectarse a la red de PLC. Para ello se configura el manejador de conexión (“driver”) según sea el medio físico y el protocolo sobre el cual se diseñó la red. En la figura 5.2 se muestra la pantalla de configuración del manejador de la red usado en esta aplicación.

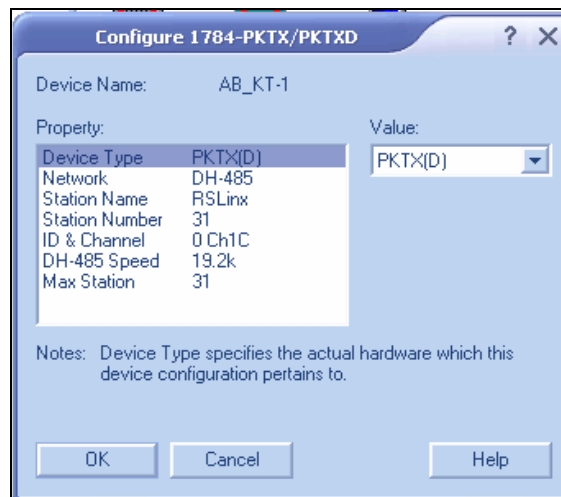


Figura 5.2 Pantalla de configuración de interfaz de red

5.2 RSLogix 500

Este software permite diseñar el programa en lógica de escalera usado en la familia de PLC's SLC 500. Para empezar a desarrollar el software se debe crear un nuevo proyecto, esto se hace por medio del menú principal, cuando se crea un nuevo proyecto se elige el procesador que se usará, así como el tipo de gabinete que se utiliza, como se muestra en la figura 5.3.

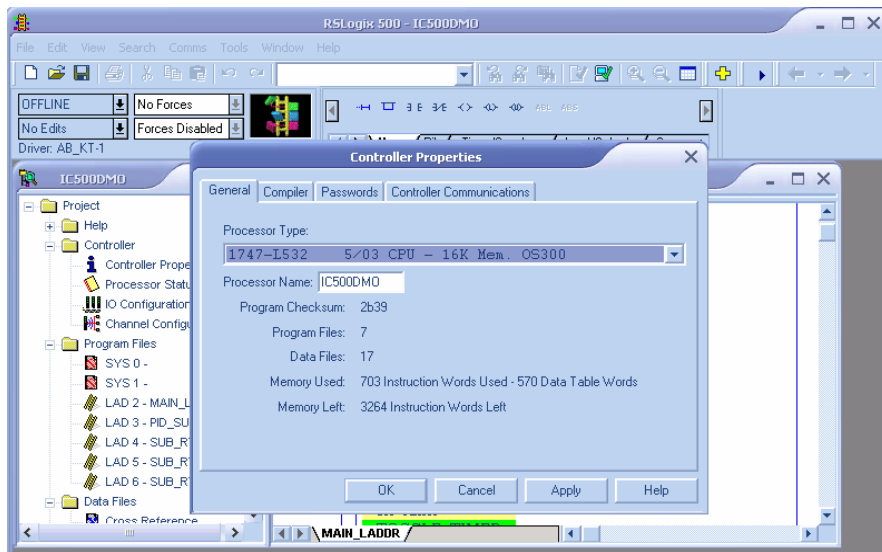


Figura 5.3 Pantalla de Rslogix500 utilizada para crear un proyecto

Una vez creado el proyecto en la pantalla principal del programa aparece un icono que llamado “IO Configuration”, que permite colocar las tarjetas con las que cuenta el gabinete, las tarjetas se seleccionan por su código y se insertan en la ranura correspondiente en el gabinete.

La figura 5.4 muestra la configuración de las tarjetas de entrada/salida.

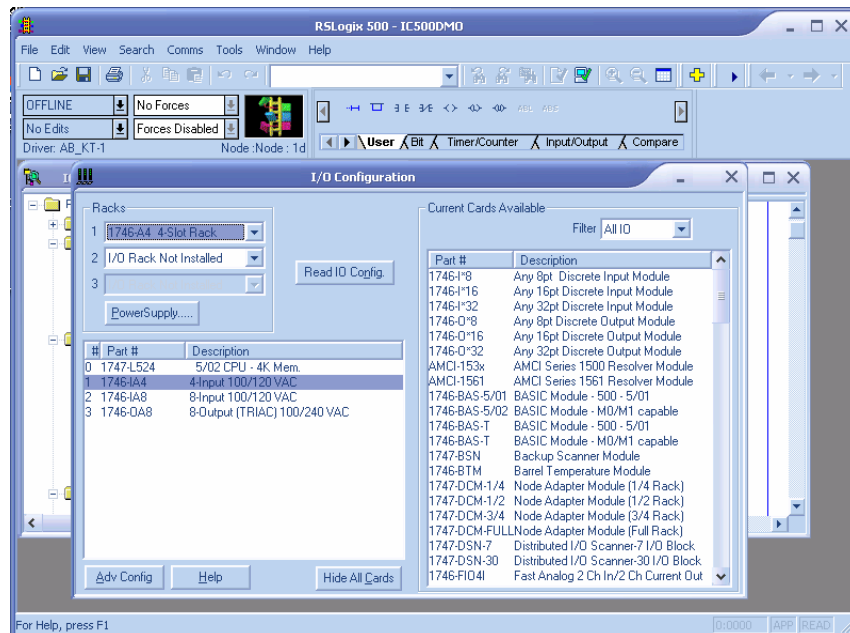


Figura 5.4 Pantalla de configuración de los dispositivos externos del PLC

Una vez configurado el PLC que se utilizará, se diseña el programa en escalera, el cual se divide en “ladders”. Las variables del programa se observan en la carpeta llamada “Data Files” donde se encuentran los diferentes tipo de variables que se pueden utilizar, las que pueden ser enteros, flotante, binarios o contadores, entre otros.

5.3 Panel Builder 32

Este software se utiliza para crear las interfaces con los usuarios por medio de las pantallas de tacto “Touch screen” PV900 y PV1000 de Allen Bradley. La figura 5.5 muestra la pantalla principal de este software.

En este proyecto el software se utilizó para adquirir una lista con los nombres y direcciones de las variables utilizadas por el PLC para su HMI, ya que el programa no ofrecía documentación al respecto y es parte de la información que el fabricante del equipo no provee a los clientes.

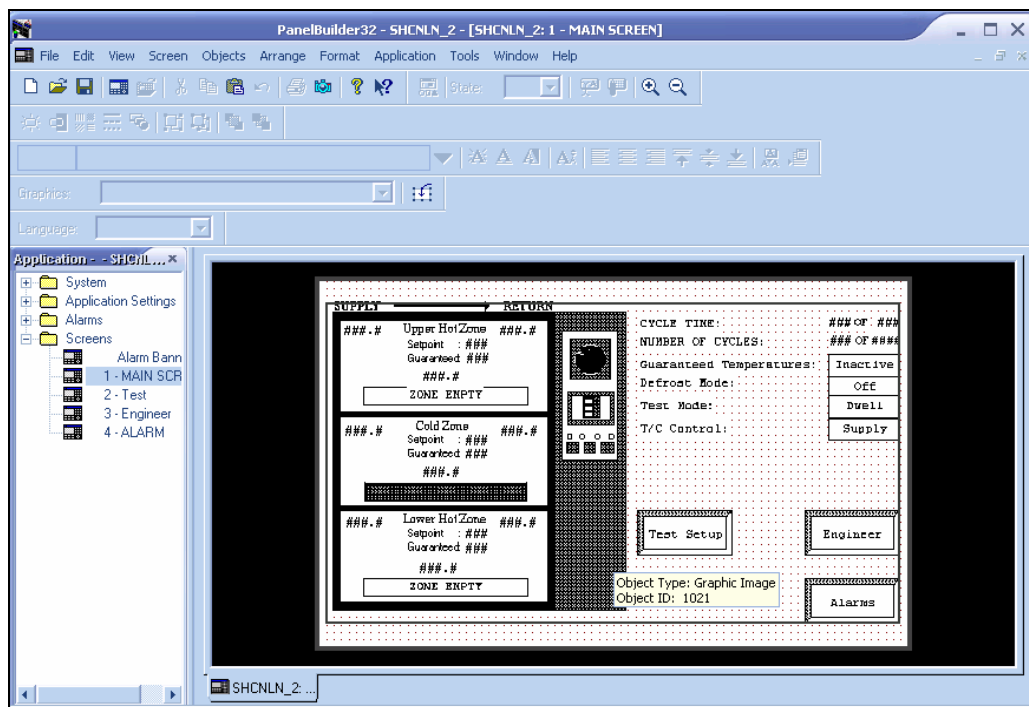


Figura 5.5 Pantalla diseñada con Panel Builder 32

5.3 Rsview32

Este software se utiliza para diseñar la interfaz del sistema con el operario. Cuando se crea proyecto se observan en las pantallas las carpetas que lo conforman. En la figura 5.6 se observa el menú principal.

En esta pantalla se pueden acceder las carpetas del programa que son:

- a. System: utilidades relacionadas con el sistema, ya sean de comunicación, direccionamiento, variables, muestreo de variables entre otras.
- b. Graphics: contiene la biblioteca de gráficos, así como las pantallas que diseña el programador para interfaz con el usuario.
- c. Alarms: la información de la configuración de las alarmas con que cuenta el sistema.
- d. Data Log: contiene la información de la configuración del módulo de recolección de información (“dataloging”)
- e. Logic and control: se encarga de la interfaz con la plataforma de programación en Visual Basic, así como la configuración de macros y tags derivados.
- f. Messenger: permite la configuración de tareas informativas vía correo electrónico o radio localizador del estado del sistema.
- g. Recipe Pro: módulo para el manejo avanzado de carga de recetas.
- h. SPC: módulo especializado para control estadístico de procesos.

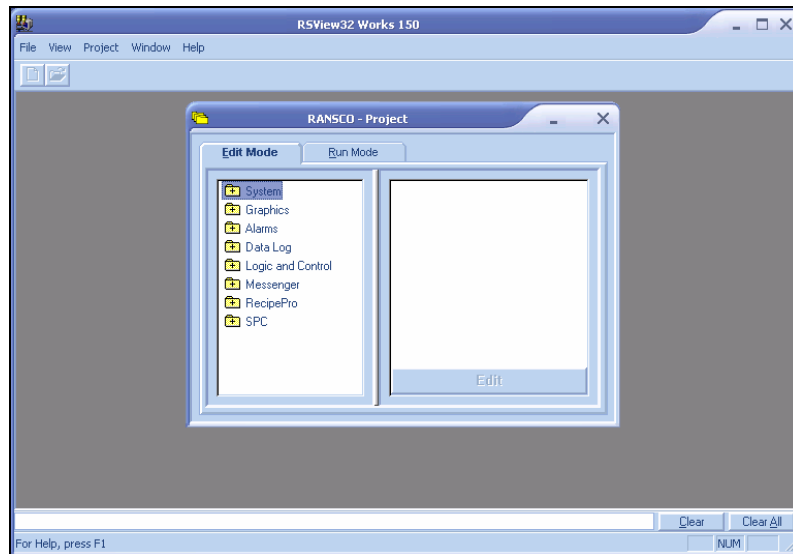


Figura 5.6 Pantalla principal Rsvie32 Works

En primera instancia, lo que se realiza es la configuración de la comunicación, en donde se elige el canal, la dirección del nodo, las variables involucradas y todo lo relativo a la operación del sistema. Después de esto se diseñan las pantallas. En la figura 5.7 se muestra el diseño para la pantalla principal del sistema. Luego de esto se asocian los tags a cada una de las pantallas diseñadas y se realizan pruebas monitoreando el funcionamiento del equipo en línea para observar correspondencia en los datos enviados por el usuario y los datos recibidos por el equipo.

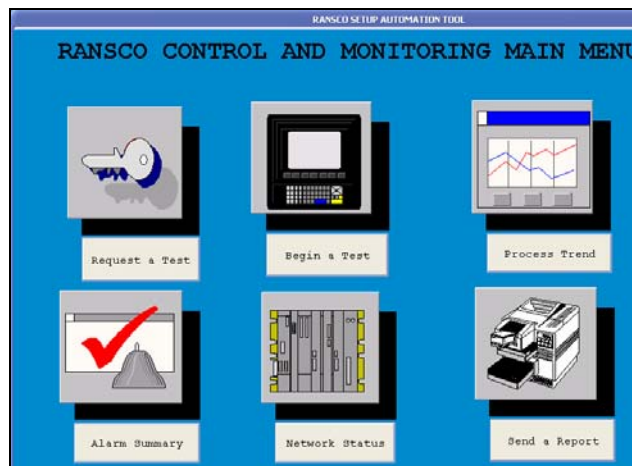


Figura 5.7 Pantalla diseñada con RsView 32

Capítulo 6

Análisis y resultados

6.1 *Diseño del hardware*

Con respecto al hardware lo que se hizo fue implementar la topología descrita en la sección 2.3.1 Para ello se instalaron los acopladores de vínculo (1747-AIC) en cada uno de los equipos y se conectaron a los PLC a través de un cable serial 1747-C11. Después, las terminales de pantalla de tacto fueron conectadas a cada uno de los acopladores de vínculo. Se comprobó funcionalidad. Sin embargo, antes de esto, se debió reconfigurar el número de nodo de PLC al cual le debían dar el apoyo, pues por defecto las pantallas no estaban configuradas para trabajar en un entorno de red.

La red implementada es de tipo bus, lo cual requiere que al nodo que se añada, se coloque un acoplador de vínculo, para aislarlo de la red cuando el equipo no funcione adecuadamente y no afectar la operación y desempeño de toda la red.

