

13  
2  
hij



**UNIVERSIDAD NACIONAL AUTÓNOMA DE  
MÉXICO**

ESCUELA NACIONAL DE ESTUDIOS PROFESIONALES  
CAMPUS "ARAGÓN"

**"DISEÑO Y DESARROLLO DEL SISTEMA DE  
MONITOREO Y CONTROL DE GABINETES DE  
MEDICIÓN DE PETROLEO CRUDO DE  
EXPORTACIÓN PARA LA TERMINAL  
MARÍTIMA PAJARITOS, VER."**

**T E S I S**

QUE PARA OBTENER EL TÍTULO DE  
**INGENIERO EN COMPUTACIÓN**

P R E S E N T A N:

**SONIA EDITH CORRAL DIAZ  
MARCO ANTONIO RIVERA REYES**

ASESOR : ING. SILVIA VEGA MUYTOY

MÉXICO

1996

**TESIS CON  
FALLA DE ORIGEN**

**TESIS CON  
FALLA DE ORIGEN**



Universidad Nacional  
Autónoma de México



**UNAM – Dirección General de Bibliotecas**  
**Tesis Digitales**  
**Restricciones de uso**

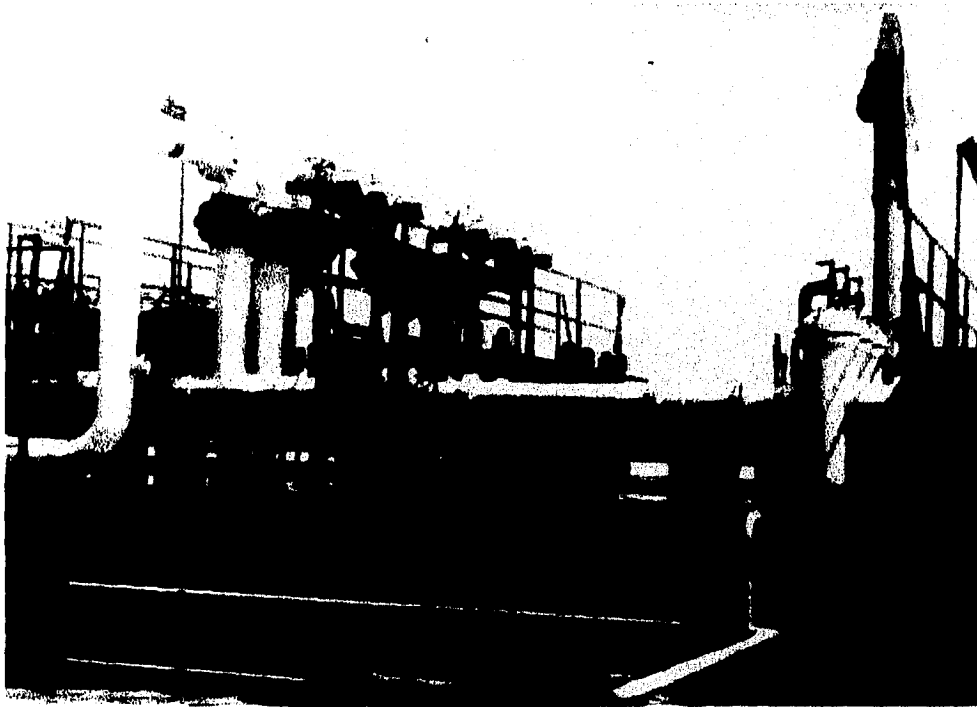
**DERECHOS RESERVADOS ©**  
**PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL**

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

**UNIVERSIDAD NACIONAL AUTONOMA DE MEXICO  
CAMPUS ARAGON**

**" DISEÑO Y DESARROLLO DEL SISTEMA DE  
MONITOREO Y CONTROL DE GABINETES DE  
MEDICION DE PETROLEO CRUDO DE EXPORTACION  
PARA LA TERMINAL MARITIMA PAJARITOS, VER. "**



**PRESENTAN:  
SONIA EDITH CORRAL DIAZ  
MARCO ANTONIO RIVERA REYES**

**MEXICO, 1996**

**A MIS PADRES:  
POR QUE ESTE LOGRO NO SOLO ES MIO, ES NUESTRO.**

**A MIS HERMANOS:  
POR HABERME APOYADO SIEMPRE A PESAR DE QUE YO NO SIEMPRE LOS  
APOYO.**

**A MI GORDITA:  
POR AYUDARME, REGAÑARME Y QUERERME. TODO AL MISMO TIEMPO Y CON  
LA MISMA INTENSIDAD.**

**A LA MEMORIA DEL ING. JUAN MENDEZ MORENO:  
POR QUE CADA CLASE, FUE UNA CATEDRA DE VIDA.**

**A TODOS USTEDES LES  
AGRADEZCO POR IGUAL  
Y LES DEDICO ESTE TRABAJO**

**MARCO A. RIVERA REYES**

MARCO A. RIVERA REYES

## AGRADECIMIENTOS



**A MIS PADRES:**

**POR ESTAR SIEMPRE CONMIGO, APOYANDOME Y RESPETANDO MIS DECISIONES. QUIERO QUE SEPAN QUE ESTE ESFUERZO FUE DE TODOS Y SIN USTEDES NO LO HUBIERA LOGRADO. GRACIAS POR SER QUIEN SON.**

**LOS AMA SU HIJA SONIA**



**CHAPIS:**

**GRACIAS POR AGUANTAR MIS ENOJOS, LA LUZ PRENDIDA EN ESTAS NOCHES DE DESVELO Y POR TODO LO QUE TUVIMOS QUE PASAR PARA LLEGAR HASTA AQUI. TUSABES QUE TE QUIERO MUCHO, OJALA NUNCA LO OLVIDES.**

**SONIA**



**GORDITO:**

**GRACIAS POR SER MI AMIGO, MI PAREJA Y POR SER LO MEJOR DEL MUNDO. GRACIAS POR ESTAR CONMIGO.**

**TE AMA SONIA**



**A TERE (MARGIE):**

**HOY QUIERO DARTTE LAS GRACIAS POR TODOS LOS MOMENTOS BUENOS Y MALOS QUE COMPARTIMOS Y QUE HICIERON DE MI ESTANCIA EN LA ENEP LOS MEJORES AÑOS DE MI VIDA. GRACIAS POR SER MI AMIGA. UN AMIGO ES UN HERMANO QUE ELEGIMOS.**

**SONIA (BOBY)**

**A LA ING. SELVA VEGA M:  
POR ANDAR A MI LADO EN ESTE TRABAJO  
DE FORMA POSIBLE Y POR  
SIEMPRE TENER SIEMPRE PARA  
ENCUONAR A SUS ALUMNOS.  
NO ES RICO EL QUE TIENE MUCHO,  
SINO EL QUE DA MUCHO.**



## DEDICATORIAS

**A DIOS:**



SE QUE EL CAMINO QUE ME ESPERA ES DIFICIL PERO ES TAMBIEN EL QUE MAS VALE LA PENA Y EL UNICO QUE SE PUEDE VOLTEAR A VER CON ORGULLO CUANDO SE ESTA AL FINAL DE EL. LAS UNICAS DOS ARMAS QUE TENGO PARA TRATAR DE CRUZARLO SON LA VOLUNTAD Y LA FE EN TI Y EN MI, SE QUE SI LAS MANIPULO BIEN PODRE LLEGAR AL FINAL Y SER FELIZ.

**A TI MAMI Y PAPI:**



ESTE TRABAJO ES MUY PEQUEÑO OBSEQUIO QUE LES QUIERO DAR POR SER LOS MEJORES PADRES DEL MUNDO Y POR HABER COMPARTIDO TODOS MIS DESVELOES Y TODO MI ESFUERZO.

**LOS QUIERO MUCHO: SONY**

**RUTH:**



ERES MI HERMANA FAVORITA (PORQUE ERES LA UNICA) Y POR ESTO TE DIGO QUE SIEMPRE ESTARE A TU LADO. ESTE TRABAJO ES TAMBIEN TUYO, YA QUE HAS ESTADO SIEMPRE APOYANDOME, CLARO A TU MODO, PERO TE DIGO QUE YO TAMBIEN TE QUIERO MUCHO.

**TU HERMANA SONY**



A ESA PERSONA ESPECIAL. TU MI GRAN AMOR (TONNO). ESTO NOS COSTO MUCHO TRABAJO, ENOJOS, RISAS Y NERVIOS PERO SIEMPRE ESTUVIMOS JUNTOS. SOLO ASI CONSEGUIREMOS LO QUE DESEAMOS.

**TE AMO: SONIA**



**TESIS**

**COMPLETA**

---

## INDICE

<b>INTRODUCCIÓN</b> .....	1
<b>CAPITULO 1. CONTROL Y MONITOREO</b> .....	3
1.1 Definición de un Sistema de Control.....	3
1.2 Componentes de un Sistema de Control.....	4
1.2.1 Características de un Sistema de Control.....	6
1.3 Historia del Control.....	6
1.4 El Problema del Control.....	8
1.5 Control por Computadora.....	9
1.5.1. Componentes del Control por Computadora.....	12
1.6 Tipos de Control.....	15
1.6.1. Control Supervisorio.....	15
1.6.1.1. Control Supervisorio Básico.....	16
1.6.2. Control Digital Directo.....	16
1.6.3. Control Centralizado.....	17
1.6.4. Control Distribuido.....	19
1.6.5. Control Jerárquico.....	26
1.6.5.1. Funciones en un Sistema de Control de Procesos.....	28
1.7 Definición de Monitoreo por Computadora.....	30
1.8 Importancia del Monitoreo por Computadora.....	31
1.9 Tipos de Monitoreo por Computadora.....	32
1.9.1. Monitoreo Gerencial.....	32
1.9.2. Monitoreo de Campo o Proceso.....	33
<b>CAPITULO 2. DESCRIPCIÓN DEL SISTEMA DE MONITOREO Y CONTROL</b> .....	34
2.1 Componentes del Sistema de Monitoreo.....	34
2.1.1. Sistema Operativo.....	34
2.1.1.1. S.O. Batch.....	38
2.1.1.2. S.O. Interactivo Monousuario.....	38
2.1.1.3. S.O. en Tiempo Real.....	39
2.1.1.4. S.O. en Tiempo Compartido.....	39
2.1.2. Interfaz de Operación.....	39
2.1.3. Driver de Comunicación.....	42
2.2 Componentes del Sistema de Control.....	43
2.2.1. Patines de Medición de Petróleo Crudo.....	44
2.2.2. Totalizador de Flujo Daniel 2230.....	47
2.2.3. Microcontrolador Daniel MM3000.....	49
2.2.4. Bum Brown TM71.....	53
<b>CAPITULO 3. DESARROLLO DEL SOFTWARE</b> .....	56
3.1 Antecedentes del Sistema de Monitoreo y Control.....	56
3.2 Interfaz de Operación.....	60
3.2.1. Selección de la Interfaz de Operación.....	62
3.2.2. Diseño de Pantallas de Operación.....	63
3.2.3. Programación de la Interfaz de Operación.....	77
3.3 Driver de Comunicación.....	78

---



3.3.1. Diseño.....	78
3.3.2. Programación del Driver de Comunicación.....	108
3.4 Operación del Nuevo Sistema de Monitoreo y Control. ....	108
<b>CAPITULO 4. REQUERIMIENTOS DE HARDWARE Y SOFTWARE .....</b>	<b>124</b>
4.1 Diagrama General del Sistema.....	124
4.2 Software.....	125
4.3 Hardware.....	125
4.4 Diagrama de Conexiones.....	127
<b>CAPITULO 5. INSTALACIÓN, ARRANQUE Y MANTENIMIENTO DEL SISTEMA .....</b>	<b>129</b>
5.1 Instalación.....	129
5.2 Arranque.....	131
5.3 Mantenimiento.....	132
<b>CONCLUSIONES.....</b>	<b>133</b>
<b>APENDICE A.- PROGRAMAS DE LA INTERFAZ DE OPERACIÓN.....</b>	<b>135</b>
<b>APENDICE B.- PROGRAMACIÓN DEL DRIVER DE COMUNICACIÓN .....</b>	<b>162</b>
<b>GLOSARIO .....</b>	<b>189</b>
<b>BIBLIOGRAFÍA.....</b>	<b>193</b>

---

## INTRODUCCIÓN

En el año de 1983 la compañía DANIEL INDUSTRIES INC. suministro a Petróleos Mexicanos un sistema automatizado de control para realizar la medición de petróleo crudo de exportación que se embarca en buques-tanque en la Terminal Marítima Pajaritos, localizada en Coatzacoalcos, Ver.

La medición y el control automatizado del sistema de embarque de petróleo crudo se realiza a través de un sistema integrado por unidades, basadas en microprocesador denominadas DANIEL MM3000, que funcionaban conjuntamente con una minicomputadora Hewlett-Packard A600 llamada computadora terminal de medición.

La minicomputadora Hewlett-Packard A600 que permitía realizar operaciones de monitoreo y control sobre el sistema, se encuentra fuera de servicio. Debido a daños sufridos en algunos de sus componentes y por el alto costo que implicaba su reparación y mantenimiento.

Por lo anterior la Superintendencia General de Transportación Marítima del Puerto Pajaritos, Ver., de Pemex-Refinación, solicitó al Instituto Mexicano del Petróleo realizar la modernización del sistema de medición y control, la cual implicara, la integración de un equipo de cómputo y una interfaz de operación (software) de tecnología de punta, que en lo posterior serán referidos como consola de operación, que conjuntamente realizaran las mismas funciones de monitoreo y control desempeñadas por la minicomputadora A600, proporcionando mejoras notables en cuanto a la calidad de despliegue de gráficos y al manejo del sistema.

Es importante remarcar que la filosofía de operación del sistema es similar al de la minicomputadora A600, independientemente de la tecnología de vanguardia del equipo de cómputo y del software que son suministrados.

El presente trabajo tiene la finalidad de mostrar el diseño y el desarrollo del nuevo sistema de monitoreo y control solicitado por PEMEX. Para entender este sistema es necesario conocer la diferencia entre el control y el monitoreo.

En el capítulo 1 se da un panorama general de lo que es el control y el monitoreo, así como su clasificación y diferentes elementos que los componen.

En el capítulo 2 se describen los elementos más importantes de un sistema de monitoreo, los cuales son: Sistema Operativo, Interfaz de Operación para la comunicación de la computadora con el operador y un Driver para la comunicación de la computadora con el equipo de control, así mismo se describen los elementos más importantes de un sistema de control como son: Sensores, Dispositivos de Actuación y Controlador. Y de manera particular para el nuevo sistema se definen los elementos que lo componen como: Patines de Medición, Totalizador de flujo, Microcontrolador Daniel MM3000 y un teclado Burr Brown TM71.

Para llevar a cabo la realización del nuevo sistema es necesario conocer el funcionamiento actual del sistema que se va a sustituir. En el capítulo 3 se explica este funcionamiento, así como el diseño y programación de las nuevas pantallas de operación, el diseño y desarrollo del Driver que permite la comunicación entre la interfaz de operación y el controlador Daniel MM3000 y la forma de operación del nuevo sistema de monitoreo y control.

El capítulo 4 muestra el diagrama de conexión de los elementos de hardware del nuevo sistema de monitoreo y control, el diagrama general del sistema y los requerimientos mínimos de hardware y software, así como los elementos que están instalados actualmente y los que se requieren para su mejor funcionamiento.

Por último, en el capítulo 5 se muestran los pasos a seguir para la instalación y el arranque del sistema, además del mantenimiento necesario para que este funcione de manera óptima.

---

## CAPITULO 1. CONTROL Y MONITOREO

En la actualidad los procesos en la industria son realizados en conjunto por dos partes fundamentales: el monitoreo y el control.

El monitoreo tiene la función de supervisar al proceso, mientras que el control es el encargado de hacer que el sistema funcione de acuerdo a los parámetros establecidos por el operador.

A continuación se dará la definición de control y monitoreo con sus respectivos componentes y tipos.

### 1.1 DEFINICIÓN DE UN SISTEMA DE CONTROL

Controlar implica dirigir, ordenar o mandar con el propósito de alterar lo que se este controlando de acuerdo a las necesidades presentes. Así, en el sentido más estricto, un sistema de control es aquel que modifica a su medio ambiente.

Control significa medir el valor de la variable controlada del sistema, y aplicar al sistema la variable manipulada para corregir o limitar la desviación del valor medido, respecto al valor deseado.

Un sistema de control es un ordenamiento de componentes mecánicos, eléctricos, hidráulicos o neumáticos que unidos o relacionados entre sí mandan, dirigen o regulan al mismo sistema, es decir, permite al operador controlar el flujo de energía a una máquina o proceso para obtener el desempeño deseado.

El control automático ha jugado un papel vital en el avance de la ciencia y de la ingeniería. Además de su extrema importancia en vehículos espaciales, sistemas de guía de proyectiles, sistemas de piloto automático de aeronaves, sistemas robóticos y otros, el control automático se ha vuelto parte integral e importante de los procesos industriales y de manufactura modernos. Por ejemplo, el control automático resulta esencial en el control numérico de las máquinas y herramientas en las industrias manufactureras.

También resulta esencial en operaciones industriales como el control de presión, temperatura, humedad y viscosidad, y flujo en las industrias de transformación.

Así en el sentido más amplio se puede decir que un sistema de control es cualquier interconexión de componentes que satisfacen una función deseada.

## 1.2 COMPONENTES DE UN SISTEMA DE CONTROL.

En forma general un sistema consta de tres etapas: planta o proceso, entrada y salida. Estas etapas se explican a continuación y se muestran en la figura 1.1.

**PLANTA O PROCESO:** Es una operación o desarrollo natural, caracterizado por una serie de cambios graduales, progresivamente continuos, que se suceden uno a otro de un modo relativamente fijo, y que tienden a un determinado resultado final; o a una operación voluntaria o artificial progresivamente continua, que consiste en una serie de acciones controladas o movimientos dirigidos sistemáticamente hacia determinado resultado o fin. Es cualquier operación que deba controlarse.

**ENTRADA:** Es el estímulo, la excitación o el mandato aplicado a un sistema de control para producir una respuesta específica.

**SALIDA:** Es la respuesta real que se obtiene de un sistema de control, puede ser o no igual a la respuesta implícita especificada por la entrada.



FIGURA 1.1. SISTEMA DE CONTROL

Se puede usar un **CONTROLADOR** (o regulador) para producir un comportamiento deseado de la planta, como se muestra en la figura 1.2. Este controlador genera las señales de entrada a la planta que han sido diseñadas para producir las salidas planeadas. Algunas entradas a la planta son accesibles al diseñador y otras, por lo general, no lo son. Las señales de entrada no accesibles son llamadas perturbaciones.

**PERTURBACIÓN:** Una perturbación es una señal que tiende a afectar adversamente el valor de salida de un sistema. Si la perturbación se genera dentro del sistema, se denomina *interna*, mientras que una perturbación *externa* se genera fuera del sistema y constituye una entrada.

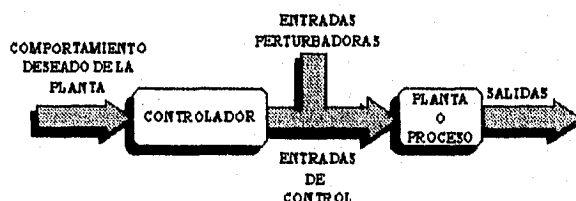


FIGURA 1.2. SISTEMA DE CONTROL CON CONTROLADOR

**RETROALIMENTACIÓN:** Cuando existe retroalimentación, el sistema compara el valor efectivo de salida de una planta con el valor deseado, determina la desviación y produce una señal de control que disminuye dicho valor a cero o a un valor muy pequeño cercano a cero. Este sistema es mostrado esquemáticamente en la figura 1.3.

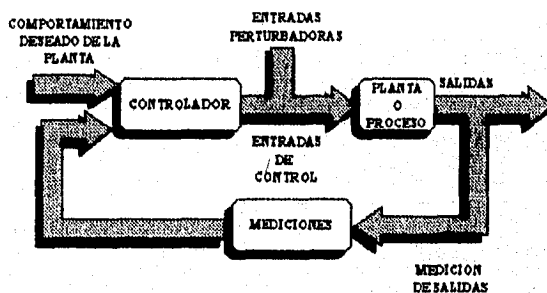


FIGURA 1.3. SISTEMA DE CONTROL CON RETROALIMENTACIÓN

### 1.2.1 CARACTERÍSTICAS DE UN SISTEMA DE CONTROL.

Un sistema de control debe ser:

**ESTABLE:** Esto significa que la respuesta a una señal, ya sea un cambio del punto de referencia o una perturbación, debe alcanzar y mantener un valor útil durante un periodo razonable.

**EXACTO:** Un sistema debe ser exacto dentro de ciertos límites especificados, el sistema debe ser capaz de reducir cualquier error a un valor aceptable.

**RÁPIDO EN SU RESPUESTA:** Un sistema debe completar su respuesta a cierta señal de entrada en un tiempo razonable. Aunque un sistema sea estable y tenga la exactitud requerida, no tiene ningún valor si el tiempo de respuesta es mucho mayor que el tiempo entre las señales.

### 1.3 HISTORIA DEL CONTROL

Algunos principios de control automático ya eran utilizados en el siglo tercero antes de Cristo; Teslbio, inventor griego, diseñó en esa época una clepsidra que utilizaba un mecanismo similar al usado en los carburadores de los automóviles actuales. El termostato y algunos aparatos que hacían que los molinos mantuvieran las aspas contra el viento, aún cuando éste cambiara de dirección, fueron inventados en los siglos XVII y XVIII respectivamente.

Sin embargo, los sistemas de control automático comenzaron su desarrollo intenso durante la revolución industrial. El primer trabajo significativo en control automático fue el regulador centrífugo de James Watt para el control de la velocidad de una máquina de vapor, en el año de 1788.

A mediados del siglo XIX, James C. Maxwell analizó por primera vez diversos tipos de gobernadores de velocidad, y relacionó el problema de la estabilidad de los

mismos con uno algebraico, el cual fue resuelto posteriormente por Hurwitz. Otros avances relevantes en las primeras etapas del desarrollo de la teoría del control se deben a Minorsky, Hazen y Nyquist, entre muchos otros. En 1922 Minorsky trabajó en controladores automáticos de dirección en barcos y mostró cómo se podría determinar la estabilidad a partir de ecuaciones diferenciales que describen el sistema. En 1932, Nyquist desarrolló un procesamiento relativamente simple para determinar la estabilidad de los sistemas de lazo cerrado sobre la base de la respuesta a lazo abierto con excitación sinusoidal en régimen permanente (análisis para sistemas retroalimentados utilizando conceptos de respuesta en frecuencia). En 1934 Hazen, quien introdujo el término **servomecanismos\*** para los sistemas de control de posición, desarrolló el diseño de servomecanismos repetidores capaces de seguir con exactitud una entrada cambiante.

Durante la Segunda Guerra Mundial, el interés en las aplicaciones bélicas, hizo que se consideraran problemas de dirección y guía de proyectiles balísticos, lo que tuvo como consecuencia el estudio de sistemas estocásticos y no lineales. Desde los últimos años de la década de 1950, gracias al advenimiento de las computadoras digitales, se han redescubierto las variables de estado y se han estudiado más profundamente problemas tales como optimización y control bajo incertidumbre; también se han desarrollado en los últimos veinte años el control numérico y el control jerarquizado.

En la actualidad las plantas modernas tienen muchas entradas y salidas, por lo que se van haciendo más complejas, ocasionándose así que la descripción de un sistema moderno de control requiera de una gran cantidad de ecuaciones. La teoría de control clásica, que trata de sistemas con una entrada y una salida, se vuelve absolutamente impotente ante sistemas de múltiples entradas y salidas. Hacia 1960, gracias a la disponibilidad de las computadoras digitales, se hizo posible el análisis de sistemas complejos en el dominio del tiempo; desde entonces se ha desarrollado la teoría de control moderna, basada en el análisis y síntesis en el dominio del tiempo, utilizando variables de estado, con lo que es posible afrontar la complejidad creciente de

---

\* Sistema de control retroalimentado en el que la salida es algún elemento mecánico, sea posición, velocidad o aceleración.



las plantas modernas y los estrictos requisitos de exactitud, peso y costo en aplicaciones militares, espaciales e industriales.

Los desarrollos más recientes en la teoría de control moderno están en el campo del control óptimo de sistemas, tanto determinísticos como estocásticos, así como en sistemas de control complejos con adaptación y aprendizaje. Ahora que las computadoras digitales se han abaratado y reducido en tamaño, éstas pueden utilizarse como parte integral de estos sistemas de control. Las aplicaciones recientes de la teoría de control moderna incluyen sistemas no ingenieriles como los de biología, biomedicina, economía y socioeconomía.

#### 1.4 EL PROBLEMA DEL CONTROL

El control de sistemas físicos con computadoras digitales se está volviendo más común, por ejemplo, en Autopilotos de Aviación, Refinerías de Petróleo, Máquinas para hacer papel e incontables servomecanismos electromecánicos.

El Problema del Control es el diseñar el sistema para obtener el desempeño deseado, siendo el mejor diseño el que tiene menor costo de computadora.

El Problema del Control se puede exponer de la siguiente forma: Un sistema físico o proceso está controlado mediante una retroalimentación, por lo tanto, la variable de salida, llamada respuesta, se ajusta mediante una señal de error. La señal de error es una medida de la diferencia entre la respuesta del sistema, determinada mediante un sensor, y la respuesta deseada.

Para procesar la señal de error es necesario un controlador o filtro, por lo cual éste es el problema, se debe escoger el más adecuado para que satisfagan las especificaciones o algunos criterios de control del sistema.

Para escoger el controlador se deben contemplar los siguientes criterios:

1. Rechazo a perturbaciones.

2. Errores Estacionarios.
3. Respuesta Transitoria.
4. Sensibilidad a cambios en los parámetros de la planta.

Para la resolución del Problema del Control se debe:

1. Escoger sensores y actuadores adecuados para actuar sobre la planta.
2. Desarrollar los modelos (ecuaciones) de la planta, sensor y actuador.
3. Diseñar el controlador basado en los modelos desarrollados y en los criterios de control.
4. Evaluar analíticamente el diseño mediante simulación y pruebas del sistema.

#### 1.5 CONTROL POR COMPUTADORA.

La primera computadora utilizada para control, fijaba las referencias de un piloto automático y se puso en marcha en 1954. Una empresa que había competido sin éxito en este producto, modificó su computadora e intentó penetrar en el mercado del control de procesos por computadora. De esta forma, las computadoras de control de procesos fueron, en un principio, resultado de un intento de los fabricantes de computadoras de ampliar su mercado o de vender equipos no aceptados por las Fuerzas Armadas. En 1958 se instaló la primer computadora como supervisor de fábrica, y en marzo de 1959 se empleó una computadora para control en bucle cerrado, aunque a través de controladores analógicos.

Durante la década de 1960 la aplicación de las computadoras digitales se ha extendido desde los sistemas de computación comercial y científicos hasta instalaciones de control de plantas y procesos. En 1962 se montaron las primeras computadoras para el control digital directo de instalaciones de proceso. Por el año de 1966, la aplicación de las computadoras digitales se había ya vuelto un enfoque normal de ciertos tipos de supervisión y control de plantas. Algunas de las primeras e importantes aplicaciones incluyen computadoras para el control de arranque, apagado, regulación y control en línea de instalaciones energéticas que incluyen caldera-turbina-generador. En las

industrias químicas y del petróleo, el control por computadora se ha aplicado a la manufactura del etileno y plantas de amoníaco. La regulación y el control del oxígeno de altos hornos es un ejemplo de la industria siderúrgica. En otras industrias, ejemplos de aplicaciones del control por computadora de diferentes grados de elaboración, incluyen el control de hornos para cemento, máquinas para la fabricación de papel, hornos de vidrio, líneas de ensamble, trabajo de linotipo en periódicos, clasificadores de carros de carga y regulación de ferrocarriles, control ambiental y acabado de textiles.

No todas las primeras aplicaciones de las computadoras digitales al control se consideraron como un éxito. Problemas tales como la ingeniería preliminar inadecuada, dificultades de programación, falta de operadores, instrumentación inadecuada y retrasos en la entrega de equipo causaron considerables dificultades. En algunos casos, se han rechazado computadoras o los proyectos han degenerado a un estado de simple regulación y recolección de datos.

La computadora digital suministra un dispositivo programable, que toma decisiones, para las aplicaciones de control. El equipo puede ir desde máquinas de un propósito general, que operan en *time-sharing* (comparten el tiempo de computadora) con una multitud de tareas, hasta equipo con un propósito especial, diseñado para llevar a cabo una tarea particular. El aspecto más importante de la computadora digital es su habilidad para procesar y transformar grandes cantidades de información de una manera precisa. Además del cálculo, la computadora digital de propósito general puede almacenar, sacar información, hacer decisiones lógicas, adaptarse a condiciones variables y finalmente, aprender.

En las aplicaciones de la computadora digital para el control, la computadora puede programarse para hacer funciones básicas tales como:

1. Recolección de los datos de la planta.
2. Identificación de la dinámica de la planta.
3. Selección de los parámetros de control.
4. Implementación de los algoritmos de control.

5. Implementar los procedimientos de optimización a fin de establecer el control óptimo.
6. Manipulación de las variables que sirven de señal a la planta.

La extensión hasta la cual se utilizan cualquiera de estas funciones en una aplicación dada, depende de la complejidad del sistema, los objetivos operantes y las capacidades de ingeniería.

La figura 1.4 muestra una computadora con las funciones potenciales de entrada y salida. Las órdenes se alimentan a la computadora por medio de cintas o discos flexibles, o mediante la manipulación operador. La información proveniente del proceso puede ser alimentada en forma binaria directamente o en forma analógica mediante un convertidor analógico-digital. La salida de la información de la computadora puede ser en disco flexible, datos impresos, o imágenes en la consola de operación. Puede obtenerse la información binaria, o la información analógica por medio de la conversión de digital a analógica. La capacidad de control de la computadora está regulada esencialmente por el tiempo de acceso o la velocidad de la computadora y de la capacidad de memoria o el número de palabras de una longitud específica que pueden almacenarse.

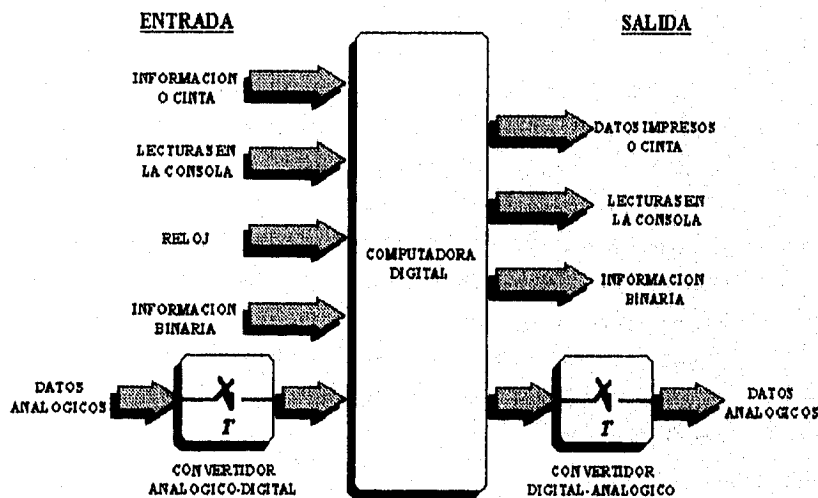


FIGURA 1.4. FUNCIONES DE ENTRADA Y SALIDA DE LA COMPUTADORA DIGITAL.

### 1.5.1 COMPONENTES DEL CONTROL POR COMPUTADORA

Estos componentes están clasificados por las principales funciones que deben darse en el sistema:

La primera de estas funciones consiste en reconocer entradas procedentes del proceso y del entorno externo, a través de detectores o sensores.

**SENSORES:** Los sensores son un tipo de transductores que convierten la información física real, como presión, temperatura, tasa de flujo, posición, etc., en señales de tipo eléctrico. Estas señales eléctricas están relacionadas con las variables físicas que se representan de un modo conocido, de manera que pueden utilizarse para supervisar y controlar un proceso.

Los sensores suelen clasificarse en función de lo que miden.

En el caso de la fabricación por elementos discretos, los sensores utilizados detectan fundamentalmente estados ON/OFF (condiciones que únicamente pueden adoptar dos posibles valores, encendido/apagado).

**ENTRADAS EXTERNAS:** Las entradas externas suelen proceder de un ser humano, con el fin de establecer las condiciones de arranque o de alterar el control de un proceso. Pueden incluir asimismo una parada de emergencia, cambios de velocidad, del tipo de proceso que se va a ejecutar, del número de piezas que se van a fabricar o de instrucciones de mezcla de lotes. A menudo se utilizan diversos tipos de interruptores y conmutadores, controles variables y teclados, para permitir entradas procedentes de los operadores humanos. También pueden medirse determinadas condiciones externas, como la temperatura, presión y humedad del entorno, y ser utilizadas para alterar el proceso de control.

El segundo tipo de funciones generales consiste en alguna clase de acondicionamiento o preparación de la señal.

**PREPARACIÓN DE LA SEÑAL:** A menudo sucede que la señal eléctrica producida por el sensor no está en forma utilizable directamente por el controlador. Una técnica muy común de preparación de la señal es la amplificación de la misma, ya que es posible que la señal no tenga la tensión, la intensidad o la amplitud requerida. Otra técnica bastante frecuente es el filtrado de la señal, a fin de conseguir la frecuencia deseada. Cuando se utiliza un controlador digital, puede ser necesario cambiar la forma de la señal, de una señal continua a una señal representada por niveles digitales, o viceversa.

Las señales eléctricas de salida procedentes del controlador necesitan a menudo un acondicionamiento en lo que afecta a la amplitud, forma o modo de actuar sobre los dispositivos de actuación. De nuevo la amplificación es una técnica muy utilizada, ya que la salida del controlador suele ser una señal de tensión e intensidad bajas, mientras que los dispositivos de actuación requieren habitualmente corrientes y/o tensiones más elevadas.

La tercera función general es la de producir acciones de salida externas o sobre el proceso.

**DISPOSITIVOS DE ACTUACIÓN:** Los dispositivos de actuación, también denominados actuadores, convierten una señal de control eléctrica en una acción física. En el caso de los procesos continuos, estos dispositivos suelen consistir en válvulas de control de flujo, bombas y dispositivos de posicionamiento. En el caso de los procesos por lotes, incluyen motores de velocidad variable, frenos y embragues. En cuanto a los procesos de fabricación de piezas discretas, son utilizados motores paso a paso, conmutadores paso a paso y relés de potencia. Al igual que antes, estas referencias son totalmente generales, y cualquier dispositivo de actuación puede ser utilizado en un proceso cualquiera.

**SALIDAS EXTERNAS:** Habitualmente es deseable que el controlador vaya indicando al operador humano cuál es el estado del proceso, o el valor de algunas variables del mismo. Las salidas externas en este caso corresponderían a medidores, indicadores, pantallas CRT (de tubo de rayos catódicos),

impresoras, timbres de alarma, luces o cualquier otro tipo de dispositivos capaces de interactuar con el operador. Esta es la *interfaz máquina-hombre*. Las salidas pueden dirigirse directamente desde el controlador a un ordenador, para el almacenamiento de datos y el análisis de resultados. En este caso se trata de una *interfaz máquina-máquina*.

La cuarta función general es la de control. El controlador es el cerebro del sistema de control.

**CONTROLADOR:** El controlador recibe las entradas procedentes de los sensores y demás dispositivos externos, realiza cálculos matemáticos y comparaciones lógicas, a fin de decidir qué es lo que debe realizarse a continuación. Luego genera las salidas correctas dirigidas a los dispositivos de actuación, que ejecutarán dicha acción. El controlador suele estar compuesto por combinaciones de amplificadores, filtros y circuitos digitales, incluyendo dispositivos lógicos y microprocesadores.

Los componentes se muestran gráficamente en la figura 1.5.