Se realizaron pruebas para valorar la conectividad de la red que consistían en apagar alguno de los equipos para observar el efecto en el rendimiento, y se valoró el rendimiento para ver si dependía del parámetro de “time out” configurado en el Rsview para el canal de comunicación, sin embargo, consultando el manual de Rsview, se determinó que con un tiempo de espera (“time out”) de 3 segundos, la red tendría un tiempo de espera aceptable, y que éste no afectaría el rendimiento global de la red.

La metodología de comunicación utilizada por los equipos se denomina “token pass” y consiste en un paquete denominado “token”, el cual circula por la red y permite que cada uno de los nodos tenga actividad de transmisión como de recepción.

El protocolo DH-485 se transmite por medio de cobre en un par trenzado, y una línea común. Algunas implementaciones requieren de resistencia terminal. Sin embargo, en este caso no, por eso se puentean esas terminales según lo que

especifica el fabricante de los acopladores de vínculo. En el anexo B.1 se muestra el esquema de cableado seguido para la implementación de la red.

Respecto a la interfaz con la PC que alberga la aplicación de monitoreo y control, se requirió instalar una tarjeta de red modelo PKTX de Allen Bradley, la cual posee un canal de comunicación DH-485 y por medio del software RsLinx se configura para obtener una comunicación efectiva con los equipos.

Como resultado de la implementación del modelo propuesto para la red se obtuvo una comunicación eficaz, siguiendo el esquema de direccionamiento mostrado en la tabla A.6 de los apéndices.

La tarjeta utilizada permite que el protocolo DH-485 administre 32 dispositivos conectados a la red, lo cuales pueden ser PLC's, PC's, pantallas de tacto, o cualquier dispositivo compatible con el protocolo. La administración de la red se hace por medio del Rslinx, ya que es en este software donde se configuran los parámetros de operación de la tarjeta de red PKTX.

Además de la implementación de la red, a nivel de hardware se instaló y configuró un lector de código de barras para realizar las lecturas de las hojas de petición para las pruebas. Este lector se instaló en el servidor por medio de un sistema que lo multiplexa con la señal de teclado, por lo que ambos pueden funcionar de manera simultánea.

Se debió instalar una biblioteca de soporte para que el lector funcionara de manera adecuada. Ya con toda la plataforma en hardware funcionando, se procedió a diseñar el software.

6.2 *Diseño de la estructura de datos*

Después de implementada y analizada la estructura de hardware, se comenzó con el diseño del software; sin embargo, como se muestra en la figura 2.3, se requería antes de implementar la aplicación que existiera una estructura de datos robusta para dar soporte al sistema.

Esa estructura de datos se dividió en tres elementos principales. El primero de ellos, fue la base de datos de variables en el equipo. La estructura de esta base de datos ya estaba contenida dentro de la plataforma de desarrollo en RsView 32; sin embargo, la información referente a las variables por monitorear y controlar debía ser cargada manualmente, ya que el PLC por su programación de fábrica no tenía la biblioteca con las variables de proceso.

Para conseguir la lista de variables por controlar y monitorear se debió recurrir al software de la interfaz manual, de manera que accediendo al programa de las pantallas, se pudo extraer toda la información requerida para completar la base de datos de variables en los equipos.

Gracias a esta información se podían leer y escribir parámetros dentro de los equipos. Esta base de datos de variables era primordial para el desarrollo de la aplicación, ya que sin ella era imposible acceder al control de los equipos, en este caso el PLC de cada uno de los nodos. En el apéndice A.3 se muestra un listado de de las variables de proceso extraídas de la pantalla de tacto.

Las variables fueron organizadas según su ubicación en los equipos, pues esto permitía la facilidad de ubicación a la hora de depurar el programa y un mejor orden de la información dentro de la base de datos.

La segunda estructura de datos diseñada para dar soporte al sistema, fue la base de datos de pruebas solicitadas, la cual fue implementada en un formato de datos sobre un archivo separado por comas (CVS) por su facilidad de manejo dentro de plataforma de desarrollo, ya que no requería de un motor de base de datos, sino que utilizaban rutinas sencillas de software.

Finalmente se diseñó e implementó una base de datos para guardar información sobre las pruebas. Dicha base de datos permitía llevar un control sobre la ejecución de las pruebas.

La base de datos de proceso fue desarrollada siguiendo un esquema jerárquico en el cual crearon e interlizaron cuatro tablas las cuales en conjunto formaban la estructura de datos que daban soporte a las pruebas en ejecución. En la figura 6.1 se muestra el diagrama relacional con el cual se implementó la base de datos para manejo y seguimiento de los experimentos.

La implementación práctica de la base de datos se realizó con Microsoft Access, y la información se administraba por medio de un componente de acceso a base de datos (DAO), con el cual fue posible enviar información desde la aplicación en Visual Basic. Por medio del componente mencionado, se realizaban las consultas para generar los reportes y otras operaciones que requerían consultar información de la base de datos del proceso. Una vez diseñada y probada la estructura de datos se procedió a diseñar e implementar la aplicación requerida para poder interactuar con lo equipos de manera amigable.

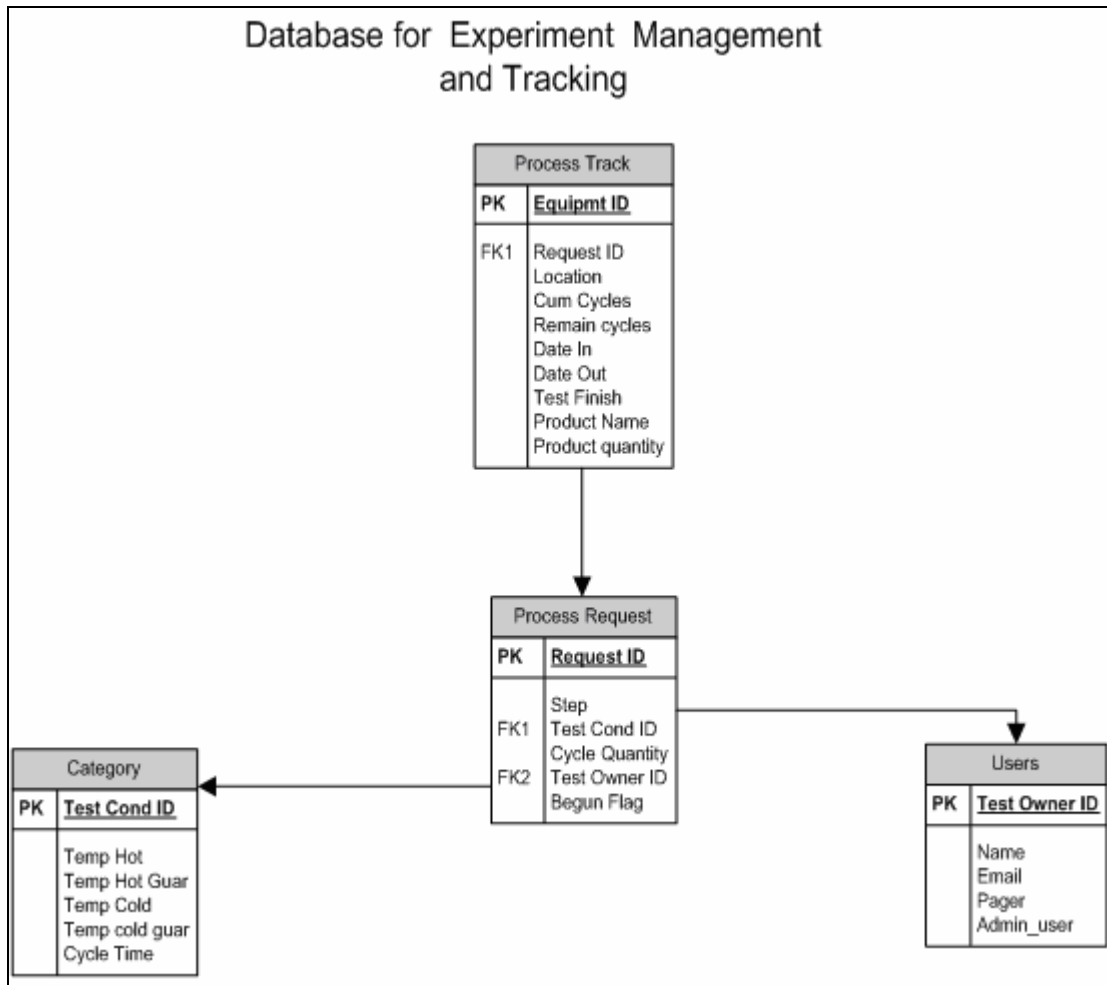


Figura 6.1 Diagrama del diseño de la base datos para proceso

6.3 Diseño del software

Ya con una estructura de hardware y de datos consolidada, era posible iniciar con el diseño de la aplicación que utilizarían los usuarios para controlar y monitorear los equipos. El diseño de software se realizó por módulos, para facilitar el proceso de prueba, y con ello ir integrando cada uno de los módulos a la plataforma principal.

En los siguientes apartados se discute la implementación de cada uno de los módulos que comprenden la plataforma.

6.3.1 Módulo de carga de solicitud de prueba

Se diseñó un módulo para que los clientes del laboratorio pudieran hacer peticiones de prueba de choque térmico de manera remota, para ello realizan una conexión de escritorio remoto al servidor y ejecutan una aplicación que se diseñó en VBA (Visual Basic for Applications) donde se introducen parámetros de la prueba.

Esta aplicación actualiza la base de datos de petición de pruebas que se ejecuta sobre una hoja de Microsoft Excel en formato CVS. En la figura 6.2 se muestra la pantalla que el usuario accede para hacer la petición de las pruebas. La información que se introduce en este módulo es la generada por el sistema R2D2 y modificada con el generador de códigos de barras, con el cual se generan las peticiones.

Inicialmente lo que se pretendía era generar una conexión directa a la base de datos de R2D2 para cargar la información; sin embargo, por disposición del administrador de R2D2 el acceso fue negado y se recurrió a una solución alternativa, en la cual se sustituyó dicha conexión por la implementación del módulo de petición de prueba descrito en esta sección. Este módulo permite agregar pruebas a un experimento determinado y sirve de base para hacer la carga de recetas a los equipos según sea el requerimiento de los clientes.



The image shows a screenshot of a software application window titled "ADD REQUEST". Inside the window, the main heading is "ADD REQUEST TEST". Below this heading, there are four input fields with labels: "EPO# :", "Product Name :", "QTY :", and "Requester :". Each label is followed by a white rectangular input box. At the bottom of the window, there are three buttons: "Submit Request", "Add Step to Test", and "Quit". The window has a standard Windows-style title bar with a close button (X) in the top right corner.

Figura 6.2 Interfaz de usuario para petición de pruebas

Para desarrollar el sistema de carga de recetas, tuvieron que programarse dos funciones, la primera para agregar la información general de la prueba y la segunda, para registrar la información referente a los parámetros de configuración del equipo.

En la figura 6.3 se muestra un diagrama de flujo que describe la rutina desarrollada para agregar la información general de la prueba. En primera instancia se valida el formato de la información capturada, y se asigna la ruta del archivo donde se encuentra la base de datos de solicitud de pruebas. Luego de ello se procede a revisar en la base de datos para verificar que la receta no haya sido cargada anteriormente, esto para evitar solicitudes duplicadas. Si la receta existe, se procede a enviar un mensaje de error. Caso contrario al mensaje de error, se carga la información de la prueba dentro de la base de datos.

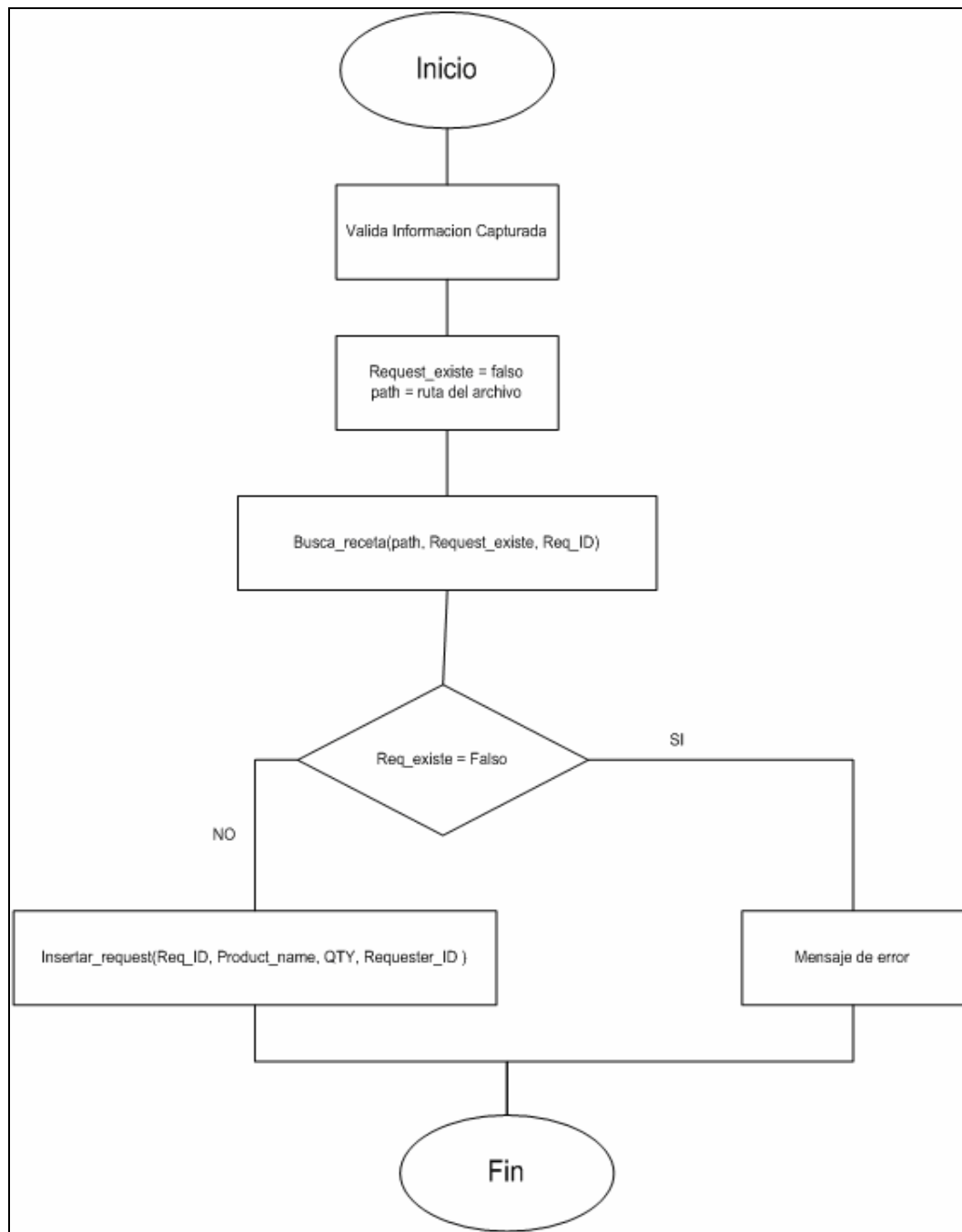


Figura 6.3 Diagrama de flujo de la operación de agregar prueba

Para cargar la información de los parámetros de la prueba se sigue el mismo proceso que se describió para cargar la información general. El diagrama de la figura 6.4 muestra dicho procedimiento.