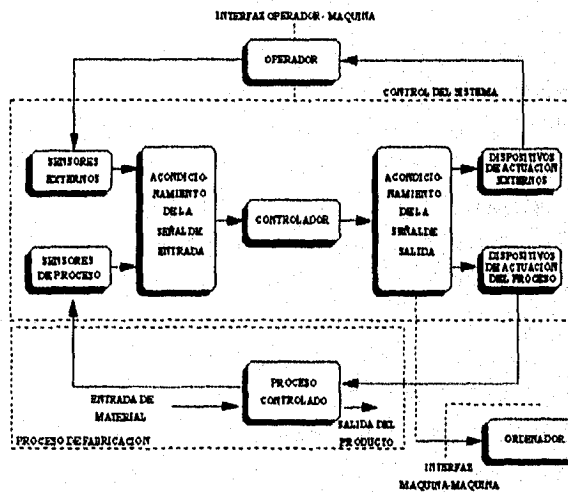


FIGURA 1.5. SISTEMA DE CONTROL POR COMPUTADORA

La implementación del control por computadora de una planta o proceso es un problema desafiante en la ingeniería de sistemas. Debe darse una detallada consideración al análisis de los costos, la definición del sistema, el control, el diseño y la estrategia, así como la instalación e implementación.

## 1.6 TIPOS DE CONTROL

Existen cinco tipos de control por computadora, son divididos de acuerdo a su interacción con el proceso y se explican a continuación:

### 1.6.1 CONTROL SUPERVISORIO

Es la forma más elemental de uso de la computadora. No es la mera adquisición y almacenamiento de datos, sino que incluye su manipulación, elaboración de informes de su evolución después de tratarlos y su comparación con los umbrales fijados, generando las correspondientes señales de alarma en caso de que se sobrepasen éstos. No interviene en el bucle de control del cual es independiente. En la figura 1.6 se muestra un sistema de control supervisorio.

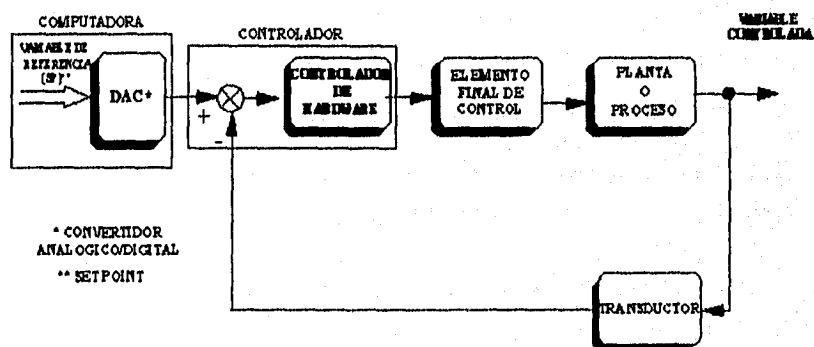


FIGURA 1.6. SISTEMA DE CONTROL SUPERVISORIO



### 1.6.1.1 CONTROL SUPERVISORIO BÁSICO

Este tipo de supervisión supone ya una intervención en el bucle de control. A partir de las medidas que realiza de las variables y de las órdenes superiores, elabora las consignas de los controladores, que pueden seguir siendo de tipo analógico.

### 1.6.2 CONTROL DIGITAL DIRECTO

En este tipo de control la computadora se hace cargo directamente de la adquisición de los datos, elaboración de las órdenes de control y de su envío a los actuadores. En este caso, la computadora es el corazón del bucle de control, ejecutando los algoritmos establecidos. Estos algoritmos de control ya no están restringidos al PID (lazo de control Proporcional Integral Derivativo), que lleva a cabo el controlador analógico, sino que puede disponer de otros e incluir toma de decisiones lógicas y correcciones complejas. Hay que destacar dos tipos de componentes que aparecen en el bucle de control por exigencia de la incorporación de la computadora dentro del lazo: los **convertidores y multiplexores**.

Los **convertidores** transforman las señales analógicas de los sensores en señales digitales para su manejo por la computadora y las señales digitales son transformadas en analógicas para el mando de motores y válvulas.

Los **multiplexores**, de entrada y de salida, permiten a la computadora repartir su potencia y su tiempo entre varios bucles de control, conectando secuencialmente la computadora a los diferentes bucles a su cargo.

En la figura 1.7 se muestra la estructura anterior aplicada a un lazo de control.

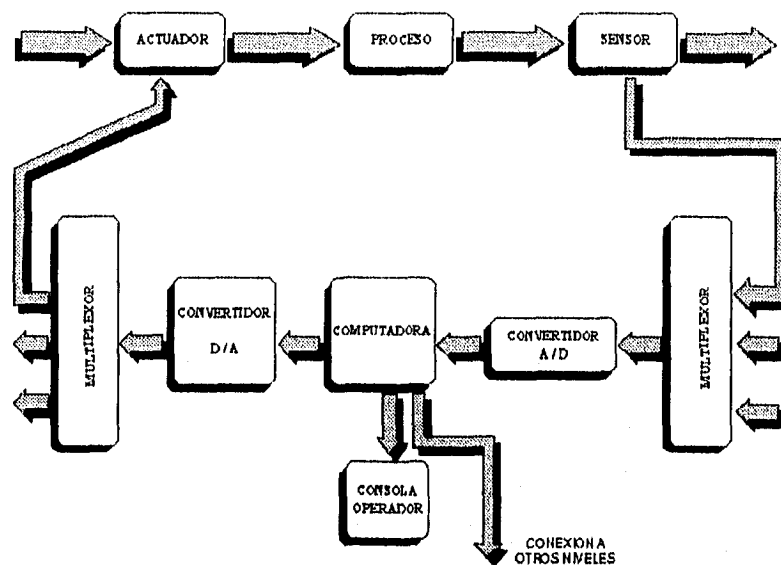


FIGURA 1.7. ESTRUCTURA DE UN CONTROL DIGITAL DIRECTO.

### 1.6.3 CONTROL CENTRALIZADO.

En el Control Centralizado la computadora con todos los componentes auxiliares (convertidores, multiplexores, consola, indicadores), se encuentran situados no junto al proceso, sino en la sala central de control. En ella se concentra toda la instrumentación de control: controladores analógicos y/o digitales y computadora supervisor, estando este núcleo de control comunicado individualmente con cada sensor y actuador situados en el campo. La estructura del control Centralizado se muestra en la figura 1.8.

En algunos casos, el sistema de control digital central posee junto a él un controlador analógico de reserva por razones de seguridad.

Este tipo de instalaciones, aunque satisfactorias porque resuelven el control de forma correcta, presentan serios inconvenientes, siendo el más importante el complejo y gigantesco sistema de comunicaciones, que lleva todas las señales de la planta a la sala de control y devuelve a aquélla todas las señales de control. Es un sistema muy caro que

exige mucha atención y experiencia para su diseño e instalación, tanto para disminuir su costo, como para evitar los problemas de ruidos eléctricos o errores en la transmisión que pueden provocar el fallo total de la planta.

La solución pasa por la descentralización que se acomete de forma progresiva y que culmina con la configuración descentralizada al disponer de microcomputadoras instaladas al pie del proceso, que se encargan de muchas de las tareas antes encomendadas al equipo central. Ha sucedido con la inteligencia lo que ocurrió con la potencia hace muchos años en los talleres: se pasó de un sólo motor gigante (que, mediante poleas u otros tipos de transmisores, movía todas las máquinas), al motor individual para cada máquina o cada movimiento, en cuanto evolucionó la tecnología de fabricación de motores. El primer paso hacia esta descentralización consistió en instalar multiplexores en campo y concentrar en ellos las señales de sensores y actuadores, compartiendo así varios de estos una transmisión entre el proceso y la sala de control. Otro segundo paso fue la instalación de los controladores junto al proceso, transmitiendo entre cada controlador y la sala central solamente la medida y la consigna, trabajando así la computadora central básicamente como supervisor y presentador de información.

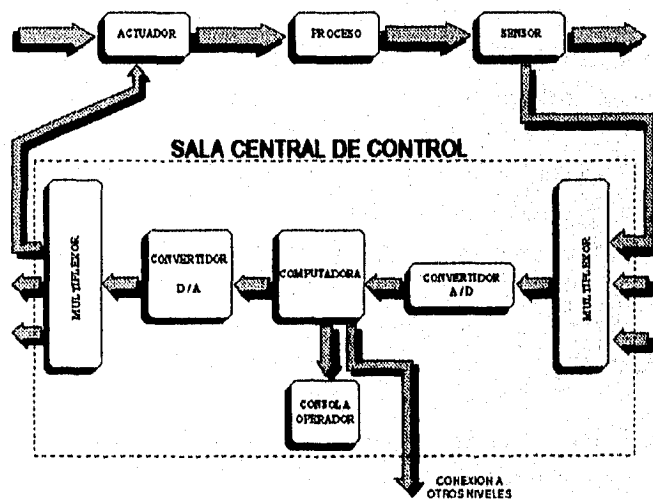


FIGURA 1.8. ESTRUCTURA DE UN CONTROL CENTRALIZADO.

#### 1.6.4 CONTROL DISTRIBUIDO.

El Control distribuido permite el reparto del procesamiento de las diferentes tareas entre los diversos elementos de control. En lugar de disponerse de un ordenador único localizado en un control central, y que realiza todo el procesamiento, cada lazo de control local dispone de capacidad de procesamiento digital. En la figura 1.9 se muestra el diagrama de bloques de un sistema de control distribuido, que permite la existencia de puntos de supervisión tanto remotos como locales. Este sistema permite también la modificación de los valores de referencia, y el encadenamiento de los diferentes lazos locales de control mediante el ordenador central, a través del cual se realizan los intercambios de parámetros entre aquéllos. Toda la información de que dispone el lazo local de control puede pasarse al ordenador de control central, de forma que pueda ser examinada fácilmente por el operador. Al mismo tiempo, cada lazo de control individual funciona de modo independiente respecto de los demás, de forma que ni un fallo en las comunicaciones, ni una sobrecarga del ordenador central, son capaces de afectar el funcionamiento de ninguno de los lazos de control de procesos en tiempo real. El único efecto que tendrían este tipo de problemas sería el de inhabilitar la supervisión de los lazos, o el paso de información entre lazos diferentes, o la realización de modificaciones en el valor de referencia desde el control central.

En este tipo de sistemas de control cada procesador digital resuelve sus propias ecuaciones de control, bajo la supervisión del propio software. La conexión de los sensores y de los dispositivos de actuación con el sistema se realiza a través de convertidores A/D y D/A. En la mayor parte de los casos resulta además muy fácil la adición de nuevas funciones lógicas, así como de nuevos requerimientos, simplemente reprogramando uno o más de los procesadores digitales.

La red de comunicación entre los distintos recursos locales y la sala central de control es un elemento fundamental en este tipo de configuración. Las ventajas de una instalación de este tipo son innumerables: desde la proximidad del control a los procesos, con lo que se supone un aumento de la flexibilidad, como el mismo aumento de la potencia del control, el incremento de la modularidad, la reducción de la complejidad del control facilitando el diseño, la instalación, la puesta en marcha y el

funcionamiento normal, la protección de las comunicaciones y la posibilidad de adoptar medidas de redundancia y actuaciones en caso de fallo y su reparación.

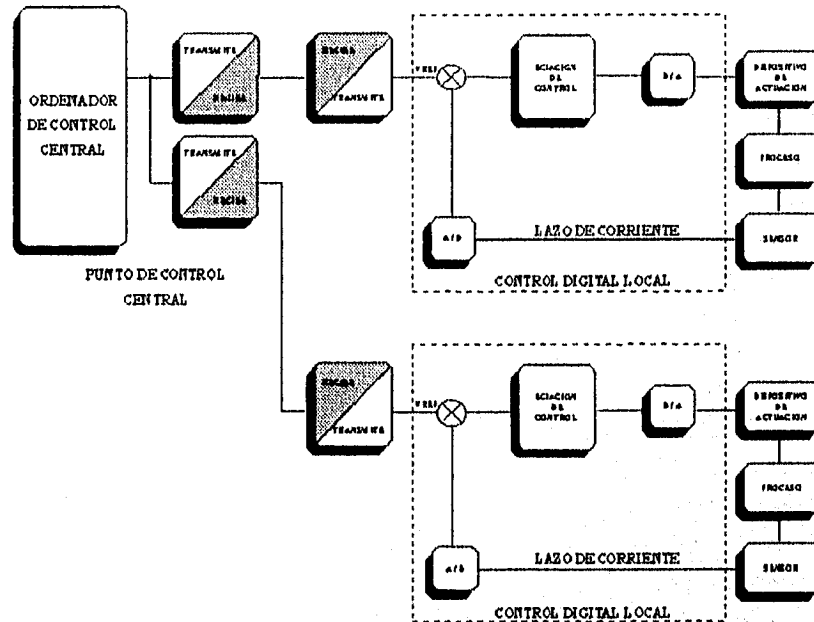


FIGURA 1.9. DIAGRAMA DE BLOQUES DE UN SISTEMA DE CONTROL DISTRIBUIDO.

Puesto que el control distribuido supone una forma o filosofía de abordar la realización de las funciones necesarias y ya que maneja instalaciones de costo muy elevado, debe ofrecer gran fiabilidad y estar preparado para seguir trabajando aún en caso de falla. Además tanto los diferentes recursos como las comunicaciones, deben ser capaces de admitir la máxima carga de trabajo sin degradación en su funcionamiento. Por eso, se ha evolucionado hacia un conjunto de productos normalizados, que se conocen como *módulos* y que pueden interconectarse de muy diversas formas para ajustarse a las necesidades de los procesos de cualquier tamaño. Cada módulo se presenta en una tarjeta de circuito impreso y para comunicarse entre ellos se utilizan protocolos específicos o buses comerciales, mientras que, entre conjuntos de módulos empleados en procesos distintos, se emplean protocolos comerciales normalizados.

Las funciones que ha de realizar un sistema de control están asociadas con uno o varios módulos. Los módulos que realizan una función o varias, si éstas van unidas como es el caso de la adquisición de datos y control de variables, se agrupan en unidades físicas denominadas *odos*. Estos *odos* están unidos entre si y con la computadora supervisor mediante un sistema de comunicaciones basado normalmente en un cable coaxial capaz de transmitir información a 2 MBPS (Megabits/s) a una distancia hasta de 3 Km., tipo red local.

Actualmente, este cable esta siendo sustituido por una fibra óptica. Este sistema de comunicación se conoce como *Data Highway*. En la figura 1.10 se presenta un esquema de un sistema de control de procesos distribuido con tres nodos unidos por líneas de comunicación (*Data Highway*) normales y de reserva (1A y 2A), cada una con su paralela de emergencia (1B y 2B).

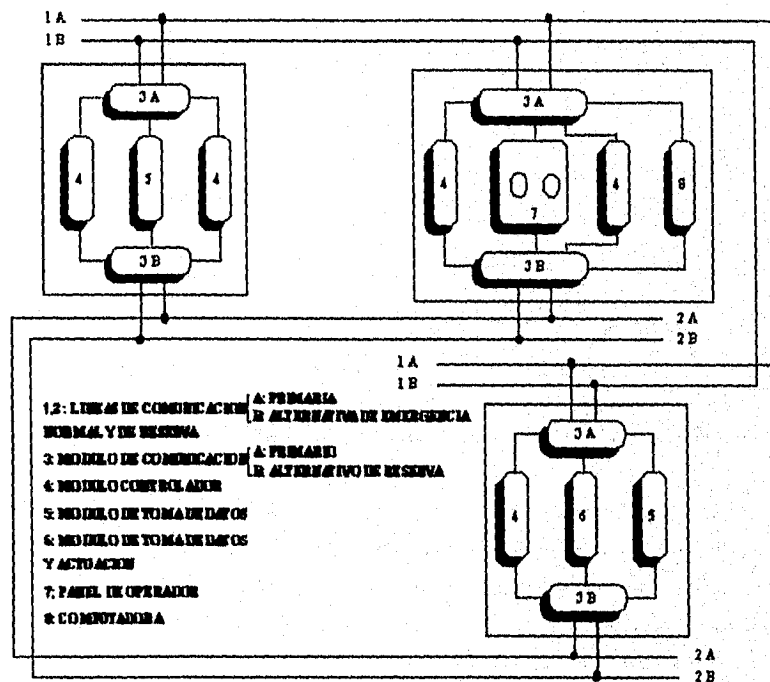


FIGURA 1.10. ESTRUCTURA FÍSICA DE UN SISTEMA DISTRIBUIDO

En la figura 1.11 se presenta la estructura típica de un nodo de adquisición de datos y control. Este está formado por un conjunto de módulos que es completamente reconfigurable según las necesidades de cada punto de la instalación, comunicándose entre sí por un bus local, mientras que el nodo se comunica con el resto de los nodos por la línea de comunicaciones redundante, a la que accede por módulos de comunicación.

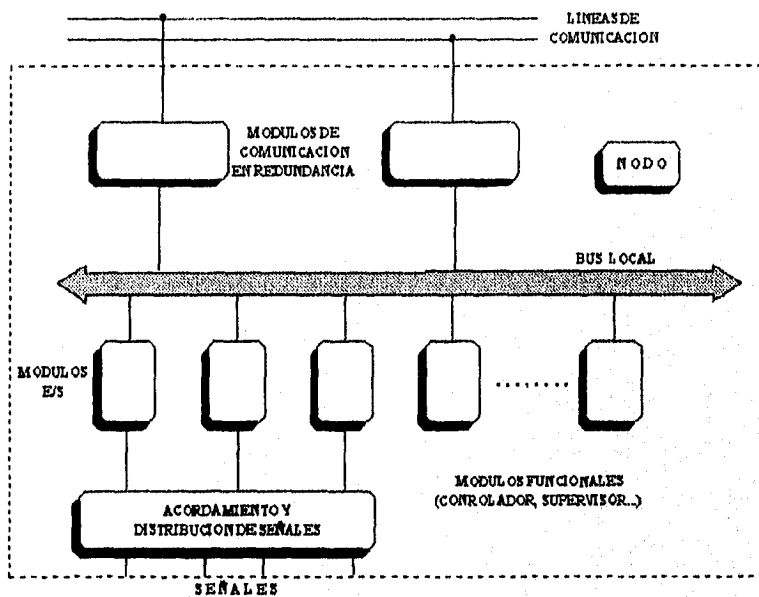


FIGURA 1.11. ESTRUCTURA DE UN NODO DE ADQUISICIÓN DE DATOS Y CONTROL.

La figura 1.12 muestra la estructura de un módulo de entradas analógicas. Además de los convertidores A/D y la interfaz para el bus local del nodo, incluye un microprocesador y una memoria RAM de datos donde se almacenan las medidas que, cuando son pedidas por el controlador, se le envían a través del bus local, generalmente de tipo paralelo, para permitir una rápida transmisión.