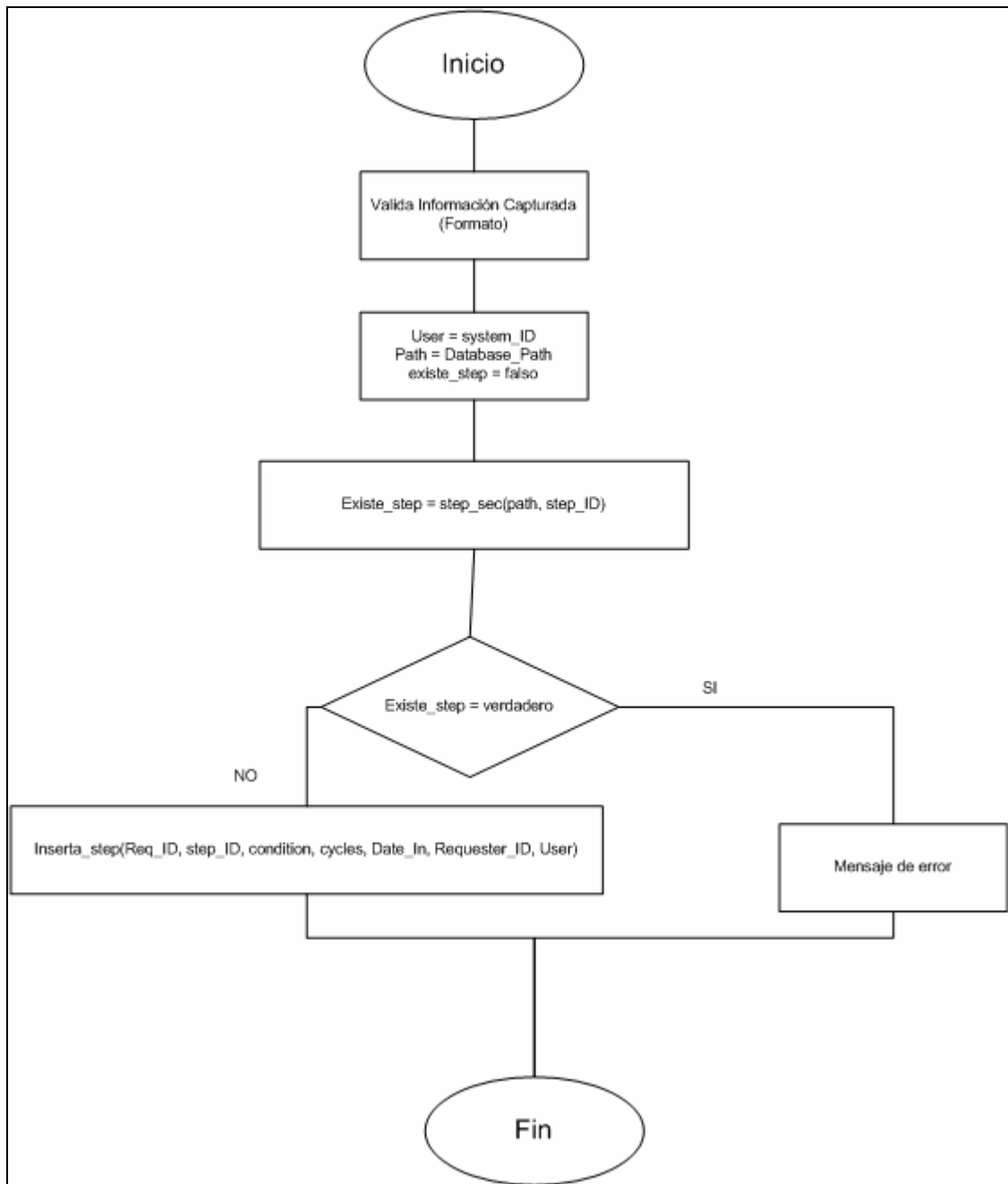


Figura 6.4 Diagrama de flujo de la operación de agregar paso

6.3.2 Módulo de carga de recetas

Este bloque de tareas tiene la función más importante de toda la plataforma pues es el que permite la carga de recetas a los equipos de manera automática. En la figura 6.5 se muestra la interfaz de usuario que fue implementada para manejar las operaciones relacionadas con carga de recetas en los equipo; seguidamente se discute la función de cada una de la operaciones.

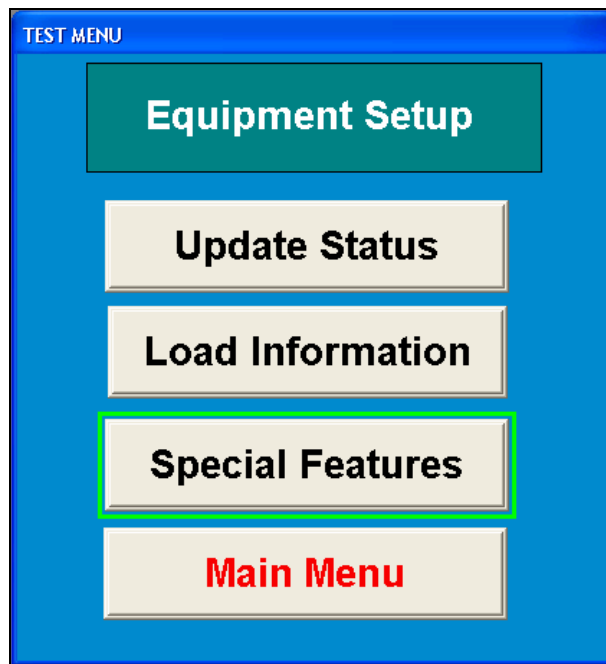


Figura 6.5 Pantalla de usuario del módulo de configuración de equipo

En este menú, se escogen las operaciones relacionadas con la configuración del equipo. El proceso de configuración del equipo se lleva a cabo como sigue: en primera instancia se tiene que verificar que el equipo esté detenido, por cuestiones de seguridad. Seguidamente se procede realizar una actualización del estado de las pruebas.

La figura 6.6, muestra la interfaz utilizada para hacer dicha actualización.

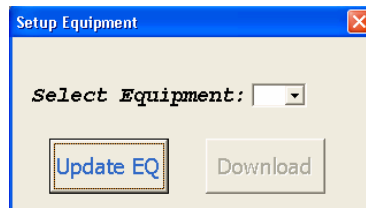


Figura 6.6 Pantalla para realizar actualizaciones del equipo

Además de las actualizaciones de la base de datos de proceso, esta pantalla se utiliza en el último paso de la configuración de los hornos, que es el envío de información al equipo seleccionado por el usuario. Seguidamente se describen los algoritmos de actualización como el de envío de datos.

La figura 6.7 muestra el diagrama de flujo de la información, cuando el usuario escoge la operación de actualizar (“update”) el equipo. El proceso que sigue la subrutina que realiza la operación, valida el formato del número de equipo, luego de esto la rutina obtiene la información de las pruebas que se ejecutan en el equipo, las carga en variables temporales y luego abre la base de datos, realiza la consulta y actualiza los valores necesarios dentro de la tabla de pruebas. Una vez realizada la actualización de la información, se envía un mensaje de confirmación al usuario y procede a cerrar la base de datos, para mantener la integridad de la información.

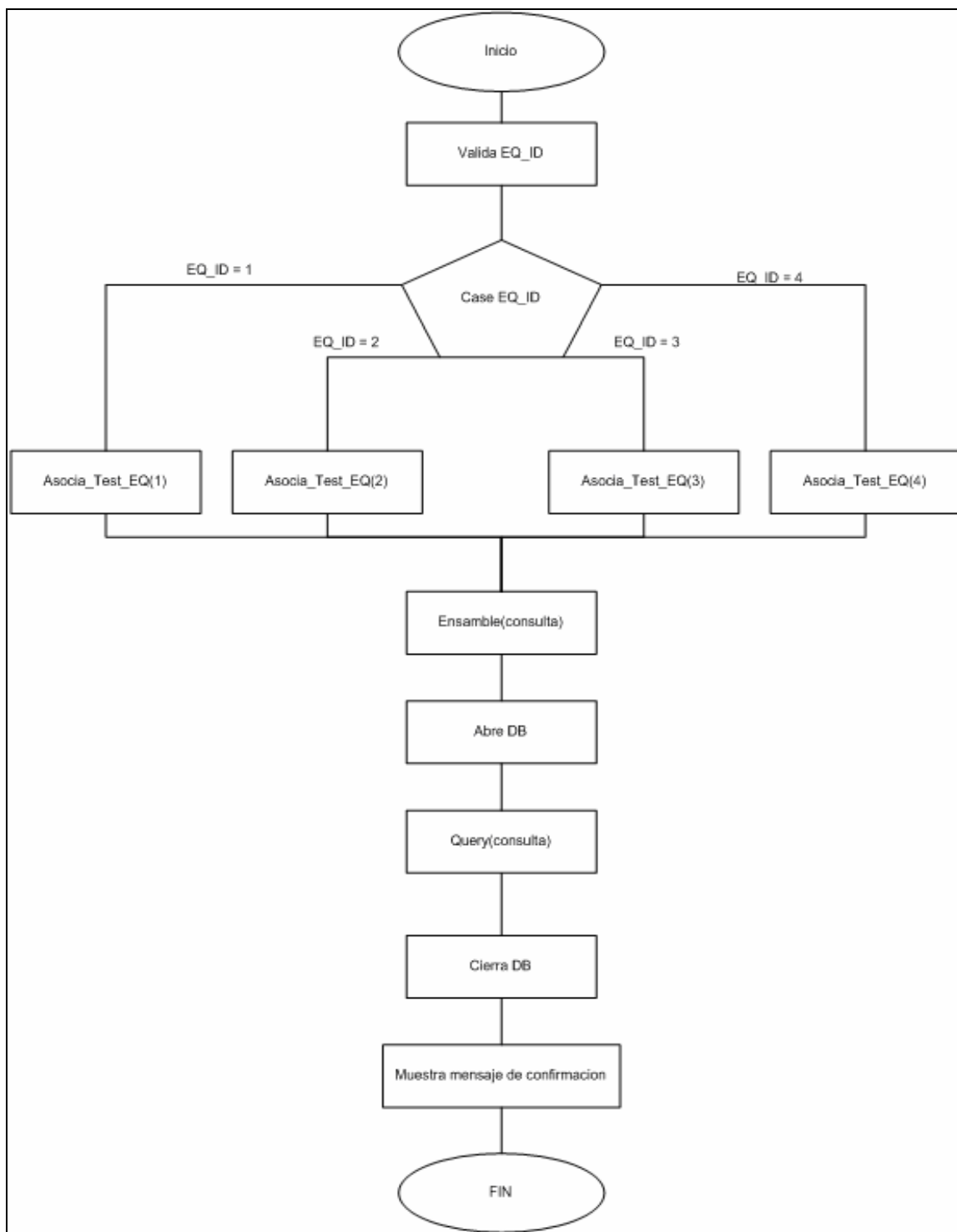


Figura 6.7 Diagrama de flujo para la actualización de los equipos

En la figura 6.6, se muestra tanto la opción de actualización como la de carga: “download”. La opción de carga de los parámetros realiza una búsqueda de pruebas en espera asociadas a un equipo y la descarga los parámetros de configuración al horno seleccionado.

Para implementar el algoritmo de descarga de parámetros se utilizó el diagrama de flujo que se muestra en la figura 6.8. Este diagrama muestra que inicialmente se verifica que el equipo puede ser accedido, luego de ello se realiza una consulta a la base de datos del proceso para obtener la lista de pruebas en curso para el equipo seleccionado.

Después de que se obtiene el resultado de la consulta, si se tiene solo una prueba, inmediatamente se procede a cargarla. En caso de encontrar más de una prueba asociada al equipo se escoge la que tenga menor cantidad de ciclos que ejecutar ya que una vez arrancada la prueba el equipo se detiene hasta terminar la cantidad de ciclos cargadas dentro de la configuración.

La cantidad de ciclos en espera para una prueba que se ejecuta de manera paralela se reduce cada vez que se carga el equipo con la receta, es decir, que si por ejemplo existían tres pruebas con las siguientes condiciones: prueba A = 100 ciclos, prueba B = 50 ciclos y prueba C = 125 ciclos, todas solicitando la misma receta R y el administrador del equipo determinó que se ejecutarían todas dentro de un mismo horno.