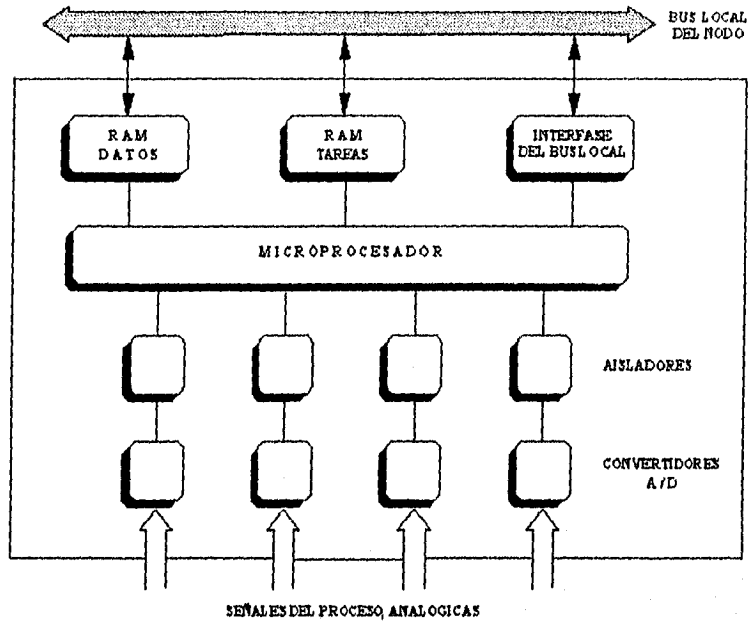


FIGURA 1.12. MODULO DE ENTRADAS ANALÓGICAS.

La clave de los nodos del sistema de control es su flexibilidad para adaptarse a cada aplicación, su manejo de las redundancias y la capacidad para recuperación de errores, reconfigurándose en caso de falla. Esta redundancia, que se da en la comunicación, puede extenderse a cualquiera de los módulos cuando sea necesario, consiguiendo una fiabilidad máxima. Por otra parte, al incrementarse la capacidad de las tarjetas de E/S con la inclusión de microprocesadores y memoria de datos, puede reducirse el costo del cableado a los sensores, sustituyendo los cables dedicados para cada sensor por un elemento emisor/receptor trabajando con estaciones remotas de medida con su propio microprocesador, conectadas mediante una línea digital serie, que además de reducir el costo, reduce también la posibilidad de perturbaciones exteriores sobre las señales. La comunicación del nodo con la línea de comunicación del sistema se hace a través de un módulo de comunicación, redundante, encargado de todo el proceso de envío y recepción de mensajes, y que lleva a cabo la introducción de



formatos, elaboración de protocolos y respuesta a ellos, detección de errores de transmisión, etc., todo según el modelo de comunicación adoptado.

Otros elementos de un sistema distribuido, además de los nodos con toda su flexibilidad y otros módulos que pueden integrarlos, son la estación de operador y la computadora central. La estación del operador es un módulo especial integrable con otros para formar un nodo. Este módulo consiste en una o varias pantallas (normalmente a color), teclados especializados de tipo funcional, uno o varios procesadores y una memoria local secundaria. En ella se almacena una copia, actualizada continuamente, de la base de datos del proceso que se puede encontrar distribuido en los nodos, para permitir su presentación, modificación, elaboración de datos estadísticos y su empleo en la etapa de arranque del proceso. Esta estructura del puesto del operador permite al sistema operar sin necesidad de una computadora supervisora.

En un control distribuido el éxito del control del sistema como tal, reside en un sistema de comunicaciones. Debe ser capaz de soportar configuraciones muy diferentes, una base de datos distribuida entre los nodos y garantizar las prestaciones adecuadas, incluso en los casos de máxima sobrecarga. Aunque la garantía absoluta no existe, se han desarrollado configuraciones del sistema que permiten trabajar en las peores condiciones. Por las variedades existentes en los nodos, el sistema de comunicación debe poseer un control distribuido de sí mismo, de forma que no exista un control único y fijo de la comunicación y los mensajes no deben tener formato fijo. La estructura del mensaje en este caso es:

**[cabecera (fuente, destino, control), cuerpo del mensaje (longitud variable),  
comprobación]**

donde la comprobación consiste en un código de redundancia, normalmente cíclico.

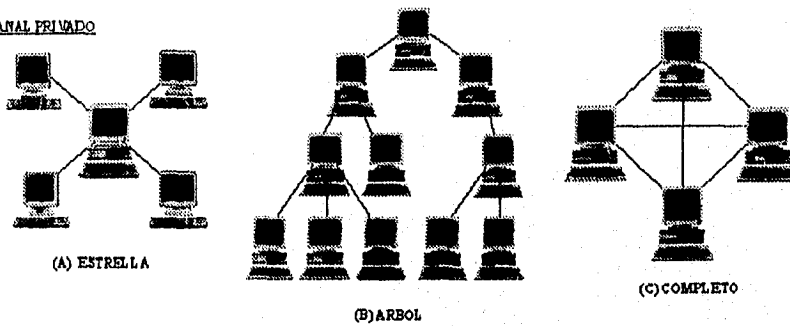
La comunicación entre dos estaciones de una red cualquiera ha de estar sometida a unas reglas que se conocen como *protocolo*. De los 7 niveles de protocolo existentes de la *norma ISO*, solamente se han aplicado las normas existentes para los



dos inferiores: *el físico y el de conexión*, quedando los otros a cargo de la programación específica y reduciéndose así el tiempo de comunicación.

Otro aspecto a considerar es la arquitectura de comunicación. Se agrupa en dos grandes bloques, que se muestran en la figura 1.13. El primero comprende aquellas soluciones en las que cada par de usuarios dispone para su comunicación de un canal privado. Si solamente uno de los usuarios puede comunicarse con todos, tenemos la **estructura de estrella (a)**; si cada uno puede comunicarse con todos sus inferiores, la **estructura es de árbol (b)**; y si todos se comunican entre sí, se trata de una **estructura completa (c)**. En los del segundo bloque, de canales de emisión, hay un canal único compartido por todos, por el que se pueden mandar mensajes globales o específicos. Si este canal es abierto, se denomina **bus (d)**, pero si es cerrado, se trata de un **anillo (e)**.

## CANAL PRIVADO



## CANAL COMPARTIDO

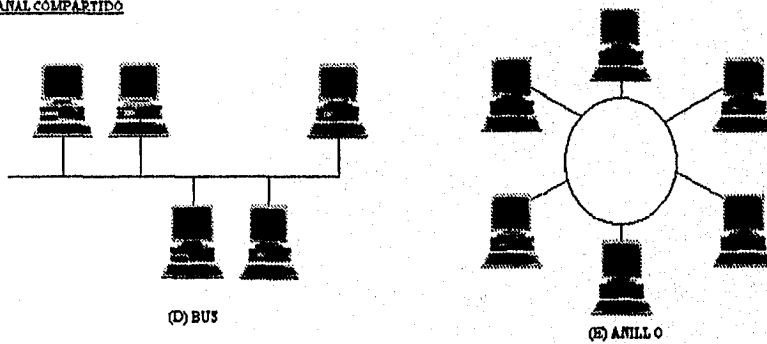


FIGURA 1.13. ARQUITECTURAS DE COMUNICACIÓN.

1.6.5 CONTROL JERÁRQUICO.

El control directo o supervisor de un sistema no constituye al final más que una parte del control por computadora que puede implantarse en un proceso. El resultado es un sistema de control global de todas las actividades de un proceso, mediante una estructura de diversos tipos de computadoras, controladores de regulación, controladores lógicos programables, minicomputadoras y grandes computadoras, e incluso, computadoras personales y estaciones de trabajo inteligentes, conectados todos ellos por distintos tipos de comunicaciones y organizados por niveles con una estructura jerarquizada. En la figura 1.14 se muestra un esquema-cuadro de esta estructura con sus niveles y funciones.

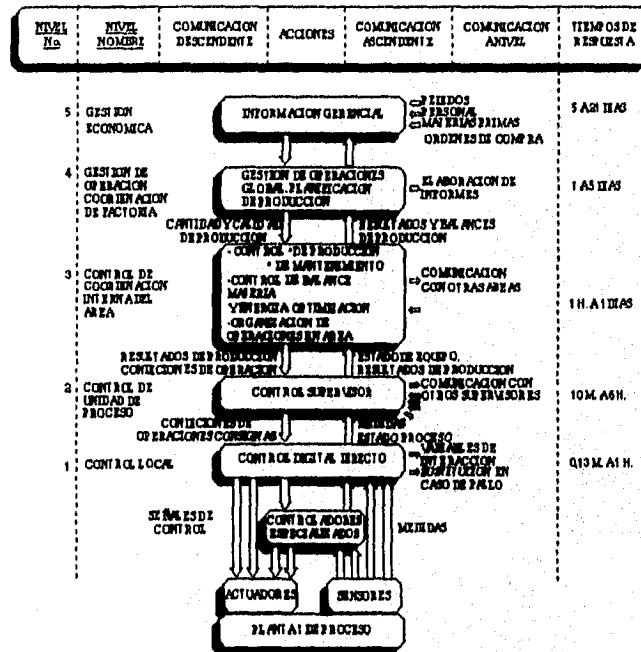


FIGURA 1.14. NIVELES Y TAREAS DE UN SISTEMA DE CONTROL DE PROCESOS JERÁRQUICO.

**EL NIVEL MÁS BAJO, DE CONTROL LOCAL**, tiene como objetivo el control por retroalimentación. La computadora que lo realiza tiene a su cargo la adquisición de datos de los sensores y su corrección y acondicionamiento, la comparación con los límites y generación de alarmas que se envían al nivel superior, y, sobre todo la determinación de la señal a enviar a los actuadores de acuerdo con los resultados del algoritmo de control seleccionado por el nivel superior y de acuerdo con la consigna fijada por éste. Esta computadora (mini o micro) puede controlar, a la vez, varios bucles, realizar controles en cascada o de otros tipos y suele tener encomendada la actuación en arranque y paro.

**EL NIVEL DE SUPERVISIÓN** comprueba también los valores de las variables y sus tendencias, generando las alarmas oportunas si sobrepasan los valores fijados, y tomando las acciones correctivas necesarias para eliminar las tendencias anómalas. Hace el papel, además, de coordinación de varios controles directos de nivel inferior.

**EL TERCER NIVEL, DE COORDINACIÓN DE ÁREA**, tiene como finalidad el control de la producción de toda el área mediante un balance de materiales y energía que se encarga de optimizar (por lo que se le conoce también por ese nombre). Según esto, se establecen las condiciones de operación de cada uno de los procesos del área, que envía al correspondiente control supervisor.

**EL CUARTO NIVEL, DIRECCIÓN DEL PROCESO**, integra todas las áreas y planifica la producción del conjunto, con la secuencia apropiada para las distintas secciones, organizando así la colaboración necesaria entre ellas.

**EL NIVEL SUPERIOR**, finalmente, parte de una información económica: pedidos, recursos, costos y mercado, estableciendo los planes de producción y la política a seguir.

### 1.6.5.1 FUNCIONES EN UN SISTEMA DE CONTROL DE PROCESOS

Un sistema de control de procesos debe realizar tres acciones sobre el sistema de forma automática:

**1a. FUNCIONAMIENTO CONTINUO DEL PROCESO**, manteniendo las condiciones fijadas, supervisión y manejo de las condiciones de emergencia, detectando la aparición de éstas e, incluso, previéndolas y actuado de forma apropiada.

**2a. REALIZACIÓN DE CORRECCIONES**, generación de alarmas de distintos grados e, incluso, la parada automática.

**3a. CAMBIO DE LAS CONDICIONES OPERACIONALES**, en función del mercado, de las características de los productos y del criterio económico correspondiente.

De estas tres acciones, las dos primeras se ejecutan entre los dos niveles inferiores de todo sistema de control de procesos y son el objetivo de los sistemas distribuidos, quedando la tercera, que corresponde a las tareas de los niveles superiores, para ser realizada por computadoras potentes mediante los paquetes de programas apropiados.

Las funciones que realiza un sistema de control de procesos en los dos niveles inferiores son:

#### **ADQUISICIÓN DE DATOS.**

Las señales suministradas por los sensores rara vez pueden ser directamente utilizadas por el sistema para el control siendo necesario someterlas a diversas transformaciones. En primer lugar, las señales analógicas han de muestrearse y digitalizarse de manera periódica y las digitales pueden atacar el sistema de interrupciones o ser, también, muestreadas. Posteriormente, deben someterse a un pretratamiento que comprende la adecuación de unidades, filtrado, estimación de

gradientes y utilizar los valores obtenidos para estimar otros que no pueden medirse directamente. Pueden incluirse aquí la compresión de datos con vistas a su almacenamiento y la agrupación de los valores y parámetros medidos en el mismo instante, con el fin de tener reunidos todos los que definen el estado de un proceso en cada momento. El almacenamiento de los datos también está dentro de esta función.

#### **CONTROL DE LAS VARIABLES DE PROCESO.**

Es el núcleo de todo sistema. Involucra el tratamiento de la información obtenida por la adquisición de datos y lo hace mediante un conjunto de algoritmos, unos cooperadores y otros, mutuamente excluyentes, en cuyo caso hay que decidir cuál se va a emplear. En la base de datos correspondiente debe disponerse de los datos y parámetros que necesita cada algoritmo para funcionar, que pueden estar dados de antemano o ser obtenidos mediante otros programas a partir de medidas realizadas (caso de la identificación o estimación de parámetros).

#### **COMUNICACIÓN CON EL OPERADOR.**

Esta comunicación se realiza mediante dispositivos específicos, situados normalmente en la sala de control, que suministran la información procedente del resto de los componentes del sistema distribuido, después de haberla tratado convenientemente. El sistema debe ser capaz de presentar al operador, con las salvaguardas necesarias, toda la información que pueda contener de las diferentes bases de datos, así como, de forma clara, el estado de funcionamiento del proceso en ese instante, las condiciones de alarma y las tendencias de evolución de las variables, y todo esto tanto de forma global como con distintos niveles de detalle.

Además, el operador debe poder realizar una serie de acciones sobre el sistema, que se pueden agrupar en las siguientes:

1. Inicialización o alteración de los valores de consigna.
2. Modificación de los parámetros de los algoritmos (constantes de los PID).

3. Modificación de los algoritmos (tanto de adquisición como de control) adición de otros nuevos y selección del que se quiere utilizar en cada momento.

Cambio de las supervisiones de control.

### **SUPERVISIÓN.**

Ya que el sistema está constituido por un conjunto de procesadores organizados con una estructura jerarquizada, las funciones más rápidas (y más críticas en el tiempo) se encomiendan a los niveles más bajos, mientras que las más complejas (pero menos críticas en el tiempo), quedan en los niveles superiores. De estas últimas se encarga la computadora potente que se conecta al sistema distribuido. La estructura de estos niveles se concibe y programa de forma específica para cada caso empleando lenguajes de alto nivel. Algunas veces, esta computadora de control supervisor se encarga también de la comunicación con el operador.

### **1.7 DEFINICIÓN DE MONITOREO POR COMPUTADORA.**

Las primeras computadoras fueron empleadas principalmente para el monitoreo y almacenamiento de gran cantidad de datos para su posterior procesamiento, actualmente es posible un control total por computadora.

En un sistema de monitoreo por computadora, el operador humano manipula directamente los controladores del proceso. La computadora recibe cientos de entradas del proceso por medio de una interfaz de operación, la cual es capaz de detectar señales físicas como contactos cerrados y voltajes analógicos, convertir señales analógicas a palabras digitales y convertir palabras digitales a formatos binarios para el proceso por computadora, pero dichos datos no solo son desplegados sino también realiza cálculos y provee datos de corrección al operador. El operador toma decisiones en base al análisis hecho por la computadora y ejecuta las acciones pertinentes.

Este concepto es ilustrado esquemáticamente en la figura 1.15.



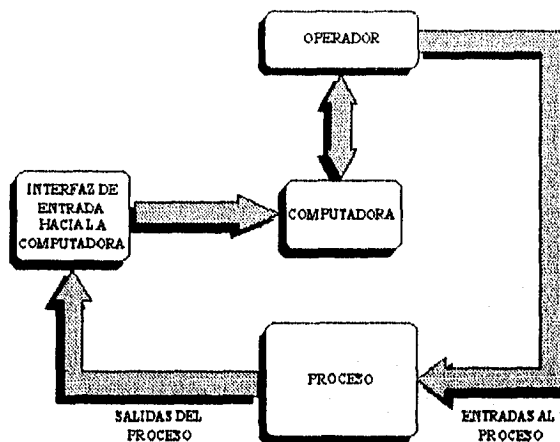


FIGURA 1.15. SISTEMA DE MONITOREO POR COMPUTADORA.

### 1.8 IMPORTANCIA DEL MONITOREO POR COMPUTADORA

El monitoreo es fundamental para la fiabilidad y seguridad del funcionamiento del proceso. Detecta la falla, la localiza y avisa de su existencia.

Para el monitoreo por computadora son utilizadas las llamadas GUI (*Graphic User Interface*), las cuales por medio de gráficos de colores, gráficas de tendencia real e histórica y mensajes de alarma, son capaces de dar al usuario una visión real del proceso, además de proporcionar posibles soluciones a los cambios en las variables controladas por medio de datos de corrección aproximados.

Gracias a estas GUI los operadores son capaces de evitar fallas en el proceso, y en caso de un mal funcionamiento corregir el instrumento o en caso extremo parar la producción.



## 1.9 TIPOS DE MONITOREO POR COMPUTADORA

En forma general se puede dividir el monitoreo en dos grupos de acuerdo al tipo de información que proporcionan, ya que uno lleva información directamente al consejo directivo (Gerencial) y otro lleva información específica para el operador y por tal es llamado monitoreo de campo o proceso.

### 1.9.1 MONITOREO GERENCIAL

El objetivo del monitoreo gerencial es llevar información al consejo directivo de la empresa para lograr una optimización de los recursos tomando las decisiones necesarias para ello.

La información no se recopila directamente del campo, sino que es previamente procesada por la estación monitora del cuarto de control, generalmente esta estación se encuentra en red empleándose por lo regular el cable coaxial, o la Fibra Óptica y protocolos abiertos como TCP/IP, IPX o NETBIOS para su conexión. Otra forma de adquisición de los datos es por medio de discos flexibles o cintas magnéticas.