A la hora de descargar la información el sistema buscaría en la tabla de recetas los valores para los parámetros del horno relacionados con el identificador R, y tomaría como cantidad de ciclos para la prueba la prueba que tenga una menor cantidad, en este caso sería la de 50, en la base de datos aparecería que la prueba B tiene 0 ciclos en espera, la prueba A tendría 50 ciclos y la prueba C tendría 75 ciclos.

Con el ejemplo anterior se ilustró el algoritmo de selección de prioridad para una cola de pruebas que se ejecutan dentro de un mismo horno bajo la modalidad de multi-experimento.

Una vez cargada la información el usuario recibe un mensaje de confirmación que genera el sistema. La validación de la configuración del equipo se realizó por medio de la pantalla de tacto, ya que ella se podía leer y mostrar el estado de los parámetros de configuración de los equipos y compararlos para verificar si coincidían.

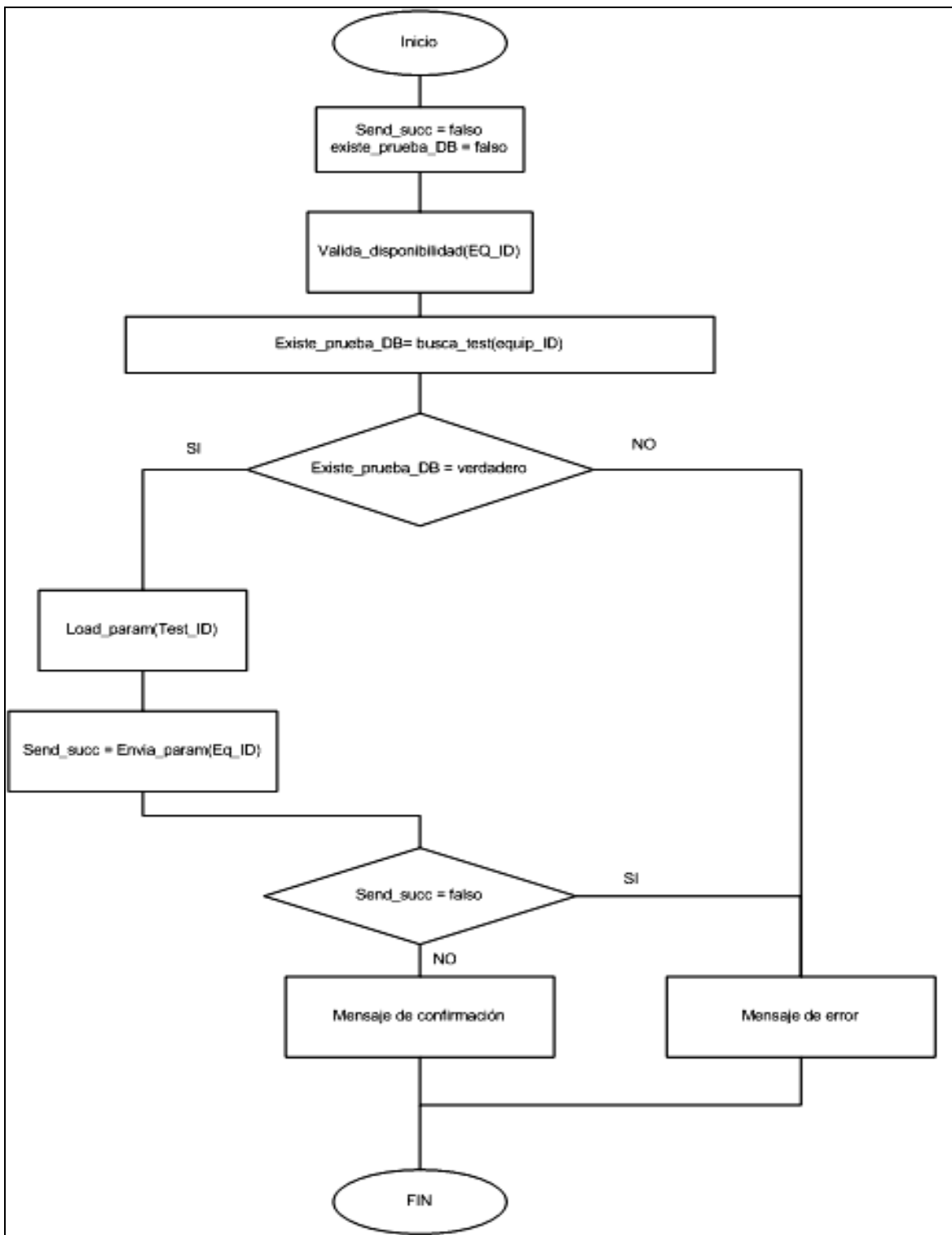


Figura 6.8 Diagrama de flujo para la descarga de información a los equipos

Una vez conocido el proceso de configuración de los equipos, se procede a discutir como fue desarrollada la aplicación que permitía asignar pruebas de la base de datos de petición a la base de datos de proceso.

Según se observa en la figura 6.5, dentro del menú de pruebas existían tres opciones: la primera, que era la relacionada con la acción de actualizar y descargar información a los equipos, ya fue explicada; ahora se analizará como fue implementada la sección encargada de asociar pruebas con los equipos. En el menú de la figura 6.5 esta opción se nombró: "Load information", y en la figura 6.9 se muestra la interfaz que fue generada para realizar dicha asociación de pruebas.

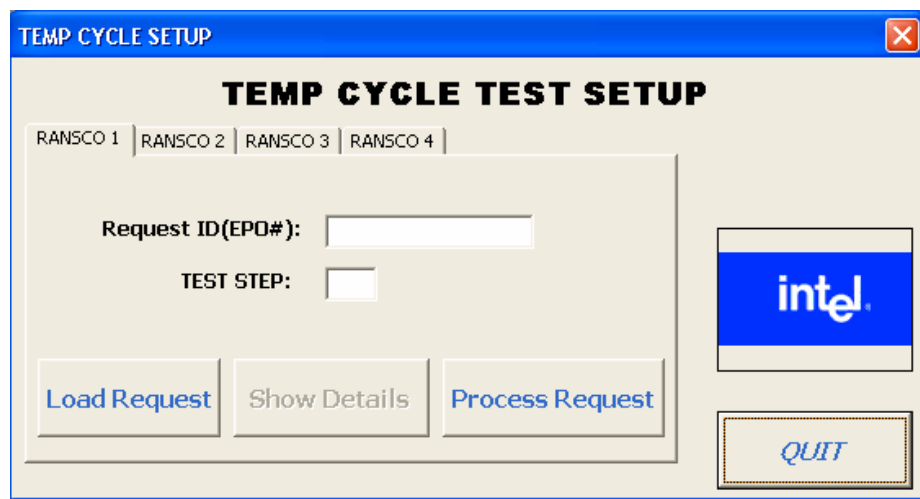


Figura 6.9 Interfaz de usuario generada para asociar las pruebas con los equipos

El modo de operación de esta pantalla se describe como sigue: en primera instancia el usuario escoge el equipo en el cual quiere cargar la prueba, seguidamente se lee la identificación de la prueba, junto con el número de paso que se quiere ejecutar. Esta función debió ser implementada de esta manera debido a que en un experimento pueden existir más de un paso, por ello se clasificaron las estaciones del experimento en lo que se constituyen como pasos.

Una vez llenada la información, se procedió a cargar la información de la prueba (identificación de la prueba + número de paso) desde la base de datos de peticiones, luego de ello, se activó el botón de "show details" con el cual se podía

observar que tipo de receta era y la cantidad de ciclos contenida dentro del paso del experimento.

Ya con la información lista se procedía a procesar la petición, lo cual significaba que en la base de datos de proceso se llenaban las tablas con el número de equipo asociado, así como el tipo de receta, la cantidad de ciclos, la hora de entrada y operador que la ejecutó. En la figura 6.10 se muestra el diagrama de flujo seguido para obtener los datos de la receta a partir del identificador y el número de paso del experimento.

En el diagrama 6.10 el procedimiento de `Asigna_temp_global ()`, guarda los valores de la prueba relacionados con los parámetros para usarlos luego en el procedimiento de llenado de información en las tablas de la base de datos de proceso.

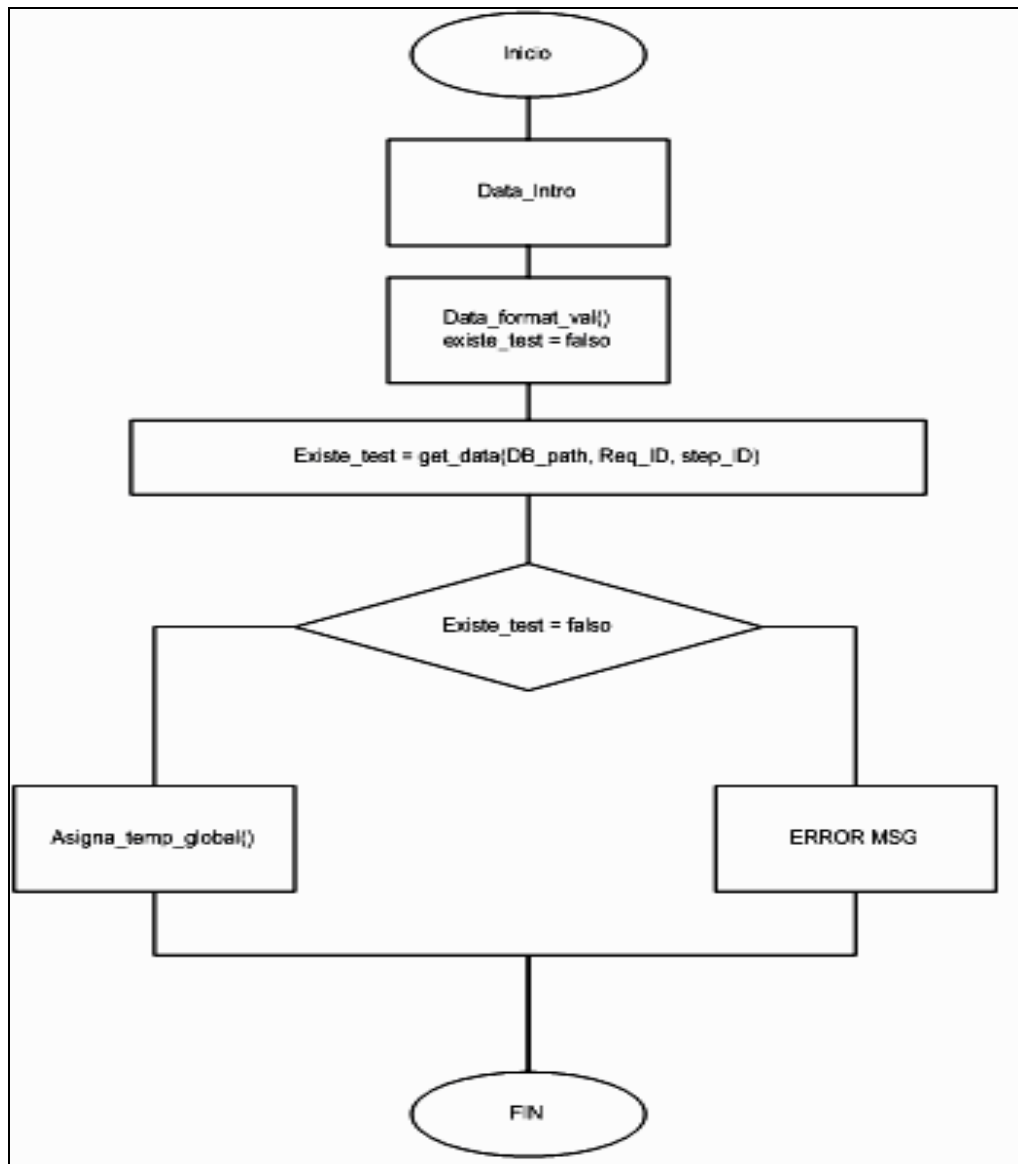


Figura 6.10 Diagrama de flujo del proceso de carga petición de prueba

En la figura 6.11 se especifica el algoritmo utilizado para procesar la petición de prueba y de esta forma cargar la información requerida a la base de datos de proceso. En primera instancia se valida si el equipo está ocupado, luego se realiza la carga de la información, según sea el identificador del equipo y la receta que se quiera ejecutar, ya que a la hora de cargar esa información se valida si el equipo es compatible con la receta requerida para la prueba.

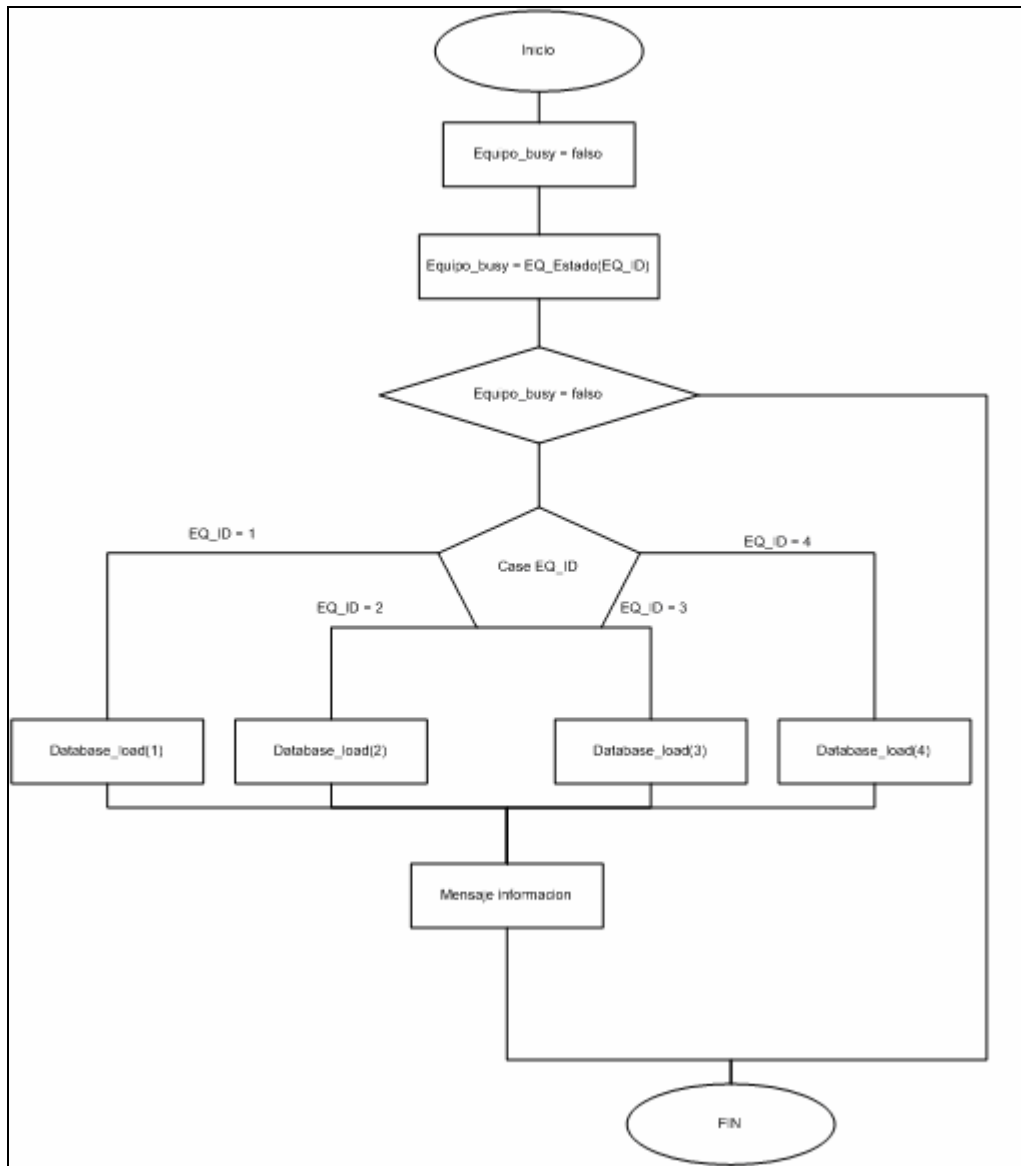


Figura 6.11 Diagrama de flujo para la carga de petición a REQ DB

En cuanto a la opción de características especiales que posee el sistema, fueron implementados solamente dos; la primera consiste en una pantalla para agregar un operador a la base de datos de proceso.

Para ello se creó una interfaz de usuario la cual se muestra en la figura 6.12. De esta interfaz se toma información y se llena la tabla correspondiente dentro de la base de datos.

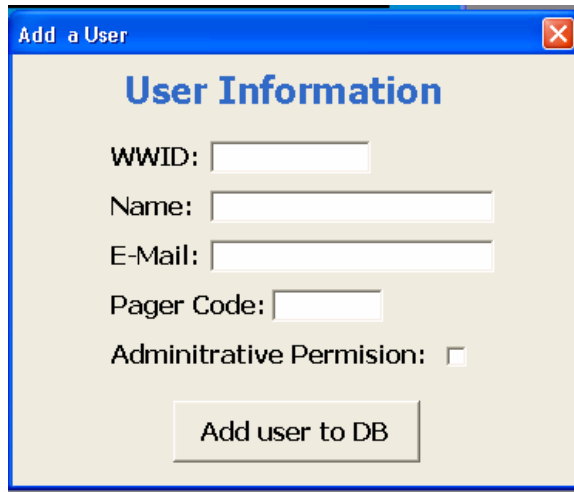


Figura 6.12 Interfaz diseñada para agregar un usuario a la PROC DB

La otra operación implementada dentro de este bloque, fue la de eliminar una prueba en la base de datos de proceso. Para ello el sistema lo que hacía era cambiar el identificador con el cual se había guardado la prueba, no se borraba ya que para los usuarios era importante conocer si alguna de las pruebas fue reasignada a otro equipo, o simplemente no se ejecutó.

En la figura 6.13 se tiene una imagen de la interfaz usada para este propósito. En ella lo que se solicita al usuario es que se ingrese el número de experimento y el paso que quiere modificar.

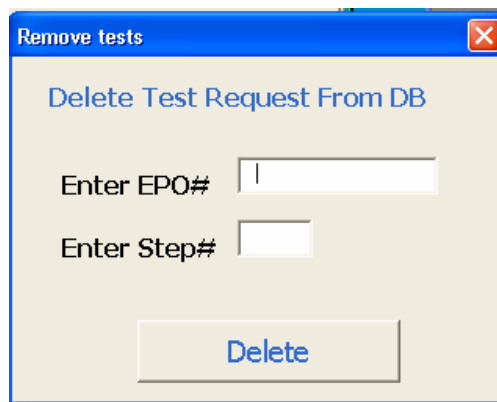


Figura 6.13 Interfaz para modificar identificador de prueba

6.3.3 Módulo de monitoreo de proceso

El monitoreo de los equipos se manejó desde dos enfoques: monitoreo en tiempo real y el monitoreo histórico. Para ejecutar el primero de ellos, se utilizó uno de los componentes de la plataforma de desarrollo Rsviwe, denominada TrendX.

Este componente se implementó sobre una de las pantallas de Rsviwe y se configuró siguiendo los pasos que muestran en la figura 6.14

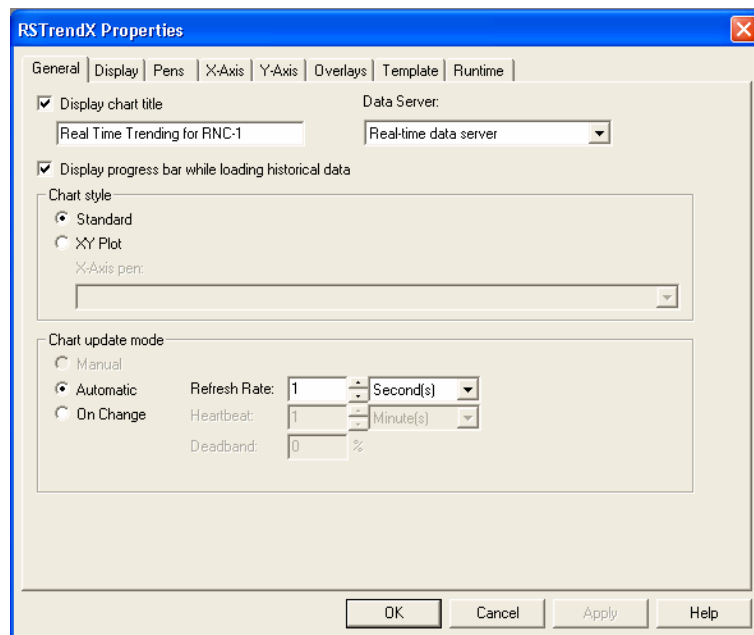


Figura 6.14 Pantalla para configuración del TrendX

Al final lo que se obtuvo fue una imagen de la tendencia de la temperatura en la que se tomaban una muestra de temperatura cada segundo. De esta manera se reconstruía el gráfico del proceso para las cámaras que contenían producto dentro de los hornos.

Con esta aplicación se logró cumplir con el objetivo de monitorear el equipo en tiempo real y así poder detectar fallas dentro del equipo basándose en los patrones de temperatura leídos dentro de la cámara del horno. Esto lo que mejoró fue la capacidad por parte del personal de mantenimiento para corregir problemas antes de que se presenten situaciones de mayor impacto para los clientes.

La figura 6.15 muestra la pantalla para monitoreo de las pruebas en tiempo real.

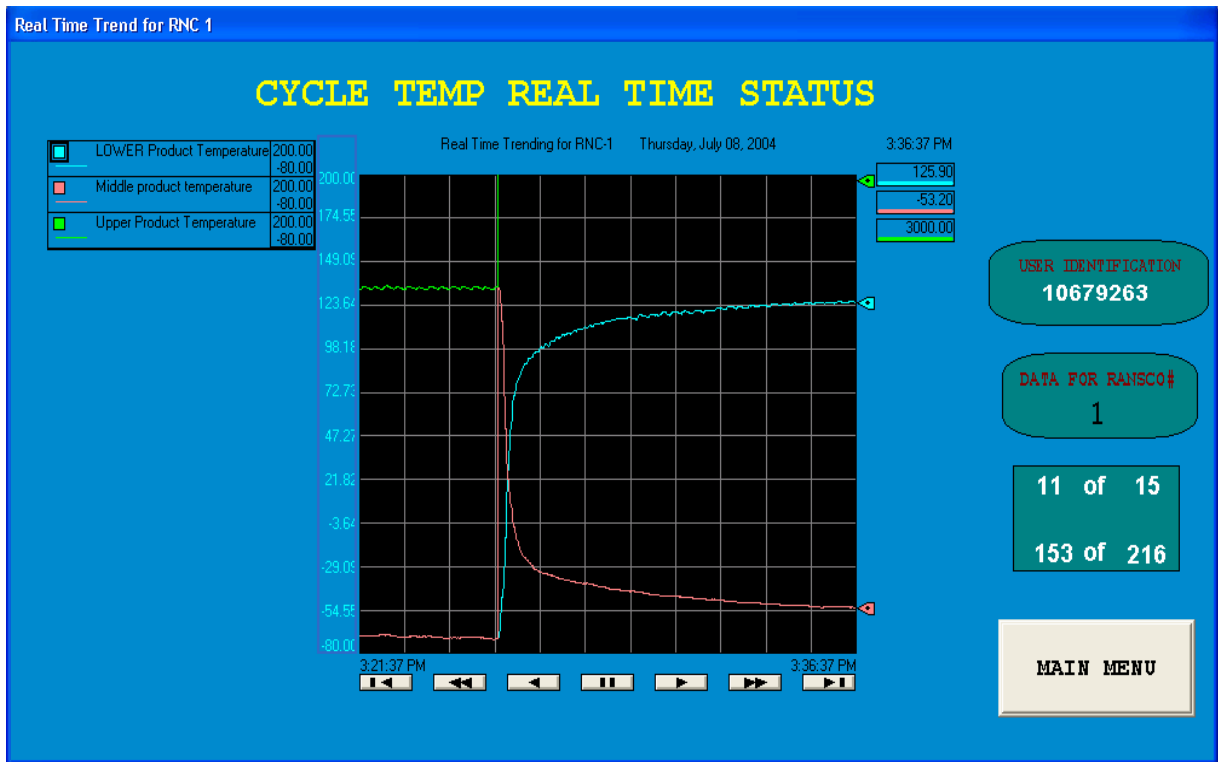


Figura 6.15 Pantalla para monitoreo de equipos en tiempo real

Luego de analizada la opción de monitoreo en tiempo real, se discute la opción relacionada con monitoreo histórico. Igual al caso anterior, esta operación se implementó usando uno de los componentes predeterminados para esta función dentro de RSVIEW.

En la figura 6.16 se observan los parámetros relevantes que fueron tomados en cuenta para configurar el componente de manera que se comporte como se deseaba desde el punto de vista del cliente final.

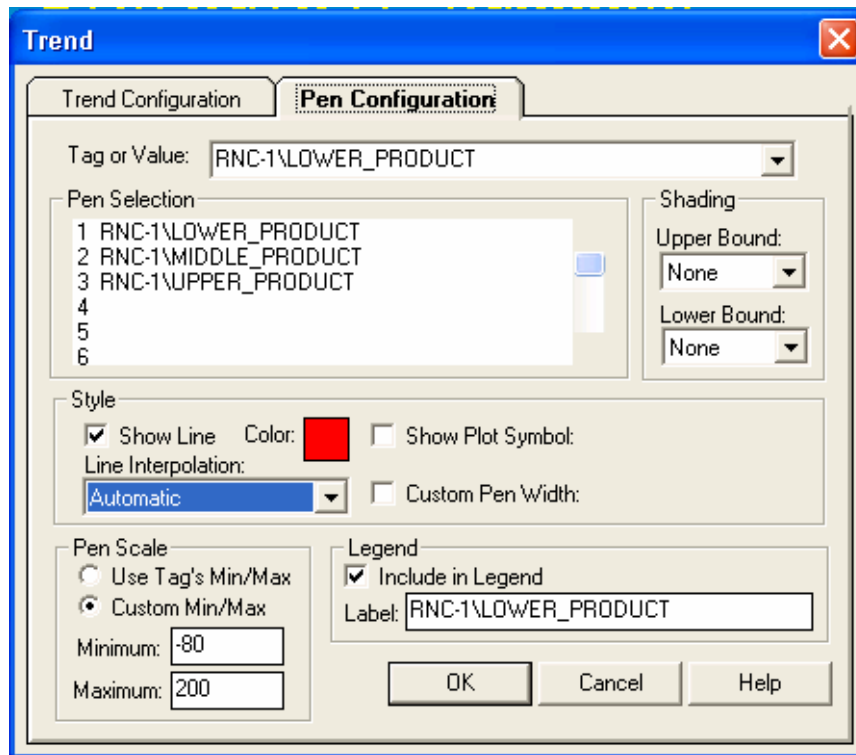


Figura 6.16 Pantalla para configuración del TrendX histórico

Como resultado de esta implementación se obtuvo una pantalla que mostraba al usuario una ventana de operación con las últimas 12 horas de proceso. Para ello se tomaban muestras cada 10 segundos.