La información que se lleva al consejo puede ser:

- Totales de producción por día o mes.
- Número de piezas en almacén.
- Materias primas existentes.
- Estadísticas de ventas
- Solicitudes de pedidos y costos.

Esta información generalmente es innecesaria para el operador debido a que la mayoría de las veces no influye en el proceso.

### 1.9.2 MONITOREO DE CAMPO O PROCESO

En el monitoreo de campo o proceso la información es enviada al operador, este recibe la información técnica del proceso como:

- Variables del proceso (temperaturas, presión, flujos, etc.).
- Alarmas.
- Gráficas de tendencia en tiempo real.
- Gráficas de tendencias históricas.
- Cálculos de variables para corregir posibles errores en el proceso.

Generalmente los datos del proceso son recopilados por un controlador o varios controladores conectados en red utilizando protocolos propios como SINEC, TIWAY, PROFIBUS, etc., y llegan a la estación monitorea por un puerto RS-232, 422 o 485.

Se puede pensar que en el Monitoreo y Control se realizan las mismas tareas, pero como se ha explicado anteriormente el monitoreo lleva a cabo funciones diferentes. Es importante mencionar que algunos de sus componentes son los mismos pero son procesados de diferente manera, tal es el caso de las entradas del proceso.

El monitoreo es una parte fundamental del control pero puede hacerse de manera independiente dependiendo de la aplicación.

---

## CAPITULO 2. DESCRIPCIÓN DEL SISTEMA DE MONITOREO Y CONTROL

La mayoría de los sistemas de monitoreo y control se dividen en sistemas con elementos independientes.

En este capítulo se explican de manera general los componentes de un sistema de monitoreo y los componentes de un sistema de control. Así mismo se describen los componentes del sistema instalado en la Terminal Marítima Pajaritos, Ver., dividiendo sus componentes de monitoreo y control.

### 2.1 COMPONENTES DEL SISTEMA DE MONITOREO.

Un sistema de monitoreo esta constituido por:

- Sistema Operativo, para un proceso de control es un Sistema Operativo en Tiempo Real
- Interfaz de Operación, pudiendo ser de cualquier marca dependiendo de las necesidades y de la calidad que se requiera.
- Driver, el cual puede ser de una compañía o bien, si no existe un driver en el mercado para la aplicación que se esta realizando puede ser creado por uno mismo.

#### 2.1.1 SISTEMA OPERATIVO

El Sistema Operativo es un conjunto de programas que se utilizan para administrar los recursos y operaciones en general de un sistema de computo.

Es un programa maestro de control que maneja la computadora y actúa como planificador y agente de transito. Es el primer programa que se carga (copia) en la memoria de la computadora después de que ésta sea encendida, y el núcleo (kernel) del mismo debe estar siempre residente en memoria. El Sistema Operativo puede ser desarrollado por el fabricante del hardware en el que se ejecuta o por una casa independiente de software.

El Sistema Operativo es un componente importante en un sistema informático, ya que determina los estándares para los programas de aplicación que se ejecutarán allí. Todos los programas deben ser escritos para <<hablar>> con el Sistema Operativo.

También llamado *executive* o *supervisor*, el Sistema Operativo desarrolla las siguientes tareas:

#### **ADMINISTRACIÓN DE TRABAJOS**

En computadoras pequeñas, el Sistema Operativo responde a los comandos del usuario y carga en memoria el programa de aplicación requerido para su ejecución. En grandes computadoras, el Sistema Operativo lleva a cabo las Instrucciones de control de trabajos (JCL), las que pueden describir la mezcla de programas que deben ser ejecutadas para una tanda completa.

#### **ADMINISTRACIÓN DE TAREAS**

En computadoras monotareas, el Sistema Operativo no tiene virtualmente ninguna administración de tareas que realizar, pero en computadoras multitareas es responsable de la operación simultánea de uno o más programas (trabajos). Los Sistemas Operativos avanzados poseen la habilidad de asignar prioridades a los programas de modo tal que se ejecute un trabajo antes de otro.

Para proveer de un tiempo de respuesta rápido a los usuarios de terminales, puede asignársele una prioridad baja a los programas que serán procesados por lotes y una prioridad más alta a los programas interactivos. Los Sistemas Operativos más avanzados poseen la capacidad de acelerar o retardar un trabajo por medio de comandos del operador, a esta capacidad se le llama sintonía fina.

La ejecución de tareas múltiples se logra diseñando la computadora para que permita la ejecución de instrucciones al mismo tiempo que los datos entren o salgan de la computadora. En los segundos que tarda un usuario en teclear datos, se pueden ejecutar millones de instrucciones para decenas o aun cientos de otros usuarios. En los milisegundos que tardan los datos en circular hacia o desde el disco, se pueden ejecutar miles de instrucciones para otra tarea.

## ADMINISTRACIÓN DE DATOS

Una de las funciones principales de un Sistema Operativo es la de llevar la cuenta de los datos en el disco; de aquí la denominación DOS (Disk Operating System - Sistema Operativo en Disco). Los programas de aplicación no saben donde se encuentran almacenados los datos o como obtenerlos. Estos conocimientos están contenidos en las rutinas de métodos de acceso del sistema o controladores de dispositivos. Cuando un programa está listo para aceptar datos, le pasa una señal al Sistema Operativo mediante un mensaje codificado. El Sistema Operativo busca el dato y lo entrega al programa. A la inversa, cuando el programa está listo para emitir datos, el Sistema Operativo transfiere estos datos del programa al espacio disponible en el disco.

## ADMINISTRACIÓN DE DISPOSITIVOS

El Sistema Operativo es capaz de administrar todos los dispositivos, no solamente los controladores de disco, sino también las entradas y las salidas a la pantalla de presentación y a la impresora. Mantienen la configuración de los periféricos dentro del Sistema Operativo, estos pueden ser reemplazados cambiando únicamente la rutina que se encarga de controlarlo.

En el mundo de las computadoras personales, el manejo de dispositivos es dejado en manos de los fabricantes de software, estos escriben programas que acceden directamente a la pantalla y la impresora, ya que el DOS o no soporta el dispositivo o agrega mucha carga de procesamiento. Como resultado de ello, los fabricantes de software se han hecho responsables de la provisión de controladores (rutinas) para todos los estándares populares de presentación e impresoras, agregando una carga enorme al desarrollo de sus aplicaciones.

## SEGURIDAD

Los Sistemas Operativos multiusuario mantienen una lista de los usuarios autorizados y proveen protección de claves de acceso (passwords) contra los usuarios no autorizados que pueden intentar introducirse en el sistema. Los Sistemas Operativos más potentes mantienen un registro de la actividad en el sistema y la contabilidad del tiempo de usuario con fines de facturación. También proveen rutinas de respaldo y recuperación para comenzar nuevamente en caso de un fallo del sistema.



## HISTORIA

Los Sistemas Operativos de los años de 1945 eran extraños, el modo usual de operación consistía en que el programador firmaba para tener acceso a un bloque de tiempo en una hoja de registro situada en la pared, después bajaba al cuarto de máquinas, insertaba su tablero enchufable en la computadora y pasaba las siguientes horas esperando que ninguno de los 20,000 tubos de vacío (más o menos) se fundieran durante la ejecución de su programa. Virtualmente todo los problemas eran los cálculos numéricos, como la elaboración de tabla de senos y cosenos.

Al inicio de la década de 1950, la rutina había mejorado un poco con la introducción de las tarjetas perforadas. Ahora era posible escribir programas en tarjetas y leerlos, en vez de utilizar tableros enchufables.

Los primeros Sistemas Operativos como los conocemos actualmente fueron desarrollados a fines de los años cincuenta, para administrar el almacenamiento en cinta y disco, pero los programadores a menudo se sentían más cómodos escribiendo y utilizando sus propias rutinas de Entrada/Salida. A mediados de los sesenta, los Sistemas Operativos se hicieron indispensables para arreglárselas con las complejidades del tiempo compartido y las multitareas. En la actualidad, todas las computadoras de propósito múltiple, desde micro hasta macrocomputadoras, utilizan un Sistema Operativo. Los dispositivos para usos especiales, como herramientas, juegos y juguetes, no lo usan. Generalmente emplean un programa único que ejecuta todas las tareas requeridas de entrada, procesamiento y salida.

A partir de la década de los 80's dos Sistemas Operativos han dominado la escena de la computadora personal: MS-DOS, escrito por Microsoft, Inc., para la IBM PC y otras máquinas que utilizan el CPU basado en el microprocesador Intel 8088 y sus sucesores, y UNIX, que domina en las computadoras personales mayores que hacen uso de la familia de CPU basada en el microprocesador Motorola 68000. Aunque la versión inicial de MS-DOS era relativamente primitiva, versiones subsiguientes han incluido más y más características de UNIX, lo que no es totalmente sorprendente dado que Microsoft es un proveedor importante de UNIX, que usa el nombre comercial de XENIX.

Un avance interesante que empezó a tomar su sitio a mediados de la década de 1980 es el desarrollo de redes de computadoras personales que corren **Sistema Operativos en Red y Sistemas Operativos Distribuidos**. En un Sistema Operativo en

*Red*, los usuarios tienen conocimiento de la existencia de múltiples computadoras y pueden ingresar en máquinas remotas y reproducir archivos de una máquina a la otra. Cada máquina ejecuta su Sistema Operativo local y tiene un usuario propio (o usuarios).

En cambio, *un Sistema Operativo Distribuido*, es aquel que se presenta ante sus usuarios como un sistema uniprocador tradicional, aunque en realidad esté compuesto de múltiples procesadores. En un Sistema Distribuido real, los usuarios no tienen conocimiento de dónde se están ejecutando sus programas o de dónde están ubicados sus archivos; todo esto se maneja en forma automática y eficiente por medio del Sistema Operativo.

#### 2.1.1.1 S.O. BATCH

Cuando se desarrollaron por primera vez los sistemas Batch los trabajos se agrupaban y se ejecutaban, sin embargo, por contar únicamente con tarjetas y cintas el acceso a los datos y a los programas era en forma secuencial y uno a la vez. Cuando se hizo posible el almacenamiento en disco se hizo posible el acceso directo, los sistemas Batch ya no se definían por las agrupaciones sino tomando en cuenta las siguientes características:

- a) La falta de interacción entre el usuario y su trabajo.
- b) En un trabajo con varios pasos alguno de ellos puede depender de los anteriores, por lo tanto la ejecución de un programa dependía de su correcta ejecución.
- c) Los programas tienen que depurarse estáticamente. Un programador no puede modificar su programa que se está procesando sino hasta el final de su compilación.

#### 2.1.1.2 S.O. INTERACTIVO MONOUSUARIO.

Facilita la comunicación en línea entre el usuario y el sistema. El usuario da instrucciones al S.O., recibiendo una respuesta inmediata generalmente se emplea un teclado para la entrada de datos, una impresora o una pantalla para la salida, la mayoría de los sistemas tienen editor de textos interactivo para capturar programas y un depurador interactivo para la ayuda de la depuración del programa.

### 2.1.1.3 S.O. EN TIEMPO REAL

El Sistema Operativo de Tiempo Real es un Programa de control maestro capaz de brindar una respuesta inmediata a las señales de entrada y a las transacciones.

Se utiliza frecuentemente como dispositivo de control en una aplicación especializada. Unos sensores proporcionan los datos al ordenador, este se encarga de analizar los datos ajustando en lo posible los controles para modificar las entradas del sensor.

Los sistemas de Tiempo Real son empleados en algunos sistemas industriales de control, sistemas médicos computerizados y algunos sistemas de tráfico.

Un sistema en Tiempo Real tiene restricciones fijas, el procesamiento tiene que realizarse dentro de los límites definidos, de lo contrario el sistema fallará. Es decir, una respuesta correcta fuera del tiempo determinado es incorrecta.

### 2.1.1.4 S.O. EN TIEMPO COMPARTIDO

Se caracteriza por utilizar la planificación del CPU y la multiprogramación para dotar a cada usuario de una pequeña parte del ordenador compartido, además cada usuario tiene en memoria un programa independiente.

Cuando se ejecuta lo hace normalmente durante un breve tiempo.

### 2.1.2. INTERFAZ DE OPERACIÓN

La principal limitación que se tenía cuando se utilizaban computadoras personales en aplicaciones de control no era el hardware sino el software ya que el DOS sólo puede ejecutar una tarea a la vez. Por ejemplo, la medición de una variable analógica en tiempo real y su almacenamiento en disco duro necesita de catorce interrupciones por parte del DOS para acceder al disco, por lo que se tienen que deshabilitar otras interrupciones. El resultado es una tarea inaceptable de control. Por esto se hicieron modificaciones en las rutinas de interrupción del DOS, pero se tuvieron complicaciones en la estructura del software, en el depurador y en la sincronización del reloj de temporización. Como resultado de las complicaciones se decidió buscar mejores alternativas y continuar el desarrollo de software multitareas.



El Sistema Operativo multitareas iRMX de Intel represento la mejor opción. Este sistema aunado a Windows le proporciono al DOS el control en tiempo real. El iRMX para Windows encapsula al DOS corriendo en cualquier máquina IBM o compatible 386 o superior convirtiéndola en una máquina virtual, permitiéndole ejecutar multitareas, abriendo las puertas para el uso de las computadoras personales en la administración de procesos y aplicaciones de control. Esto permitió que al monitoreo y control de procesos se sumaran nuevas herramientas como los programas de visualización y el software orientado a objetos.

En los nuevos sistemas de control de procesos la presentación de resultados ha dejado de hacerse mediante indicadores de aguja o similares, para ser sustituidos por un pantalla con visualización en color de los reguladores con sus alarmas, incorporando además, la presentación de históricos, tendencias, gráficas esquemáticas del proceso, etc. Esta interfaz no sólo trata de la presentación de resultados de cualquier tipo o con cualquier forma, sino, de ejecución de órdenes del operador, modificando características del sistema, órdenes dadas mediante un lenguaje gráfico o por un teclado especial con teclas funcionales. Estos tipos de teclados se muestran en la figura 2.1



FIGURA 2.1. TECLADOS FUNCIONALES

Otros términos que se utilizan para referirse a las interfaces de operación basadas en microprocesador son: GUI (Graphical User Interface), MMI (Man Machine Interface) y HMI (Human Machine Interface).

En resumen, la interfaz de operación es una combinación de menús, diseños de pantalla, órdenes de teclado, lenguaje de órdenes y pantallas de ayuda, que en conjunto conforman la manera en que el usuario interactúa con la computadora, un ejemplo de esto se muestra en la figura 2.2. El hardware, tal como un ratón o una pantalla táctil, también queda incluido en esta categoría.



FIGURA 2.2. INTERFAZ DE OPERACIÓN

Una interfaz de operación correctamente diseñada es vital para el éxito de un paquete de software. Con el tiempo, el video interactivo, el reconocimiento de voz y la comprensión del lenguaje natural provocarán cambios dramáticos en las interfaces de operación de hoy en día.

Actualmente son utilizadas redes con estaciones de trabajo cliente/servidor, con modernas interfaces gráficas de usuario para los sistemas de automatización de planta.

### 2.1.3. DRIVER DE COMUNICACIÓN.

**DRIVER:** Es un controlador o conductor, también es llamado *device driver* (controlador de dispositivos).

Es una rutina de programa que contiene las instrucciones necesarias para controlar la operación de un dispositivo periférico. Los controladores contienen información detallada acerca de los dispositivos que manejan; por ejemplo, la cantidad de sectores por pistas o el número de líneas de resolución en pantalla. Contienen el código de máquina preciso para activar todas las funciones de cada dispositivo.

Los controladores básicos vienen con el Sistema Operativo, y deben agregarse controladores adicionales cuando se instalan nuevos dispositivos periféricos. Por ejemplo, si usted agrega un ratón o un reproductor de CD ROM a su computadora personal, deberá instalar el controlador apropiado para que el Sistema Operativo sepa cómo manejarlo. Muchos programas de procesamiento de texto, publicaciones de oficina y de gráficos, vienen con controladores para las pantallas e impresoras más comunes con el fin de proveer un control total sobre la presentación e impresión de un documento.

**DRIVER DE COMUNICACIÓN:** Es un dispositivo que provee señales o corrientes eléctricas para activar una línea de transmisión o una pantalla de presentación.

Es un programa de comunicaciones para computadoras personales que administra la transmisión de datos de y hacia el puerto serial de la computadora, así mismo provee de transmisión ASCII simple para transferencias locales. Habitualmente, puede emular varias terminales no inteligentes para poder conectarse a redes de mini y macrocomputadoras.

En la figura 2.3. se presenta un diagrama funcional de un driver de comunicaciones, en él se muestra una interfaz de operación la cual para comunicarse con un equipo de control cualquiera (PLC, DCS, Controlador basado en microprocesador, etc.) utiliza un driver de comunicaciones que adecua el mensaje enviado por la interfaz de operación para que lo reconozca el controlador e interpreta la respuesta del mismo para que la reconozca la interfaz de operación. Físicamente la interfaz de operación y el driver de comunicaciones se ejecutan dentro de la estación de trabajo (PC).

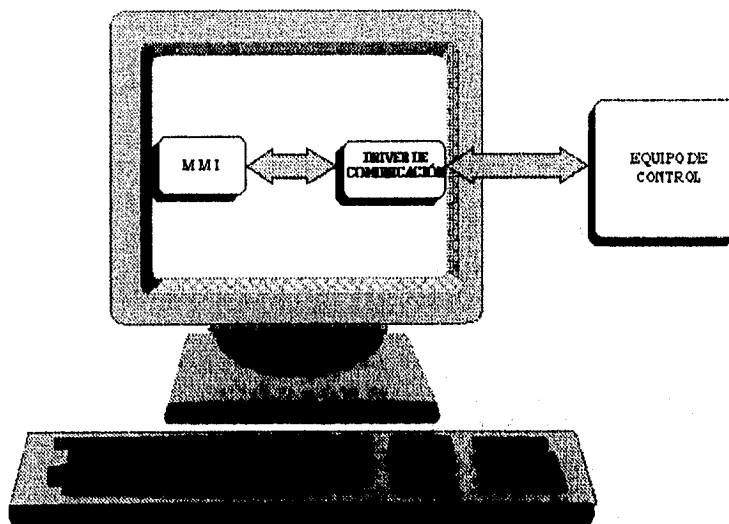


FIGURA 2.3. DRIVER DE COMUNICACIÓN

Generalmente estos drivers de comunicación digital son diseñados para transmitir sobre RS-232.