Algo importante de recalcar es que la operación de esta pantalla estaba ligada con el archivo de recolección de datos para proceso: "data logging" lo cual permitía tener una integridad de datos mostrados con datos recolectados para respaldo de proceso y análisis de operación. En la figura 6.17 se muestra una pantalla capturada en la cual se observa la reconstrucción de un gráfico de operación histórico.

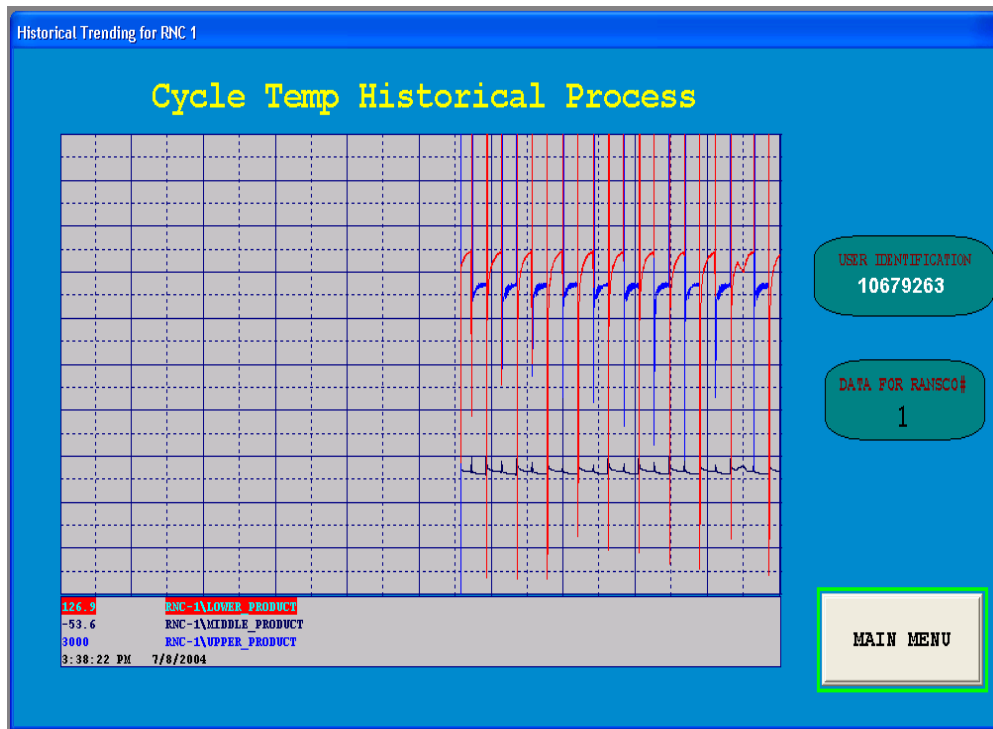


Figura 6.17 Gráfica histórica generada por el sistema

6.3.4 Módulo de control de alarmas

El objetivo que se planteaba con el módulo de control de alarmas era el de tener la capacidad para detectar y corregir las alarmas de los equipos. En este caso lo que hizo fue utilizar la capacidad de la plataforma de desarrollo Rsview para realizar esta operación y, por medio del componente “alarm summary”, mostrar a los usuarios el origen de la falla (equipo), así como el parámetro que estaba fuera de operación.

La asociación de las variables con el componente se ejecutó desde la base de datos de variables (Tag database). En esta existía un campo que indicaba si la variable debía disparar un mensaje de alarma, y las condiciones bajo las cuales se debía disparar dicho mensaje. En el apéndice A.4 se muestra la lista de variables a las cuales se les activó dicha característica.

En la parte de control de las alarmas, se implementó un módulo que reanudara la operación de los equipos de manera remota, es decir, se podían

eliminar las alarmas con solo presionar un botón desde la aplicación, sin tener que ir hasta el equipo.

Con respecto a la notificación remota, vía beeper o correo electrónico, se implementó configurando el componente de Messenger contenido en Rsviiew. Sin embargo a la hora de ejecutarlo se generaron conflictos con la aplicación de manejo de correo Outlook, por lo que se decidió removerla y dejar esta capacidad del sistema para una próxima versión del mismo. La figura 6.18 muestra la imagen de la interfaz diseñada para el usuario.

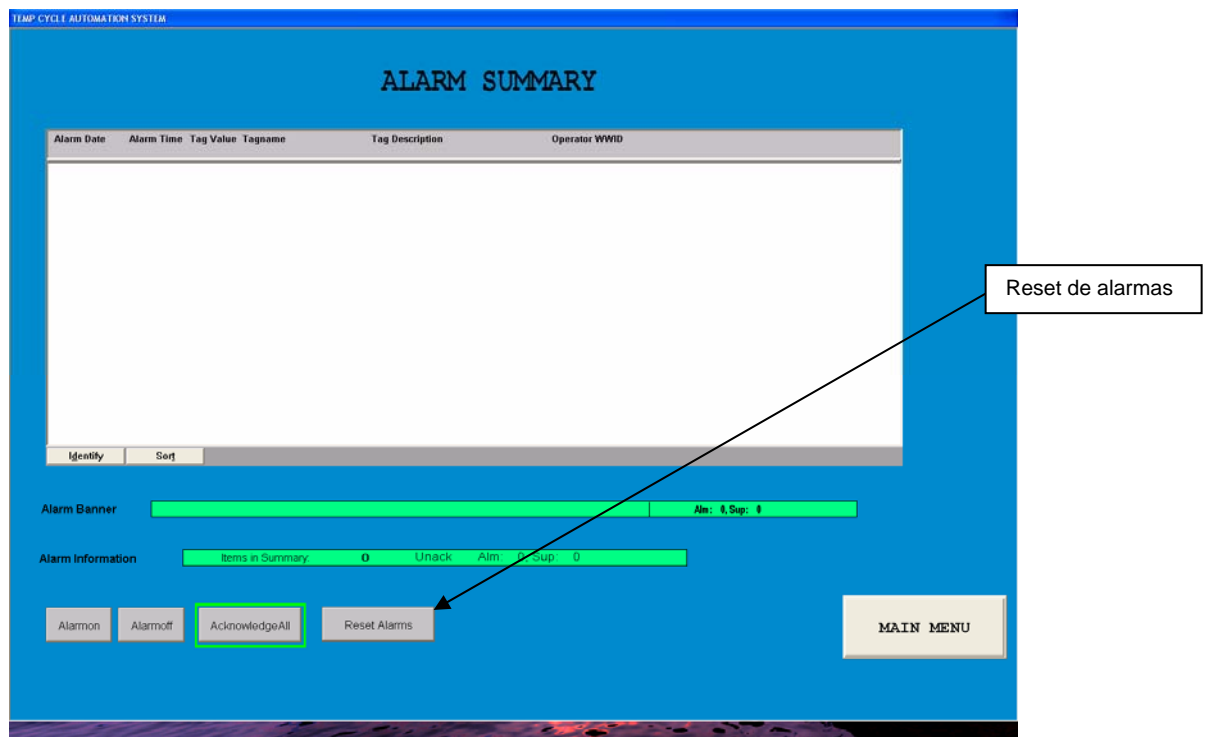


Figura 6.18 Pantalla para notificar alarmas de los equipos

6.3.5 Módulo de monitoreo virtual del equipo

Dado que al desarrollar la plataforma para control y monitoreo de los equipos remotamente, la pantalla de tacto usada anteriormente para este propósito perdería uso, entonces se decidió implementar un módulo que emulara la vista de dicha pantalla, ya que en ocasiones se hacia necesario poder leer valores de operación que únicamente se mostraban en este medio. Para ello se creó la interfaz que se

muestra en la figura 6.19. En dicha figura lo que se hizo fue asignar los tags respectivos a esos equipos de manera que a la hora de invocar la pantalla se tuviera una copia de los datos que se estaban mostrando en ese mismo instante en la pantalla de tacto ubicada en el panel frontal del horno.

Para cada uno de los equipos, se diseñó una pantalla en Rsview. La pantalla muestra los parámetros de configuración así como datos de avance de la prueba. También se muestra indicadores del estado del equipo, como modo de prueba, tipo de control y estado de la rampa de temperatura, entre otros.

Al final se obtuvo una correspondencia correcta entre la información desplegada en la pantalla de tacto en el equipo como en la emulación dentro de la plataforma lo cual garantizó que la implementación había sido ejecutada exitosamente.

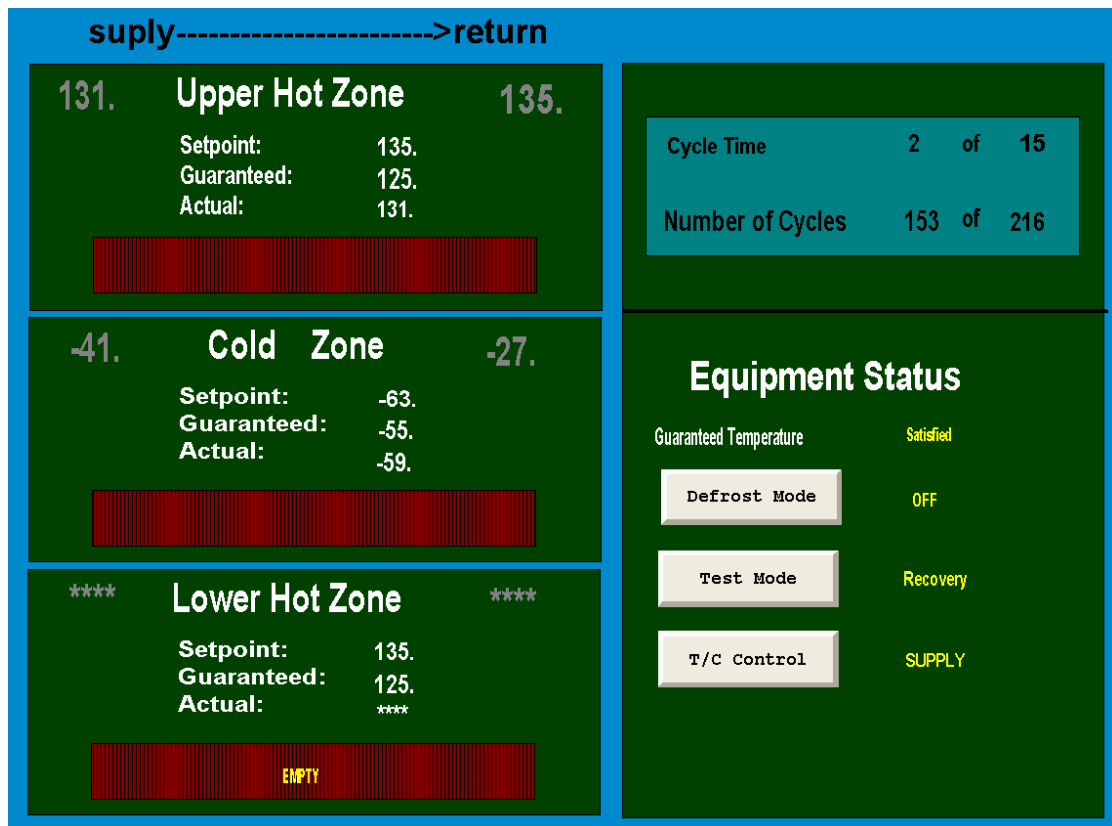


Figura 6.19 Imagen de la pantalla de tacto emulada

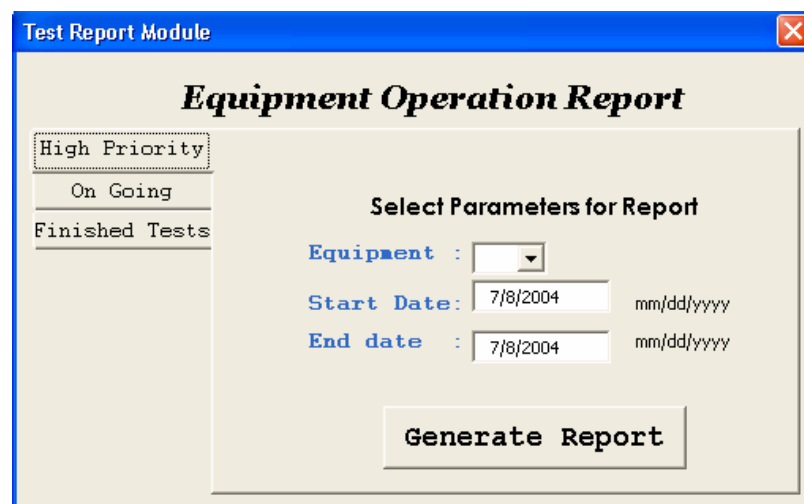
6.3.6 Módulo de reporte de operación de los equipos

Debido a que el proceso en cuestión requería un mejoramiento en la calidad, se debía tener un mecanismo que permitiera llevar registro de las pruebas ejecutadas, así como información de las pruebas en proceso de ejecución. De esta manera se estableció un sistema básico de reportes que respondía a tres criterios de filtro: estado de la prueba, rango de fecha, número de equipo.

Fue de esta forma como se establecieron tres niveles de estado de la prueba, el primero de ellos era el de “alta prioridad” (“High priority”). En este caso el sistema generaba una hoja de Microsoft Excel con la información de la prueba que se estaba ejecutando que tenía más alta prioridad, o sea, menor número de ciclos por ejecutar.

La otra categoría para establecer el estado de la prueba era la denominada “pruebas en proceso” (“On going”). Bajo este concepto, se mostraban todas la pruebas tanto en espera como las que se estaban ejecutando de manera paralela en alguno de los equipos.