El sistema de monitoreo que la compañía Daniel Industries Inc. suministro a Petróleos Mexicanos para su terminal marítima Pajaritos está constituido por una minicomputadora HP-A600 que utilizaba el S.O. UNIX, una interfaz de operación y un driver de comunicación desarrollados por Daniel Industries Inc. en lenguaje Fortran.

## 2.2 COMPONENTES DEL SISTEMA DE CONTROL.

Los componentes de un sistema de control son los siguientes:

- Sensores
- Dispositivos de Actuación
- Controlador

Los *sensores* son un tipo de transductores que convierten la información física real en señales de tipo eléctrico. Ejemplo de ellos son los termopares, placas de orificios, medidores de turbina, etc.

Los *dispositivos de actuación* también denominados actuadores convierten una señal eléctrica en una acción física, un ejemplo de ellos son las válvulas de control de flujo, motores de velocidad variable, etc.

Los *controladores* reciben las entradas procedentes de los sensores y realizan los cálculos matemáticos a fin de decidir que es lo que debe realizarse a continuación. Luego genera las salidas correctas dirigidas a los actuadores que ejecutarán dicha acción. Estos controladores pueden ser PLC's, DCS o controladores basados en microprocesador.

El control y la medición del sistema de embarque de petróleo crudo consta de cuatro partes principales que son:

- Patines de Medición.
- Totalizador de Flujo Daniel 2233
- Microcontrolador Daniel MM3000
- Burr Brown TM71

En este sistema los patines de medición contienen todos los sensores y actuadores necesarios para realizar la medición del flujo, los totalizadores de flujo Daniel 2233, el Burr Brown y el Daniel MM3000 conforman el sistema de control.

### 2.2.1. PATINES DE MEDICIÓN DE PETRÓLEO CRUDO.

El patín de medición se encarga de medir el flujo de crudo usando el movimiento del fluido para mover un rotor de turbina, por lo que es un turbo medidor de flujo. La velocidad de rotación de la turbina varía con el flujo. Un sensor magnético detecta el paso de cada hoja del rotor y transmite un pulso de voltaje a un procesador. Entonces, cada pulso individual representa un volumen específico de líquido o gas. Estas señales de salida tienen una respuesta lineal sobre flujos de 0.25 a 16,000 galones por minuto.

Donde el rotor de giro libre está continuamente moviéndose bajo la presión del fluido, las pérdidas por presión no son muy altas. Su exactitud es de  $\pm 0.25\%$  de líquidos y de  $\pm 1\%$  de gas.

## CAPITULO 2. DESCRIPCION DEL SISTEMA DE MONITOREO Y CONTROL

---

La Terminal Marítima Pajaritos se compone de seis subsistemas muelle/boya, de los cuales cinco son usados para el embarque de petróleo crudo. Cada subsistema cuenta con dos patines de medición y un probador. El subsistema restante consta de un patín de medición y de un probador. Este subsistema es utilizado para el embarque de otro tipo de hidrocarburos.

Se tiene un total de once patines de medición con un flujo máximo por patín de 10,000 BPH (Barriles Por Hora).

Cada patín de medición consta de tres o cuatro líneas de medición y una línea común de reserva (spare), la cual es compartida por los dos patines de medición. Los cinco subsistemas de medición permiten cargar simultáneamente hasta diez buques-tanque (dos por cada subsistema).

Los patines de medición poseen un probador para que las mediciones de flujo sean lo mas confiable posible, permitiéndonos realizar ajustes sobre el factor de medición, ya sea de acuerdo a las variaciones sobre las condiciones de flujo, presión y temperatura sobre cada tren de medición\*, o bien ya sea por una solicitud específica sobre un tren determinado.

Cuando un medidor es probado el controlador desviaré el flujo a través del probador abriendo la válvula de retorno del probador, la válvula de desvío del medidor y cerrando la válvula de bloqueo del medidor. Si cualquiera de las acciones de las válvulas falla, la prueba será abortada. Después de que el flujo se ha establecido en el probador, el controlador esperara que las temperaturas del probador y el medidor coincidan con  $\pm 2^\circ$ . Si las temperaturas no coinciden en un lapso de dos minutos, la prueba será abortada.

La prueba consiste en pasar una esfera de hule macizo a través del tubo del probador. Por cada prueba el controlador limpiara el contador interno del probador y arrojara un comando de control para lanzar la esfera nuevamente. Mientras la esfera esta entre los detectores, el controlador grabara los valores de las temperaturas y presiones del medidor y del probador.

En cada paso de la esfera un factor de prueba es calculado. Después de cuatro pruebas completas, el controlador compara los cuatro factores de prueba contra el

---

\* Tren de medición es igual a una línea de medición.

promedio de las cuatro. Si cada uno de los factores de prueba esta dentro del  $\pm 0.02\%$  del promedio, el controlador promedia los datos de las cuatro pruebas y calcula un factor de medición basado en los promedios. Un reporte de la prueba del medidor se imprime documentando los cálculos.

Si los cuatro factores de prueba no están dentro del  $\pm 0.02\%$  de su promedio, los datos de la primera prueba son descartados y otra prueba es realizada. Cuatro pruebas consecutivas dentro de la tolerancia del  $\pm 0.02\%$  son necesarias para una prueba exitosa. Si esto no se cumple en diez pruebas, la prueba es abortada. En tal caso, un reporte de prueba abreviado es impreso mostrando exclusivamente los datos de las tres últimas pruebas.

El número máximo de intentos para una prueba son 10 lanzamientos de la esfera y la tolerancia entre pruebas es del  $\pm 0.02\%$  con respecto al promedio. Estos datos pueden ser cambiados a través del teclado.

Periódicamente se hace una certificación a los patines de medición. La certificación consiste en realizar pruebas de verificación mecánica de componentes, con prioridad en las válvulas de bloqueo de doble sello.

En la figura 2.4 se muestra la estructura física de un patín de medición con su respectivo probador.



FIGURA 2.4 PATÍN DE MEDICIÓN

### 2.2.2 TOTALIZADOR DE FLUJO DANIEL 2233.

Un totalizador de flujo es un dispositivo electrónico basado en microprocesador que permite cuantificar el flujo que circula por un línea de medición.

El totalizador del flujo Daniel 2233 es una computadora de flujo digital basada en microprocesador y es usado con un medidor de turbina en líquidos o medidores de desplazamiento positivo para medir y desplegar el rango del flujo y compensar el flujo total. Opcionalmente para los probadores unidireccionales o bidireccionales la unidad puede acumular y desplegar la densidad y los pulsos netos por prueba de cada medidor.

La interfaz con el operador es un teclado de control de veinticuatro teclas para introducir datos y un display alfanumérico de ocho caracteres. Esta interfaz permite al operador introducir y cambiar parámetros de medición e introducir límites de alarma relacionados a valores críticos de los transductores. Puede seleccionar también el despliegue de los valores de los transductores, rangos de flujo y el flujo total. Opcionalmente el volumen totalizado puede ser desplegado en un contador electromecánico de seis dígitos montado en el panel frontal del instrumento.

El totalizador de flujo Daniel 2233 se encuentra dentro de un gabinete Industrial estándar de Daniel Industries, Inc., de 4 pulgadas de ancho por 8-1/16 pulgadas de alto por 21-5/16 pulgadas de longitud. Estas dimensiones incluyen una fuente de poder de 24 VDC o 115/230 VAC.

El totalizador de flujo Daniel 2233 es mostrado en la figura 2.5.

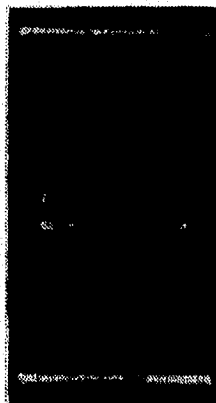


FIGURA 2.5 TOTALIZADOR DE FLUJO DANIEL 2233



**DESCRIPCIÓN DEL TECLADO DEL DANIEL 2233**

1. **ENTER (ENTR).** Introduce a la memoria cualquier dato mostrado en el display alfanumérico
2. **DISPLAY (DSPY).** Llama un dato en blanco cuando la operación esta interrumpida "display timeout".
3. **TECLADO NUMÉRICO.** Números (0-9), punto (.) y signo de menos (-) para introducir datos numéricos o códigos de funciones.
4. **READ (READ).** Al introducir el código de función a través de uno, dos o tres dígitos y presionar el botón READ el totalizador de flujo despliega el dato solicitado (temperatura, flujo bruto, flujo neto, etc.).
5. **FIXED (FXD).** Presionando el botón FXD se desplegarán los datos almacenados por el operador (presión, temperatura, gravedad específica, etc.). El dato es desplegado con un asterisco (\*), el cual indica que éste no está siendo utilizado para realizar cálculos.
6. **VARIABLE (VAR).** Presionando el botón VAR se desplegarán los datos del transductor o del calculo de la computadora. El dato es desplegado con un asterisco (\*), el cual indica que éste no esta siendo utilizado para realizar cálculos.
7. **CLEAR (CLR).** Presionando CLR removerá los datos o valores del display y mostrara "0.0".
8. **COMMAND (CMD).** Introduzca el código numérico con uno, dos o tres dígitos, presione la tecla CMD con lo que se ejecutará el comando especificado. Este comando incluye el despliegue de errores e inicialización de los totales (los totales sólo pueden inicializarse cuando el indicador verde este encendido).
9. **UP ARROW (↑).** Presionando la tecla ↑ la computadora realizará una de las siguientes acciones:
  - a) Lectura de Datos- La tecla ↑ provoca que la computadora muestre el código de datos previo.
  - b) Introduce Datos- La tecla ↑ indica a la computadora que el siguiente dato es exponencial.
10. **DOWN ARROW (↓).** Presionando la tecla ↓ provoca que la computadora muestre el código de datos posterior.

**11 PRINT (PRNT).** Presionando la tecla PRNT se inicia la salida de los datos seleccionados por el operador hacia una impresora externa.

El display del totalizador de flujo esta compuesto por:

- Ocho dígitos Alfanuméricos.
- Dieciséis segmentos de led
- 64 caracteres completos de código ASCII

Se tiene un contador mecánico de seis dígitos sin reset para totales (el operador selecciona para netos, gruesos o masa).

Por ultimo se cuenta con tres indicadores (led) de estado los cuales indican:

*Led Rojo:* Indica un error en la corriente o una condición de alarma. Este led es encendido si el contador Watch-Dog sale de operación o existe otra condición de alarma.

*Led Amarillo:* Indica que una condición de error se presentó después de que la última alarma había sido reconocida desde el teclado, aún cuando la condición de error no haya sido muy grande.

*Led Verde:* Le indica al operador que puede introducir o cambiar datos en la computadora desde el teclado.

### **2.2.3 MICROCONTROLADOR DANIEL MM3000.**

El Microcontrolador Daniel MM3000 provee funciones de medición de líquido y gas confiables y exactas a un bajo costo.

El Microcontrolador Daniel MM3000 sirve para controlar el flujo sobre uno o dos patines de medición realizando la apertura y cierre de las válvulas de paso que permiten la carga de buques-tanque.

Por el Daniel MM3000 pasan los datos de medición de la carga y este los transmite hacia la computadora. El controlador Daniel MM3000 recibe directamente

entradas de campo y calcula volúmenes de flujo, estos volúmenes son reportados a la computadora.

### **CARACTERÍSTICAS**

1. Esta basado en un microprocesador con poderosas capacidades computacionales y de control.
2. Software estándar para medición de gas de acuerdo a las normas AGA-3 (American Gas Association), AGA-5, AGANX-19 (Opcionalmente para ISO - International Standard Organization-).
3. Software estándar para la medición de líquidos de acuerdo a la última edición de API y ASTM (Normas estándares de medición).
4. Gran capacidad de entradas y salidas, incluyendo entradas analógicas, salidas analógicas, acumuladores, entradas digitales, salidas digitales y control de probador.
5. Múltiple medición de totalizadores de flujo
6. Interfaz de operación en el panel frontal.
7. Impresión automática de recibos.
8. Batería de respaldo para memoria RAM de 27 días.
9. Múltiples puertos RS-232 para comunicaciones con periféricos, computador servidor e instrumentos.
10. Algoritmos PID estándar para el control de flujo.

### **DESCRIPCIÓN DEL EQUIPO**

La computadora esta dentro de un gabinete robusto, el cual incluye un gabinete para las tarjetas, tarjeta madre y fuente de poder. El gabinete soporta múltiples tarjetas de circuito impreso, incluyendo CPU, RAM, PROM y tarjetas de Entrada/Salida. Todas las tarjetas en el gabinete están colocadas en ranuras para ser fácilmente removibles

para mantenimiento o reconfiguración. Las señales de Entrada/Salida llegan por conectores montados en el gabinete. El diagrama a bloques del microcontrolador Daniel MM3000 se muestra en la figura 2.6.

Un microprocesador de 16 bits es empleado para la computadora. Múltiples coprocesadores pueden ser utilizados para salvar con rapidez lo realizado. El microcontrolador Daniel MM3000 contiene opciones de hardware que soportan varios microprocesadores matemáticos.

Los tableros de Entrada/Salida son versátiles y modulares. Los tableros tienen un tipo de Entrada/Salida o múltiples tipos de Entrada/Salida dependiendo de su uso. Las Tarjetas individuales pueden configurarse por hardware, software o una combinación de ambas. Las tarjetas de Entrada/Salida no tienen ranuras direccionables.

La capacidad de la memoria puede ser expandida para almacenar grandes programas durante la vida del producto. La porción de la memoria RAM de bajo poder es respaldada por la batería. Existen tarjetas modulares o individuales de memoria que son configuradas para almacenar el estado actual de el proceso en la memoria.

La tarjeta madre esta basada en la arquitectura de multibus. Las características del sistema de bus son las siguientes:

- 16 líneas de datos bidireccionales.
- 20 líneas de direcciones de triple estado.
- Capacidad de multiprocesamiento.
- Múltiples niveles de interrupciones.
- Líneas de prioridad para configuraciones multimaster.

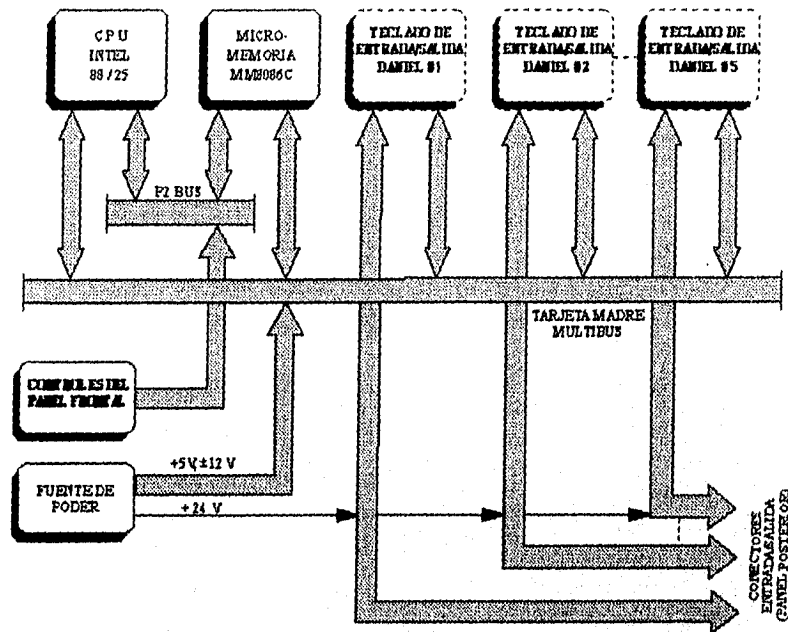


FIGURA 2.6 DIAGRAMA A BLOQUES DE UN MICROCONTROLADOR DANIEL MM3000

El panel frontal del microcontrolador Daniel MM3000 tiene los siguientes controles e indicadores:

1. Llave interruptora con tres posiciones (OFF-ON-LOCK). La llave es irremovible en la posición ON. Cuando la llave está en la posición LOCK, el botón reset y el interruptor del panel no funcionan.
2. RUN LED (Indicador de ejecución). Una luz verde indica que la computadora esta ejecutando instrucciones, no indica que la fuente de poder esta encendida.
3. RESET SWITCH (Interruptor de reinicio). Un interruptor momentáneo que provoca que se reinicie el programa.

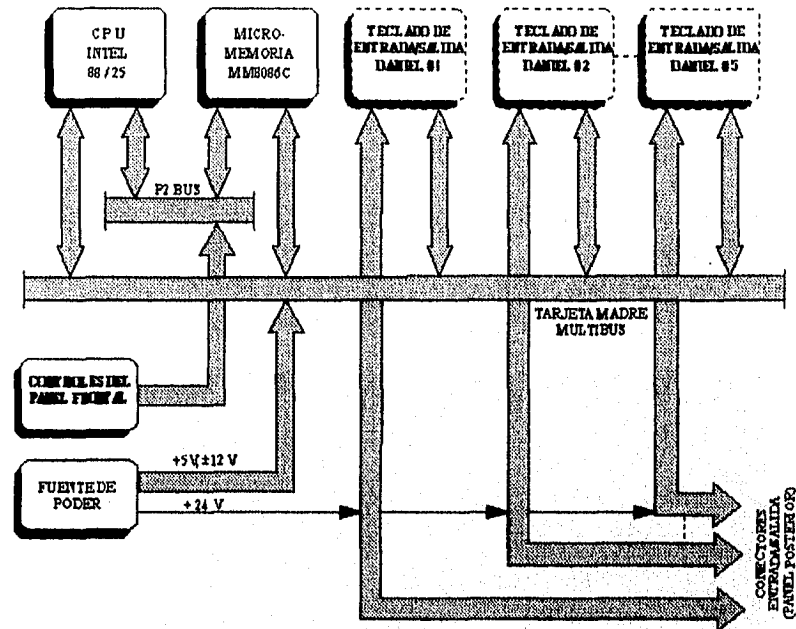


FIGURA 2.6 DIAGRAMA A BLOQUES DE UN MICROCONTROLADOR DANIEL MM3000

El panel frontal del microcontrolador Daniel MM3000 tiene los siguientes controles e indicadores:

1. Llave interruptora con tres posiciones (OFF-ON-LOCK). La llave es irremovible en la posición ON. Cuando la llave está en la posición LOCK, el botón reset y el interruptor del panel no funcionan.
2. RUN LED (Indicador de ejecución). Una luz verde indica que la computadora está ejecutando instrucciones, no indica que la fuente de poder está encendida.
3. RESET SWITCH (Interruptor de reinicio). Un interruptor momentáneo que provoca que se reinicie el programa.