Finalmente el último criterio, fue el denominado “pruebas finalizadas”, (“finished”), en la cual de igual forma se agrupaban las pruebas terminadas para un equipo y rango de fechas definidos por el usuario. La interfaz desarrollada para generar reportes de operación de los equipos se muestra en la figura 6.20



The screenshot shows a software window titled "Test Report Module" with a close button in the top right corner. The main content area is titled "Equipment Operation Report". On the left side, there are three buttons: "High Priority" (which is highlighted with a dotted border), "On Going", and "Finished Tests". To the right of these buttons is a section titled "Select Parameters for Report". This section contains three input fields: "Equipment" (a dropdown menu), "Start Date" (a text box containing "7/8/2004" with "mm/dd/yyyy" to its right), and "End date" (a text box containing "7/8/2004" with "mm/dd/yyyy" to its right). At the bottom center of the form is a large button labeled "Generate Report".

Figura 6.20 Sistema para generar reportes de operación de los equipos

6.3.7 Otras consideraciones sobre el software

Además del desarrollo del software, la plataforma requirió que se implementara una estrategia de seguridad para proteger la aplicación de usuarios no deseados y con ello evitar la posibilidad de corrupción en la operación del sistema. Esta característica se desarrolló usando una de las opciones de Rsview denominada “user accounts”, en la cual se creó una lista de usuarios con derechos de operación sobre el sistema. En la figura 6.21 se muestra la pantalla generada por Rsview para autenticar la entrada de los usuarios.

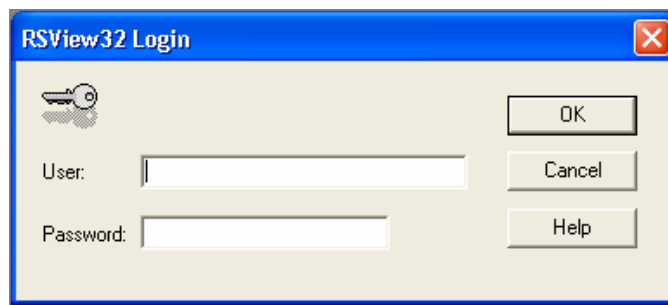


Figura 6.21 Pantalla para autenticación de usuarios

De esta forma se finaliza con la discusión diseño de software el cual fue extensivo por la gran cantidad de módulos que requerían para cumplir con todos los requerimientos de la empresa.

6.3.8 Alcances y limitaciones

De la solución que se propuso al inicio solo se tuvieron problemas en la implementación del sistema de notificación de alarmas vía correo electrónico y beeper, debido a las políticas de seguridad de los sistemas de información no permitían al sistema implementado su funcionalidad. En cuanto a la sección de reportes se sugirió que se estudiara mejor la estructura de los filtros para obtener reportes personalizables según la necesidad del usuario.

Capítulo 7

Conclusiones y recomendaciones

7.1 Conclusiones

- Las pantallas de tacto deben ser reconfiguradas para trabajar en entorno de red.
- El valor del parámetro de “time out” no afecta el rendimiento de la red de equipos.
- La estructura de datos jerarquizada permite mejor control sobre la información del sistema.
- El diseño de la aplicación por medio de módulos facilitó la prueba e integración de los mismos a la plataforma principal.
- La implantación de bases de datos sobre un archivo CVS es eficiente solamente cuando la información contenida no está relacionada con otras tablas o bases de datos.
- El sistema multiexperimento requirió que aumentara la complejidad del manejo de la información por parte de la base de datos de proceso.
- El monitoreo en tiempo real permitió generar patrones de operación para determinar el comportamiento de los hornos.
- Las bibliotecas especializadas de Rsview, aceleraron el tiempo de desarrollo del sistema.
- El control de alarmas de manera remota disminuyó la cantidad de tiempo que los equipos se mantenían detenidos cuando se presentaba un mal funcionamiento.

- La inclusión de la emulación de la pantalla de tacto dentro de la plataforma permitió tener un instrumento de comparación para validar el funcionamiento del sistema.
- El sistema de recolección de datos aumentó la confiabilidad del proceso de prueba realizado por los hornos, pues respalda su operación.
- El módulo de reportes aceleró el proceso de generación de reportes de capacidad de los hornos.

7.2 Recomendaciones

- Se debe mejorar la especificación de requerimientos a la hora de plantear nuevos módulos, esto con el fin de reducir el tiempo de desarrollo de los mismos.
- Se deberían utilizar los tiempos de mantenimiento del equipo para dar mantenimiento a la aplicación y con ello evitar que la herramienta no esté disponible cuando se la necesite.
- A la hora de realizar una expansión del sistema para dar soporte a otros equipos que no sean compatibles con la red actual, se puede utilizar una tecnología basada en el estándar OPC.
- En futuras ampliaciones del sistema, se deben mejorar los tiempos de entrega por parte de los proveedores para evitar retrasos en la implementación.
- Se debe asignar la administración del sistema a una persona que tenga suficiente conocimiento sobre los hornos y el proceso.

Bibliografía

1. Rockwell Automation. "RSview 32 User guide."
<<http://www.ab.com/manuals/9399-2S32UG.pdf>>
2. Rockwell Automation. "Analog I/O Modules for SLC 500 Programmable Controllers, Technical Data."
<<http://www.ab.com/manuals/io/1746/1746-td001a-en-p.pdf>>
3. Rockwell Automation. "1746-2.35 Discrete Input and Output Modules, Technical Data."
<<http://www.ab.com/manuals/io/1746/1746235.pdf>>
4. Rockwell Automation. "SLC 500 Analog input modules."
<<http://www.ab.com/manuals/cp/1762-UM001A-US-P.pdf>>
5. Rockwell Automation. "SLC 500 Analog I/O Modules"
<<http://www.ab.com/manuals/io/1746-6.4.pdf>>
6. Rockwell Automation. "SLC 500 Family of Small Programmable Controllers, System Overview."
<<http://www.ab.com/manuals/1747230.pdf>>

Apéndices y anexos

Apéndice A.1 Glosario

AIC: Módulo acoplador de vínculo. Permite insertar un nodo nuevo en una red DH - 485.

CVS: formato de almacenamiento de la información en el que los datos están separados por comas.

Data Log: metodología utilizada para la recolección de información de procesos basada en la realización de un muestreo de los mismos.

Data Files: Sistema de archivos utilizados dentro del programa del PLC para localizar las variables.

DH +: Protocolo de comunicación para redes industriales con características de alta velocidad.

DH-485: protocolo de comunicación utilizado en la implementación de redes industriales.

HMI: Interfaz humano-máquina

Ladder: Metodología utilizada en los PLC para subdividir el programa principal en procedimientos.

Messenger: aplicación que permite comunicar al usuario sobre el estado del equipo.

OPC: estándar para recolección de datos en aplicaciones a nivel industrial por medio de medio de un esquema cliente-servidor. Sus siglas en ingles significan: Ole for process control.

PV900: Modelo de la pantalla de tacto que utilizan los hornos.

Panelview: Modelo de la pantalla de tacto que poseen los hornos en cuestión.

Panelbuilder: Software utilizado para el desarrollo de la interfaz a través de la pantalla de tacto.

PKTX: Nombre de la serie del adaptador de red utilizado para la conexión del servidor a la red de equipos.

PLC: controlador lógico programable.

Q&R: siglas en ingles para el termino calidad y confiabilidad.

RANSCO: Nombre del fabricante de las cámaras de choque térmicos.

RSlogix: Software especializado para la programación de los controladores lógicos programables (PLC).

RSInx: Software para administrar la interfaz de los Software de la marca Rockwell.

RSVIEW32: Software utilizado para el desarrollo de la interfaz de usuario.

SLC 500: Familia de controladores lógicos de la marca Allen Brandley. Sus siglas en ingles significan: *Small logic controller*.

SPC: Siglas en ingles para *statistical process control*. Es un modulo de control estadístico para los datos recolectados en el equipo.

SQL: Sus siglas en ingles significan *Standard Query language*. Es un estándar para crear consultas en bases de datos.

TAG: se refiere a una variable dentro del sistema.

Apéndice A.2 Distribución de señales del RANSCO en el PLC

Input Module

En la tabla A.1 se muestra la descripción de las funciones que realizan cada una de las señales de entrada al módulo del PLC.

Tabla A. 1 Descripción de la función de las señales de entrada del PLC

| Entrada | Componente | Descripción |
|---------|-------------------------------------|--|
| 0 | Termostato | Sensa el estado del compresor para el HS |
| 1 | Sistema protección de aceite del HS | Sensa estado del sistema de aceite del HS |
| 2 | Sensor de presión | Detecta condición de alta presión para el HS(high stage) |
| 3 | Sensor de presión | Detecta condición de baja presión para el HS(high stage) |
| 4 | Sensor de presión | Detecta condición de baja presión para el LS(high stage) |
| 5 | Sensor de presión | Detecta condición de alta presión para el LS(Low stage) |
| 6 | Sistema protección de aceite del LS | Sensa estado del sistema |
| 7 | Termostato | Sensa el estado del compresor para el LS |
| 8 | Switch | Permite señal para apagar la activación de la alarma silenciosa. |
| 9 | Bottom Switch | Detecta cuando el elevador ha llegado a |

| | | |
|----|----------------------------------|--|
| | | su limite superior |
| 10 | TOP Switch | Detecta cuando el elevador ha llegado a su limite inferior. |
| 11 | PUSH Button | Activa señal para la transferencia manual del elevador |
| 12 | Interruptor activado por presión | Activa señal para el sistema de descogelamiento de las cámaras fría. |
| 13 | PUSH Button | Activa/desactiva señal para el proceso general, ya sea para meter mas unidades a la cámara o para revisar alguna condición |
| 14 | Switch Limit | Sistema de seguridad para abrir las cámaras. Es un PIN metálico. |
| 15 | Sensor de sobre temperatura | Detecta alguna condición de sobretemperatura en las zonas calientes, tanto superior como inferior |

Relay Output Module

Este módulo se encarga de accionar los diversos componentes que constituyen la cámara, los cuales se accionan de acuerdo al programa del PLC y las condiciones que se tengan a la entrada del sistema.

La tabla A.2 resume las señales de salida del sistema controlador PLC.

Tabla A. 2 Descripción de la función de las señales de salida del PLC

| Número de salida | Conectada a Componente | Función |
|-------------------------|---|--|
| 0 | No especificado* | Activa el sistema de presión del LS |
| 1 | Contactador | Activa el Abanico de la Zona Fría, junto con el flujo de aire caliente y líquidos en el compartimiento caliente superior(HS) |
| 2 | No especificado | Activa los abanicos de circulación de aire para las Zonas Calientes de la cámara. |
| 3 | No especificado | Activa el sistema de control de calor de la Zona Fría |
| 4 | No especificado | Activa el sistema de control de calor del compartimiento caliente superior (HS) |
| 5 | No especificado | Activa el sistema de control de calor del compartimiento caliente Inferior (LS) |
| 6 | Interruptores de accionados por presión | Activa el sistema de circulación de líquidos en la etapa caliente inferior (LS) |
| 7 | No especificado | Accionar el sistema de circulación de gas caliente en la etapa caliente inferior(LS) |

| | | |
|----|-----------------------|--|
| 8 | Sistema de movimiento | Transfiere el compartimiento hacia arriba a la siguiente etapa de la cámara. |
| 9 | Sistema de movimiento | Transfiere el compartimiento hacia abajo a la siguiente etapa de la cámara. |
| 11 | Mallori Sonalert | Activa la alarma audible cuando se presenta alguna condición especial en el funcionamiento normal del sistema. |
| 12 | No especificado | Activa sistema para purgar el Nitrógeno Gaseoso existente en el equipo |
| 13 | Módulo luminoso | Activa la luz de alarma |
| 14 | No especificado | Control para sistema de Nitrógeno Líquido. (opcional) |
| 15 | No especificado | Sistema Redundante de Control para sistema de Nitrógeno Líquido. (opcional) |

La figura A.1 muestra un diagrama eléctrico del circuito del módulo de salida:

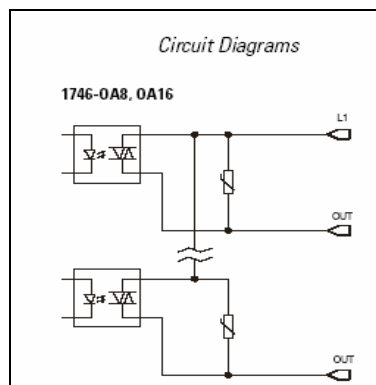


Figura A.1 Circuito de salida del 1746-0A16

Termopars Input Module

En el Ransco se pueden encontrar dos módulos de entradas para termopares, los cuales poseen cada uno de ellos cuatro entradas, para un total de ocho entradas de retroalimentación de temperatura en diversas ubicaciones dentro del equipo.

En la tabla A.3 se muestra un listado de las características de este módulo.

Tabla A. 3 Conexiones asociadas a los módulos de termopares

| Modulo # | Entrada # | Ubicación | Función |
|----------|-----------|------------|--|
| 1 | 0 | UHZ Supply | Sensa temperatura en el alimentador de aire de la zona caliente superior(HS) |
| | 1 | UHZ Return | Sensa temperatura en el recolector de aire de la zona caliente superior(HS) |
| | 2 | LHZ Supply | Sensa temperatura en el alimentador de aire de la zona caliente inferior(LS) |
| | 3 | LHZ Return | Sensa temperatura en el recolector de aire de la zona caliente inferior(LS) |
| 2 | 0 | CZ Supply | Sensa temperatura en el alimentador de aire de la zona fría (CZ) |
| | 1 | CZ Return | Sensa temperatura en el recolector de aire de la zona fría(CZ) |

Apéndice A.3 Listado de variables de proceso

En la tabla A.5 se muestra un listado con la información referente a la base de datos de variables de proceso. Este listado se consiguió por medio del software panel builder y se adaptó a cada uno de los equipos.

Tabla A. 4 Listado de las variables del PLC utilizadas

| Nombre | Tipo de dato | Dirección | Valor por defecto |
|---------------|----------------------|------------------|--------------------------|
| cycle_counter | Unsigned Integer | N7:11 | 0 |
| cycle_time | Unsigned Integer | N7:12 | 0 |
| cz_cool_out | Unsigned Integer | N11:2 | 0 |
| cz_heat_out | Unsigned Integer | N11:17 | 0 |
| cz_return | Signed Integer / Int | I:4.1 | 0 |
| cz_sp | Signed Integer / Int | N19:1 | 20 |
| cz_supply | Signed Integer / Int | I:4.0 | 0 |
| czc_d | Unsigned Integer | N10:5 | 0 |
| czc_i | Unsigned Integer | N10:4 | 0 |
| czc_p | Unsigned Integer | N10:3 | 1500 |
| czh_d | Unsigned Integer | N90:5 | 0 |
| czh_i | Unsigned Integer | N90:4 | 0 |
| czh_p | Unsigned Integer | N90:3 | 1500 |
| defrost_off | Bit | B3/38 | 0 |
| defrost_on | Bit | B3/17 | 0 |
| defrost_state | Bit | B3/7 | 0 |
| elv_in_lwr | Bit | I:2/10 | 0 |
| elv_in_up | Bit | I:2/9 | 0 |
| elv_mid | Bit | B3/30 | 0 |

| Nombre | Tipo de dato | Dirección | Valor por defecto |
|-----------------|----------------------|------------------|--------------------------|
| guarenteed_cold | Signed Integer / Int | N19:3 | 20 |
| guarenteed_hot | Signed Integer / Int | N19:2 | 20 |
| hs_oil | Bit | B3/81 | 0 |
| hs_pd | Bit | B3/80 | 0 |
| hs_ps | Bit | B3/82 | 0 |
| hs_td | Bit | B3/79 | 0 |
| lhz_d | Unsigned Integer | N60:5 | 0 |
| lhz_i | Unsigned Integer | N60:4 | 0 |
| lhz_return | Signed Integer / Int | I:3.3 | 0 |
| lhz_sp | Signed Integer / Int | N7:32 | 20 |
| lhz_supply | Signed Integer / Int | I:3.2 | 0 |
| lower_product | Signed Integer / Int | N7:104 | 0 |
| ls_oil | Bit | B3/85 | 0 |
| ls_pd | Bit | B3/84 | 0 |
| ls_ps | Bit | B3/83 | 0 |
| ls_td | Bit | B3/86 | 0 |
| lz_out | Unsigned Integer | N11:12 | 0 |
| over_temp | Bit | B3/87 | 0 |
| reset_alarms | Bit | B3/60 | 0 |
| reset_bit | Bit | B3/51 | 0 |
| tc_control | Bit | B3/15 | 0 |
| test_mode | Unsigned Integer | N7:0 | 0 |
| time_sec | Unsigned Integer | N7:22 | 0 |
| total_cycles | Unsigned Integer | N19:5 | 100 |

| Nombre | Tipo de dato | Dirección | Valor por defecto |
|---------------|----------------------|------------------|--------------------------|
| uhz_d | Unsigned Integer | N30:5 | 0 |
| uhz_i | Unsigned Integer | N30:4 | 0 |
| uhz_out | Unsigned Integer | N11:7 | 0 |
| uhz_p | Unsigned Integer | N30:3 | 1500 |
| uhz_return | Signed Integer / Int | I:3.1 | 0 |
| uhz_sp | Signed Integer / Int | N19:0 | 20 |
| uhz_supply | Signed Integer / Int | I:3.0 | 0 |

Apéndice A.4 Listado de variables asociadas con alarmas

La tabla A.5 muestra la lista de variables que disparan alarmas dentro de los equipos. Para cada equipo se manejan la misma lista de variables dentro de la base de datos general.

Tabla A. 5 Variables asociadas con alarmas para cada equipo

| Variable | Valor | Mensaje de Alarma |
|-----------------|--------------|---------------------------------------|
| HS_Td | 1 | High Stage High Discharge Temperature |
| HS_Pd | 1 | High Stage Low Oil Pressure |
| HS_OIL | 1 | High Stage High Discharge Pressure |
| HS_Ps | 1 | High Stage Low Suction Pressure |
| LS_Ps | 1 | Low Stage Low Suction Pressure |
| LS_Pd | 1 | Low Stage High Discharge Pressure |
| LS_OIL | 1 | Low Stage Low Oil Pressure |
| LS_Td | 1 | Low Stage High Discharge Temperature |
| Transfer_jam | 1 | Transfer Jam |
| OVER_TEMP | 1 | Temperature Alarm |

Apéndice A.5 Direccionamiento de red

En la tabla A.6 se muestra la tabla de direccionamiento generada para la red de equipos implementada.

Tabla A. 6 Direccionamiento utilizado en la implementación de la red

| Dirección | Nombre | Descripción |
|------------------|---------------|-----------------------------------|
| 1 | RNC_1 | PLC del equipo RANSCO 1 |
| 2 | PNV_1 | Pantalla de Tacto Equipo RANSCO 1 |
| 3 | RNC_2 | PLC del equipo RANSCO 2 |
| 4 | PNV_2 | Pantalla de Tacto Equipo RANSCO 2 |
| 5 | RNC_3 | PLC del equipo RANSCO 2 |
| 6 | PNV_3 | Pantalla de Tacto Equipo RANSCO 3 |
| 31 | Server | Servidor de monitoreo y control |

Anexo B.1 Diagramas de conexión de los dispositivos de la red

Tarjeta PKTX

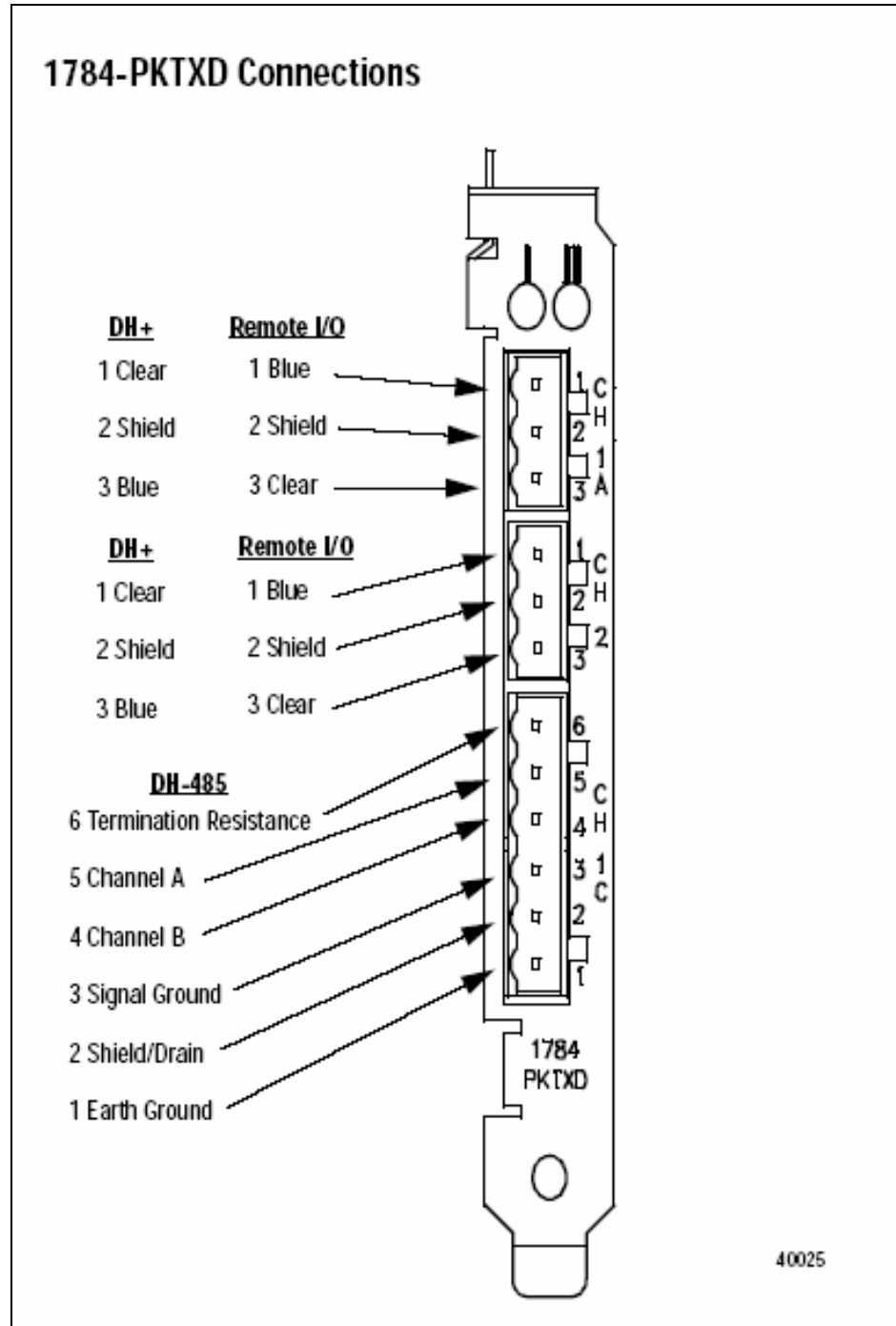


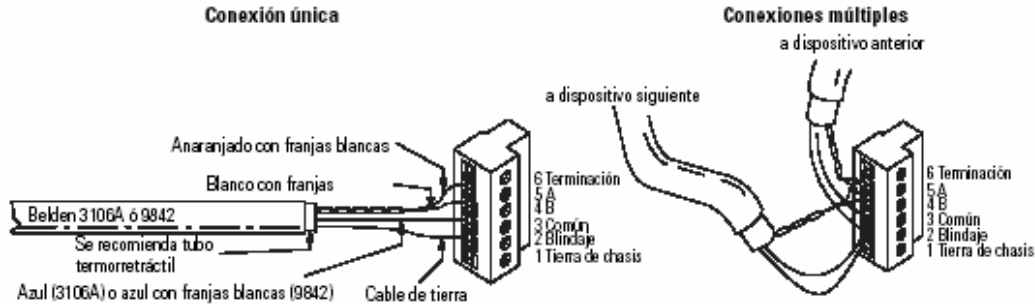
Figura A.2 Diagrama de cableado de la tarjeta de red PKTX

Diagrama de conexión de los acopladores de vínculo

Conexiones DH-485

Conexión del cable de comunicaciones al acoplador de vínculo

Conecte el bloque de terminales del acoplador de vínculo al cable Belden 3106A ó 9842 tal como se muestra a continuación.



| Belden 3106A | | | Belden 9842 | | |
|--------------------|--------------------------------|-----------------|--------------------|---|------------------------------------|
| Para este hilo/par | Conecte este hilo | A este terminal | Para este hilo/par | Conecte este hilo | A este terminal |
| Blindaje/tierra | Sin forro | 2 - Blindaje | Blindaje/tierra | Sin forro | 2 - Blindaje |
| Azul | Azul | 3 - Común | Azul/blanco | Blanco con franjas azules Azul con franjas blancas | Sin conexión, córtese 3 - Común |
| Blanco/anaranjado | Blanco con franjas anaranjadas | 4 - Datos B | Blanco/anaranjado | Blanco con franjas anaranjadas | 4 - Datos B |
| | Anaranjado con franjas blancas | 5 - Datos A | | Anaranjado con franjas blancas | 5 - Datos A |

Puesta a tierra y terminaciones de la red DH-485

Sólo uno de los acopladores de vínculo en el extremo del vínculo debe tener los terminales 1 y 2 del conector de red puenteados entre sí.

Los acopladores de vínculo en ambos extremos de la red deben tener los terminales 5 y 6 de los conectores del vínculo puenteados entre sí.

Utilice un cable de par trenzado blindado Belden 9842.

Anexo B.2 Hoja de información del proyecto

Información del estudiante

Nombre: Johan Carvajal Godínez

Cédula: 1-1104-0873

Carné ITCR: 9907367

Dirección de su residencia en época lectiva: San Joaquín de Flores, Heredia, 50 m sur, 100 m este, y 50 sur del Abastecedor el Edén.

Dirección de su residencia en época no lectiva: Pejibaye, Pérez Zeledón, 1 Km. Sur de la Escuela San Miguel .

Teléfonos en época lectiva: 374-4195

Teléfonos en época no lectiva: 352-6782

Email: johan.carvajal.godinez@intel.com

Información del Proyecto

Nombre del Proyecto: Automatización del sistema de configuración de parámetros para las pruebas de choque térmico de los Microprocesadores Intel

Área del Proyecto: Automatización y Redes

Profesora Asesora: Arys Carrasquilla Batista

Información de la Empresa

Nombre: Componentes Intel de Costa Rica

Zona: Heredia, Belén, La Ribera, Calle 129

Dirección en la cual se ubicación dentro de la empresa: CR-1, Stress LAB, Room 120.

Teléfono en la empresa: 298-7361 **Su extensión en la empresa:** 6-7361

Fax: 298-7396 **Apartado:** 845-1150, La Rivera de Belén, CR.

Actividad Principal de la empresa: Ensamble y prueba de Microprocesadores

Información del asesor en la empresa

Nombre: Jorge Luis Rojas del Barco

Puesto que ocupa: Stress Lab Owner

Departamento: Calidad y confiabilidad

Profesión: Ingeniero Electromecánico **Grado académico:** Bachillerato

Teléfono: 298-6595

Ext.: 6-6595

Fax: 298-7396

Email: jorge.luis.rojas@intel.com