4. **PANEL INTERRUPT** (*Interruptor del panel*). Un interruptor momentáneo que provoca una interrupción.

El microcontrolador Daniel MM3000 mide 19" ancho por 10.5" de alto por 13" de largo, con un peso de 30 libras incluyendo el CPU y las tarjetas de memoria. Requiere una fuente de poder de 90/130 VAC o 180/260 VAC, de 47-63 Hz., 180W, 1.5 A. La temperatura de operación nominal es de 0-55 °C y la temperatura de almacenamiento es de -25°C a +85°C.

En la figura 2.7 se muestra un microcontrolador Daniel MM3000.

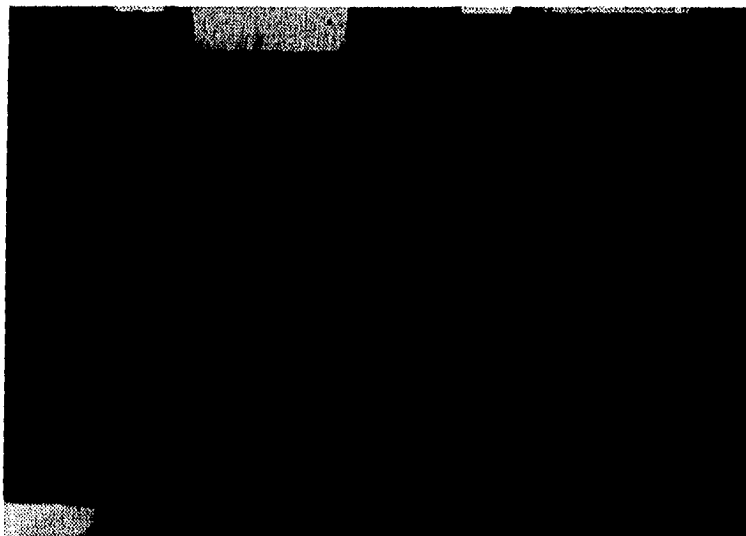


FIGURA 2.7 MICROCONTROLADOR DANIEL MM3000.

#### 2.2.4 BURR BROWN TM71

El Burr Brown modelo TM71 es una "microterminal" alfanumérica, la cual puede ser usada como terminal de entrada y salida de datos locales o remotos para el Daniel MM3000. Esta diseñado para proveer a bajo costo y en un tamaño pequeño una alternativa a la terminal CRT. Es recomendable para aplicaciones con un bajo intercambio de datos.

El panel frontal del Burr Brown TM71 esta compuesto por:

- 80 caracteres en 42 teclas.
- Display de 16 caracteres alfanuméricos.
- 4 diodos emisores de luz (led's) indicadores de estado.

En la figura 2.8 se muestra un Burr Brown TM71.



FIGURA 2.8 BURR BROWN TM71.

#### CARACTERÍSTICAS

- El tamaño de los caracteres es de 0.14".
- Control para dos servidores.
- Conector de 25 pines para transmisión de datos por RS-232C.
- Recepción de datos y funciones de control de módem.
- El puerto puede ser configurado para transmisión/recepción de datos a 110, 300, 600, 1200, 2400, 4800, 9600 y 19200 bps.
- Reseteo remoto de entrada y salida.
- Selección de paridad.

#### CONDICIONES DE ESTADO

*Numeric Keyboard (teclado numérico):* El estado Numeric Keyboard es usado para proveer al operador un teclado numérico ubicado al centro del teclado en el panel frontal. Se activa presionando una vez la tecla NEXT A/N, se desactiva presionándola dos veces.



*Message Waiting (Espera de mensaje):* El estado Message Waiting es la condición que se toma cuando el servidor envía o recibe un mensaje.

*Output Pending (Pendiente de salida):* El estado Output Pending aparece cuando el servidor esta realizando un sondeo por todas sus terminales.

*Input Display (Indicador de entradas):* El estado Input Display indica cuando esta llegando información al buffer de entrada. Esto puede suceder de dos formas:

Cuando se introduce información en modo Disponible o cuando la terminal está en modo de Composición de Mensajes.

## **MODOS DE OPERACIÓN**

Existen tres modos de operación para el Burr Brown TM71 los cuales son:

*Ready:* El modo Disponible es cuando el Burr Brown esta listo y en espera para entrar en operación. El símbolo (^) en el display indica el modo Disponible, las otras posiciones del display aparecerán en blanco. Se entra al modo Disponible cuando cualquier carácter o función es introducido al buffer de entradas desde el teclado, o cuando la terminal es puesta en modo de edición. El indicador Disponible no será visible cuando un mensaje de entrada sea desplegado, sin embargo, la terminal puede estar internamente en modo Disponible.

*Message Composition:* Para acceder al modo de Composición de Mensajes desde el modo Disponible o desde el modo de Edición. Si la terminal está en modo Disponible, introducir cualquier carácter o función desde el teclado causando que la terminal entre al modo de Composición de Mensajes. Se puede entrar al modo de Composición de Mensajes desde el modo de Edición presionando las teclas 2nd y ED/EX.

*Edit:* El modo de edición puede ser usado para modificar el contenido del buffer de salida.

Conociendo los componentes de un sistema de monitoreo y control, solo es necesario saber el funcionamiento actual del sistema y las necesidades (requerimientos) operativas del personal, para poder seleccionar las herramientas adecuadas que satisfagan los requerimientos del nuevo sistema a realizar.

---

## CAPITULO 3. DESARROLLO DEL SOFTWARE

Para llevar a cabo la realización este sistema de monitoreo que sustituye al sistema actual es necesario conocer el funcionamiento del mismo, así como los requerimientos operativos. Una vez que se tienen los elementos necesarios es posible realizar el diseño de las pantallas de operación. Para programar estas pantallas se seleccionó una interfaz de operación gráfica.

Posteriormente se presenta un análisis de las cadenas de respuesta que envía un Daniel MM3000 a las diferentes solicitudes de operación. Basándose en este análisis se desarrollo el driver para la comunicación entre la interfaz de operación seleccionada y el controlador Daniel MM3000.

### 3.1. ANTECEDENTES DEL SISTEMA DE MONITOREO Y CONTROL

La medición y el control automatizado del sistema de embarque de petróleo crudo se realiza a través de un sistema integrado por unidades, basadas en microprocesador denominadas controlador de subsistema o estación *DANIEL MM3000*, que funcionan conjuntamente con una minicomputadora Hewlett-Packard A800 llamada computadora terminal de medición.

La Terminal Marítima Pajaritos está compuesta por seis subsistemas Muelle/Boya, de los cuales cinco son usados para el embarque de petróleo crudo. Cada uno de ellos cuenta con un patín de medición de dos bandas y un probador. El subsistema restante esta compuesto por un patín de medición de una banda y un probador, utilizado para el embarque de otro tipo de hidrocarburos (gasolina, diesel, combustóleo y turbosina).

Cada subsistema dispone de un controlador *DANIEL MM3000* para monitorear el flujo sobre uno o los dos patines de medición que componen dicho subsistema, además de realizar la apertura y cierre de las válvulas de paso que permiten la carga de los buques-tanque.

Cada banda consta de tres o cuatro líneas de medición. Adicionalmente cada subsistema tiene una línea de medición común de reserva (spare), la cual es asignada a una banda o a otra dependiendo del volumen de carga. Los cinco subsistemas de medición permiten cargar simultáneamente hasta diez buques-tanque al mismo tiempo (dos por cada subsistema).

Cada línea de medición esta provista de un totalizador de flujo *DANIEL 2233*, basado en microprocesador, que realiza la medición y control de flujo.

Existe una jerarquía operacional en la cual la computadora Terminal de medición controla la carga de los buques-tanque mediante comandos, estos comandos son enviados al controlador *DANIEL MM3000*, y los datos de medición que recibe desde el campo, son transmitidos hacia la computadora.

Además del sistema automatizado de control, existe un panel gráfico por cada subsistema de medición para permitir la operación manual de las válvulas por medio de botones luminosos (pushbuttons).

Un selector de tres posiciones Indica el modo de operación del gabinete (**MANUAL, CONTROLADOR Y COMPUTADORA**). Los tres niveles de operación son posibles dentro de la jerarquía de control. El modo normal de control es completamente automático por medio de la minicomputadora, los *DANIEL MM3000* y los *DANIEL 2233*. Este modo de operación se denomina modo computadora. Si la minicomputadora falla, la operación automática de cada subsistema de medición es aún posible con los controladores *MM3000* y los *2233's*. Este modo de operación se denomina modo controlador. Si uno de los controladores falla, la operación manual de los patines de medición se realiza por medio del panel gráfico y los *2233's*. Este modo de operación se denomina modo manual, siendo el menos recomendable.

Estas tres formas de operación son explicadas a continuación:

#### **MODO MANUAL**

El modo *MANUAL/LOCAL*, es un modo restringido de operación. En el que las líneas de medición son abiertas y cerradas por la activación de los pushbuttons sobre los actuadores individuales de las válvulas de los patines. El control de flujo es realizado mediante la operación manual de las válvulas de control de flujo del patín de medición. El volumen bruto total es obtenido de los totalizadores modelo *2403*, que están montados en dichos patines.

Aunque el modo de control *MANUAL/LOCAL* es reconocido como un modo de operación válido, generalmente no es aceptado para la operación del sistema, ya que el número de datos disponibles para el personal operativo es muy reducido y todos los datos se obtienen manualmente.

El control de las válvulas operadas eléctricamente es posible en el patín, usando la manivela o los pushbuttons de cada actuador o con los pushbuttons del panel gráfico de cada subsistema de medición. El flujo a través de los tubos de medición es controlado por medio de los pushbuttons localizados en los actuadores de las válvulas de control de flujo.

Si el controlador no está operando, las señales de Viscosidad y BS&W (porcentaje de barro, sai y agua) no están disponibles y no existe pulso de muestreo.

Otra forma de operación manual se da cuando los totalizadores de flujo DANIEL 2233 funcionan y el operador abre y cierra las líneas de medición a través de los pushbuttons localizados en el panel gráfico. En esta forma de operación:

- El operador controla el flujo introduciendo los puntos de ajuste de la razón de flujo dentro de los 2233's a través de los teclados numéricos (BURR-BROWN TM71).
- El operador es capaz de monitorear temperaturas, presiones, gravedades, razones de flujo y volúmenes a través de los paneles de despliegue digital de los 2233's.
- El operador es capaz de inicializar los volúmenes brutos y netos de carga en una línea de medición e introducir factores de medición por medio de los comandos del teclado del 2233.

Cada 2233 automáticamente obtiene el factor de los pulsos de la turbina de medición a la que esta asignado y efectúa correcciones de temperatura y presión para obtener un volumen de carga constante. La razón de flujo y el volumen de carga de cada estación son calculados a mano a partir de los datos proporcionados por los 2233's.

Las únicas condiciones de alarma reconocidas en el modo manual son las alarmas estándares del 2233, tales como: alto flujo, bajo flujo y falla de comparación de pulsos. La presencia de una de estas condiciones de alarma es indicada por una lámpara localizada en la parte frontal del panel del 2233 y por un código de alarma que es mostrado en la pantalla luminosa del 2233 y en la parte superior del gabinete.

### **MODO AUTOMÁTICO**

Este modo de operación es usado solamente si la computadora Terminal de medición no esta funcionando. En este modo las operaciones de carga del buque y las pruebas del medidor son completamente automáticas, los comandos y los datos de entrada son introducidos a través del teclado del controlador de estación (*BURR-BROWN TM71*) y las capacidades de despliegue de datos y reportes están limitadas.

El controlador *DANIEL MM3000* desarrolla todas estas funciones en modo controlador o modo computadora. Sin embargo, esto permite introducir comandos y datos desde el *BURR-BROWN* sólo si el selector se encuentra en posición "controlador". El despliegue de datos solicitados desde el teclado de la computadora sólo es permitido en modo computadora.

Una impresora *DATEL* de 48 columnas montada en el panel es proporcionada para cada estación. Estas impresoras son manejadas normalmente por los controladores y su función es imprimir mensajes de alarma, reportes de prueba y boletas o notas de embarque.

### **MODO COMPUTADORA**

Este modo de operación es el modo normal, las operaciones de carga de buques-tanque y pruebas de medidores son completamente automáticas. El control es desarrollado por los controladores de estación *DANIEL MM3000* actuando en respuesta a los comandos que son transmitidos por medio del teclado de la consola de operación.

Básicamente los controladores de estación *DANIEL MM3000* funcionan exactamente como lo hacen en modo controlador, pero reciben parámetros de lote y comandos de carga desde la consola de operación, en lugar de recibirlos de sus teclados *BURR-BROWN TM71*. En este modo, los controladores imprimen reportes de entrega (*DELIVERY TICKETS*) pero no imprimen reportes de pruebas y alarmas. Estas funciones son asumidas por la consola de operación.

Por cada estación de carga Muelle/Boya, un gráfico se ve en pantalla mostrando los estados actuales de las válvulas (abierto/cerrado), flujos instantáneos, temperaturas y presiones de cada subsistema. Este gráfico muestra adicionalmente todas las líneas de medición de la estación y el probador asociado a ellas. Un desplegado de lote (*BATCH*), un desplegado de prueba de medidor (*PROOF REPORT*) son proporcionados

para cada estación, las alarmas que se presentan en el sistema de control también son desplegadas en un gráfico especial.

En el modo computadora, el operador configura lotes de entrega, modifica parámetros y realiza pruebas sobre las líneas de medición introduciendo comandos a través del teclado de la consola de operación. Un campo de diálogo sobre el monitor de la consola está reservado para este propósito. La introducción de estos comandos se realiza de forma interactiva (pregunta- respuesta).

Se tiene una impresora para su uso en el modo de operación computadora. En este modo se imprimen reportes de entrega (*DELIVERY TICKETS*), reportes de pruebas (*PROOF REPORTS*) y reportes de alarma.

En el modo computadora, los controladores *DANIEL MM3000* son las unidades de control. La consola de operación exclusivamente transmite a ellos parámetros y monitorea el progreso de las operaciones de entrega y de pruebas del medidor.

Tanto en el modo computadora como en el modo controlador de operación, las válvulas son controladas automáticamente y las líneas de medición son abiertas o cerradas por el controlador *DANIEL MM3000* como sea requerido, con el fin de conseguir la cantidad exacta de crudo en una entrega determinada. Los 2233's desarrollan funciones de control de flujo sobre líneas individuales de medición, usando puntos de ajuste para el flujo instantáneo, mismos que son recibidos desde el controlador *DANIEL MM3000*. Las pruebas de líneas de medición se desarrollan automáticamente a través del controlador *DANIEL MM3000*, pero dichas operaciones son iniciadas por comandos transmitidos desde la consola de operación.

Actualmente las actividades de carga de buques-tanque se realizan de forma automática a través de los controladores *DANIEL MM3000* de cada subsistema, es decir, se opera en modo controlador, ésto debido a que la minicomputadora HP-A600 que permitía llevar a cabo el monitoreo y control se encuentra fuera de servicio por daños sufridos en sus componentes y al alto costo de su reparación.

### 3.2 INTERFAZ DE OPERACIÓN.

El funcionamiento normal del sistema se da cuando el monitoreo y control se realiza desde la computadora terminal de medición.

Como se menciona anteriormente este modo de operación se encuentra fuera de servicio, por lo que se solicitó la modernización del sistema de monitoreo y control utilizando la misma filosofía de operación (modo computadora).

Esta modernización implica la utilización de un nuevo equipo de cómputo. Las características de este equipo de cómputo se determinan por el volumen de información que se maneja en el sistema.

El software que maneja la minicomputadora HP-A600 no puede ser recuperado por daños sufridos en el disco duro, por lo cual se desarrollo una nueva interfaz de operación.

Debido a que el operador del sistema permanece por periodos prolongados de tiempo frente a la computadora, los colores que se utilizan para la interfaz de operación no deben ser molestos a la vista del usuario.

Por esto se eligieron los siguientes colores:

- *Blanco*: Como base o fondo de todas las pantallas. Este fue elegido por el contraste que se causa con los demás colores.
- *Negro*: Este color es utilizado para todos los letreros que se despliegan en las pantallas. Esto es porque siempre resalta sobre cualquier color de fondo.
- *Rojo*: Por ser un color que a la vista llama mucho la atención es utilizado para estados de alerta y/o alarmas. En el sistema las alarmas son presentadas en este color al igual que las válvulas de paso cuando se encuentran abiertas.
- *Verde Bandera*: Este color indica un estado normal de operación, esto es que no se necesita poner mucha atención en esta parte del proceso, ya que no existe flujo en este punto.
- *Azul Cielo*: En las válvulas de paso indica transición de abierto a cerrado y viceversa. Al igual que el color anterior no tiene que llamar la atención porque es un estado normal del proceso.

- *Anaranjado Parpadeante*: Indica falla en la transición de las válvulas. No se utiliza el color rojo ya que podría confundirse con la indicación de válvula abierta. Por ser el anaranjado el color más parecido y que puede llamar igual la atención se utilizó éste.

El parpadeo va de azul cielo (estado normal de transición) al anaranjado (estado irregular de transición)

- *Gris*: Este es un color opaco que contrasta con el blanco y permite que resalten los letreros o datos que se muestran sobre él en color negro.
- *Verde Botella*: Este color sirve como base para las botoneras y la barra del sistema. Este color es oscuro, lo cual ayuda a que el título gris resalte.

Para la pantalla de comandos se tiene el gris como base y el verde botella como área de diálogo con el operador y el equipo Daniel MM3000.

Se eligió el uso de botones por la facilidad que representa para el usuario el trabajar con ellos. Esto permite que el sistema sea más amigable, además todos los botones tienen ligados un tecla específica para que el usuario tenga la opción de manejar el sistema con el mouse o con el teclado.

El uso de gráficos en el sistema tiene como objetivo presentar la información de manera clara, así el operador puede consultarla fácilmente además de detectar rápidamente las fallas en el mismo.

### 3.2.1 SELECCIÓN DE LA INTERFAZ DE OPERACIÓN

Se selecciona la interfaz de operación Intouch versión 5.0 de Wonderware debido a que cubría los requerimientos en cuanto al manejo de gráficos, manejo de reportes, manejo de alarmas, se ejecuta en plataforma Windows y posee una capacidad de crecimiento muy alta pues maneja archivos históricos, tendencias, comunicación DDE con otras aplicaciones de Windows y presenta la posibilidad de trabajar en red.

Para la comunicación con equipos de control, Intouch proporciona como utilerías Drivers de comunicación para los equipos de control de los fabricantes más importantes como Allen-Bradley, Siemens, GE-Fanuc, Telemecanique, Modicon, Bristol Babcock, Texas Instruments, etc., y presenta la posibilidad de desarrollar el Driver en lenguaje "C" para la aplicación que se desea en caso de que no exista.





### 3.2.2 DISEÑO DE LAS PANTALLAS DE OPERACIÓN.

Para el diseño de las pantallas de operación se tomo como base las pantallas que se ejecutaban en la HP-A600 por solicitud del personal operativo, esto con el fin de lograr que el personal se adaptara más rápido al manejo del nuevo sistema.

El sistema se compone de una pantalla de presentación y siete pantallas de operación, las cuales se listan a continuación:

- a) GRAPHIC DISPLAY.
- b) PROOF REPORT.
- c) PROVER DISPLAY.
- d) BATCH DISPLAY.
- e) ALARM REPORT.
- f) DELIVERY TICKET.
- g) COMMAND.

#### PANTALLA DE PRESENTACIÓN

La pantalla principal del sistema de supervisión de la Terminal Marítima Pajaritos, Ver., está dividida en las siguientes áreas:

- 1) Área para desplegado de la barra de título localizada en la parte superior de la pantalla, conteniendo el logotipo de Pemex-Refinación y el nombre de la Terminal Marítima de Pajaritos, Veracruz.
- 2) Área para desplegado de la botonera que permite seleccionar la estación de carga (muelle/boya) deseada, consistente de 10 botones (cada uno identificando una estación de carga). Esta botonera se ubica en la parte superior izquierda de la pantalla.
- 3) Área para desplegado de la botonera que permite seleccionar alguna de las siguientes solicitudes:
  - Graphic Display.
  - Proof Report.
  - Prover Display.
  - Batch Display.

- Alarm Report.
- Delivery Ticket.
- Command.

Cada una de las operaciones anteriores tiene un botón asignado para su ejecución, por lo que esta botonera está constituida de siete botones ubicándose en el lado izquierdo de la pantalla.

- 4) Área que presenta la leyenda "PETRÓLEOS MEXICANOS".
- 5) Área de desplegado principal en el cual se muestra la información generada por las diferentes solicitudes listadas en el punto 3.

Esta área se localiza en la parte central de la pantalla.

En la figura 3.1 se muestra la pantalla principal desplegada por el sistema de supervisión:

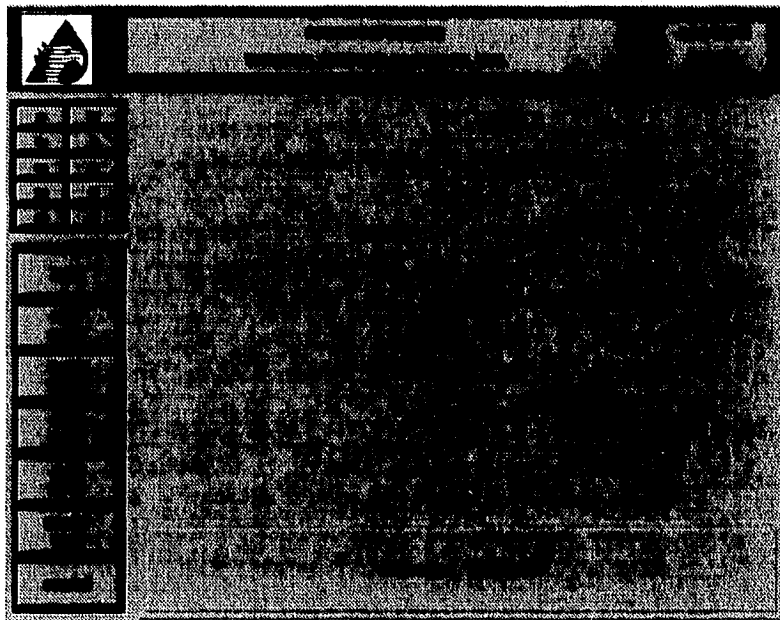


FIG. 3.1 PANTALLA PRINCIPAL



**a) GRAPHIC DISPLAY.**

La solicitud "Graphic Display" proporciona un gráfico a color para cada estación de carga (muelle/boya), en el cual los siguientes datos y elementos son desplegados:

- Las líneas de medición con su respectivo número de identificación localizado a la derecha de la línea correspondiente.
- El flujo instantáneo, temperatura y volumen bruto total correspondiente a cada línea de medición. Estos datos son desplegados a la derecha de la línea de medición.
- El número de identificación de la estación de carga (muelle/boya) correspondiente, desplegado al centro de este gráfico en la parte superior.
- La gravedad específica, presión del cabezal de entrada, presión del cabezal de salida, viscosidad y BS&W de la estación de carga correspondiente, estos son desplegados en la parte superior derecha de este gráfico.
- El flujo instantáneo total y el volumen bruto total de cada estación de carga son desplegados en la parte inferior de este gráfico.
- El probador asociado a la estación de carga que se está desplegando, éste es mostrado junto con su presión y temperatura en la parte inferior derecha de este gráfico.
- Dos válvulas de regreso del probador son desplegadas en la parte inferior de este gráfico.
- Una válvula que representa el estado de las válvulas de entrada al probador de la otra estación de carga. Esta válvula es desplegada a la derecha o a la izquierda del gráfico, dependiendo de que estación de carga se está desplegando (banda "e" o banda "w").
- La línea de medición de reserva (spare), con su respectivo número de identificación.

Las líneas de medición, los símbolos de los filtros canasta y los medidores de flujo aparecen en color azul, cuando una condición de alarma se presente en cualquiera de los filtros canasta se cambian a color anaranjado, y así permanecen mientras perdure la condición de alarma. Cuando dicha condición es retirada, el filtro canasta regresa al color azul.

Las válvulas operadas eléctricamente están coloreadas en color verde para la posición "cerrado", en color rojo para la posición "abierto", en color azul cielo para cuando se encuentran en estado de "transición" y en color anaranjado para una falla en la acción de apertura o cierre.

El símbolo del probador tiene al centro otro símbolo representando una lámpara indicadora, en verde para cuando no se está realizando ninguna prueba. Cuando se este realizando una prueba y esta se encuentra en la fase de lanzamiento (launching), la lampara cambia a color rojo y adicionalmente aparece una leyenda parpadeante con el texto "launching" junto al probador, indicando tal estado de prueba.

Cuando se está realizando una prueba y ésta es abortada por el operador o por cualquier otra circunstancia, en el gráfico junto al probador se despliega una leyenda parpadeante con el texto "Proof Aborted" indicando tal condición.

La válvula que representa el estado de las válvulas de entrada al probador de la estación de carga que no se está desplegando está coloreada de verde cuando todas las válvulas de entrada al probador están cerradas y de rojo cuando cualquiera de ellas o todas están abiertas.

En la figura 3.2 se presenta el desplegado típico de un "Graphic Display" con cuatro líneas de medición y su línea de reserva SPARE la cual se encuentra a la derecha del gráfico junto al probador que le corresponde.

El gráfico que se presenta es el de la estación de medición B-2 de la Terminal Marítima Pajaritos Ver.

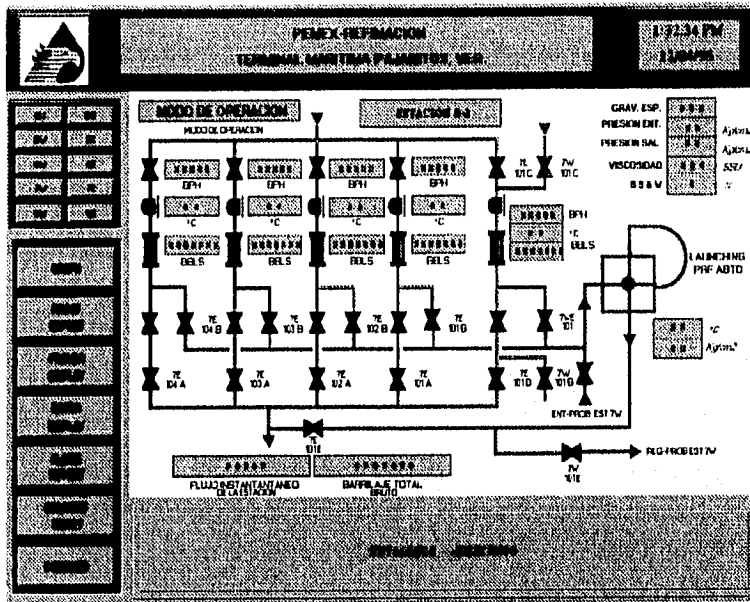


FIG. 3.2 GRAPHIC DISPLAY

**b) PROOF REPORT.**

La solicitud "Proof Report" consiste en un gráfico que muestra un reporte de la última prueba realizada sobre una de las líneas de medición correspondiente a una estación de carga.

El Proof Report esta disponible una vez que la prueba se ha completado. Podrán obtenerse copias de dicho reporte si otra prueba no ha sido iniciada sobre cualquier medidor de alguna estación de carga.

La información del reporte de prueba se despliega al centro de la pantalla en donde se observa el logotipo de PEMEX y los datos de la Terminal Marítima, posteriormente se presentan los siguientes datos:

- Número del medidor.
- Fecha

- Hora
- Los datos de la última prueba, estos datos son: número de intento, pulsos, barriles por hora, temperatura (grados Celsius), presión ( $\text{kg}/\text{cm}^2$ ) y factor de prueba.
- Gravedad específica del medidor.
- Volumen del probador certificado, expresado en barriles.
- Temperatura promedio del probador, expresado en Grados Celsius.
- Presión promedio del probador, expresado en  $\text{Kg}/\text{cm}^2$ .
- Corrección de la temperatura del acero, expresado en CTSP60.
- Corrección de la presión del acero, expresado en CPSPR.
- Corrección de la temperatura del líquido, expresado en CTLP60.
- Corrección de la presión del líquido, expresado en CPLPR.
- Volumen del probador corregido.
- Conteo de pulsaciones de ensayo promedio.
- Temperatura promedio del medidor, expresado en Grados Celsius.
- Presión promedio del medidor, expresado en  $\text{Kg}/\text{cm}^2$ .
- Corrección de la temperatura del líquido, expresado en CTLM60.
- Corrección de la presión del líquido, expresado en CPLMR.
- Conteo de pulsaciones promedio corregido.

- Factor del medidor, expresado en pulsaciones por barril.
- Factor del medidor previo.
- Porcentaje de desviación del factor previo.

A la derecha del reporte existe un botón para imprimir esta información.

En la figura 3.3 se presenta el desplegado típico de un "Proof Report".

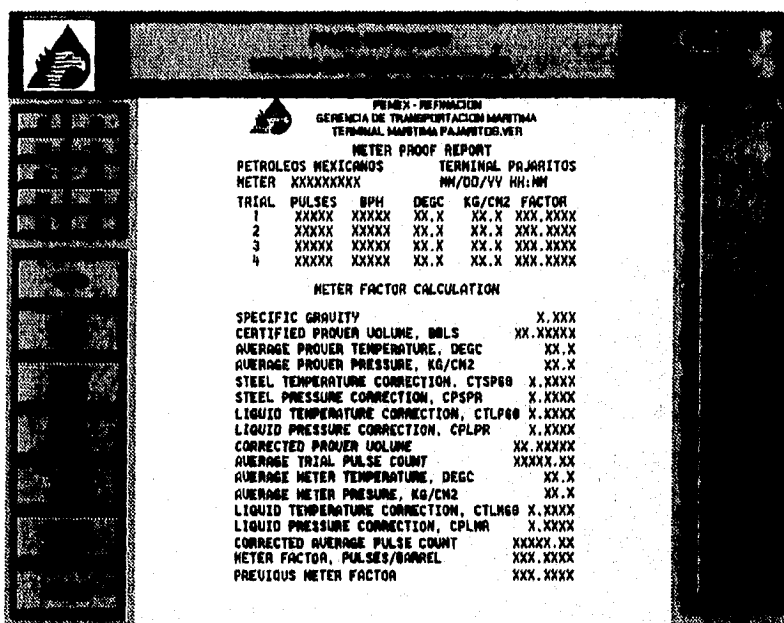


FIG. 3.3 PROOF REPORT

**c) PROVER DISPLAY**

La solicitud "Prover Display" consiste en un gráfico que está disponible por cada estación de carga (muelle/boya), el cual muestra el estado operacional actual del probador.

- En la esquina superior derecha muestra la fecha del sistema y la hora.

- Al centro muestra el nombre de la solicitud (PROVER DISPLAY).
- En la esquina superior derecha se muestra el muelle que se esta cargando.
- El campo METER ON PROOF muestra el medidor actual en prueba, en caso de que ningún probador este actualmente en prueba se muestra el ultimo medidor que ha sido probado.
- El campo METER FLOW RATE muestra el valor de flujo del medidor.
- El campo PROVING PHASE muestra estado actual de la prueba.
- El campo LAUNCH STATUS muestra el estado actual del lanzador.
- El campo METER TEMPERATURE muestra el valor de la temperatura en el medidor.
- El campo HEADER PRESSURE muestra la presión del cabezal de entrada al medidor.
- El campo PROVER TEMPERATURE muestra el valor de la temperatura en el probador.
- El campo PROVER PRESSURE muestra la presión del cabezal de entrada al probador.
- El campo SPECIFIC GRAVITY muestra la gravedad específica.
- En la parte baja de la pantalla se muestran los datos de la prueba en proceso, o los datos de la última prueba, estos datos son : número de intento, pulsos, barriles por hora, temperatura (grados Celsius), presión ( $\text{kg}/\text{cm}^2$ ) y factor de prueba.

A continuación se listan los posibles estados que puede tomar la operación de prueba:

- Idle
- Going on proof
- Temperature delay
- Launching
- Launched
- Counting
- Completed.



En la figura 3.4 se presenta el desplegado típico de un "Prover Display".

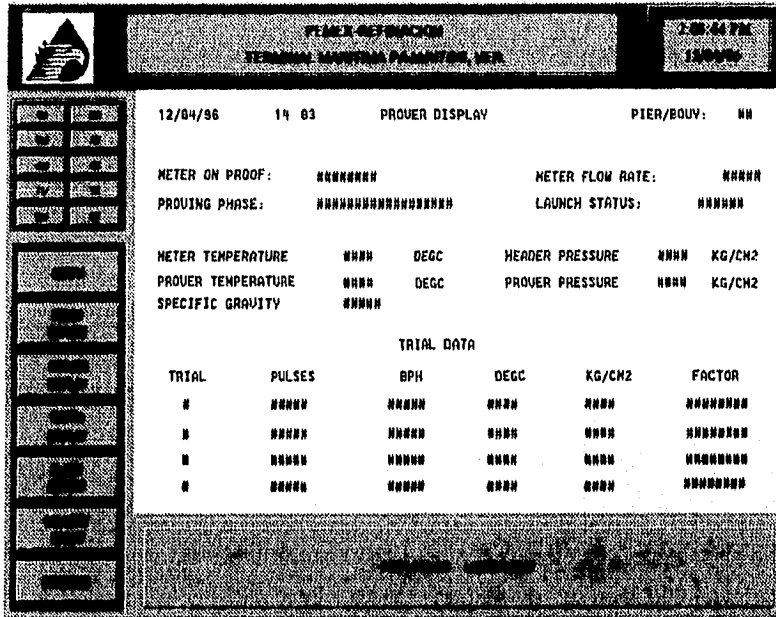


FIG. 3.4 PROVER DISPLAY

**d) BATCH DISPLAY**

La solicitud "Batch Display" es un gráfico proporcionado para cada estación de carga, el cual muestra el estado operacional actual de la carga de un buque-tanque. En este gráfico se muestra:

- Fecha.
- Hora.
- Nombre de la terminal.
- Nombre del muelle/boya.
- Nombre del buque-tanque (tanker).
- Número de embarque (shipment number).
- Estado del lote (batch status).
- Razón del flujo de arranque (start up rate).

- Volumen del lote deseado (batch volume), expresado en barriles brutos.
- Razón de flujo de carga (load rate).
- Volumen de terminado deseado (topoff volume), expresado en barriles brutos.
- Razón de flujo de terminado (topoff rate).
- En la parte baja de la pantalla se muestran los datos del lote de carga, estos datos son: medidor, factor de pulsaciones/bbl, temperatura (Grados Celsius), Punto de ajuste fijado al rango de flujo, rango de barriles por hora, barriles brutos y barriles netos. Además se tiene una suma total de los últimos cuatro datos del medidor.

Todos estos datos son introducidos por el operador cuando configura el lote de entrega; cualquiera de estos datos puede ser cambiado en cualquier instante durante el desarrollo de la entrega hasta que el lote se de por terminado.

El estado del lote (batch) puede ser cualquiera de los siguientes:

- **Idle:** ningún lote esta configurado o concluido.
- **Set up:** el lote ha sido configurado, pero no ha sido iniciado.
- **Starting:** lote iniciado, la razón de flujo de arranque aún no ha sido alcanzada.
- **At start up:** la razón de flujo esta dentro del 1 % de la razón de flujo de arranque.
- **Ramping up:** incrementando hacia la razón de flujo de carga.
- **At load rate:** la razón de flujo esta dentro del 1 % de la razón de flujo de carga.
- **Topping off:** el volumen de terminado ha sido alcanzado, la razón de flujo de terminado es habilitada.
- **Suspending:** el flujo ha sido suspendido, ya sea por que el volumen del lote ha sido alcanzado o por que el operador lo ha solicitado.
- **Suspended:** todas las líneas de medición son cerradas.
- **Terminating:** los totales han sido alcanzados.

En la figura 3.5 se presenta el desplegado típico de un "Batch Display".

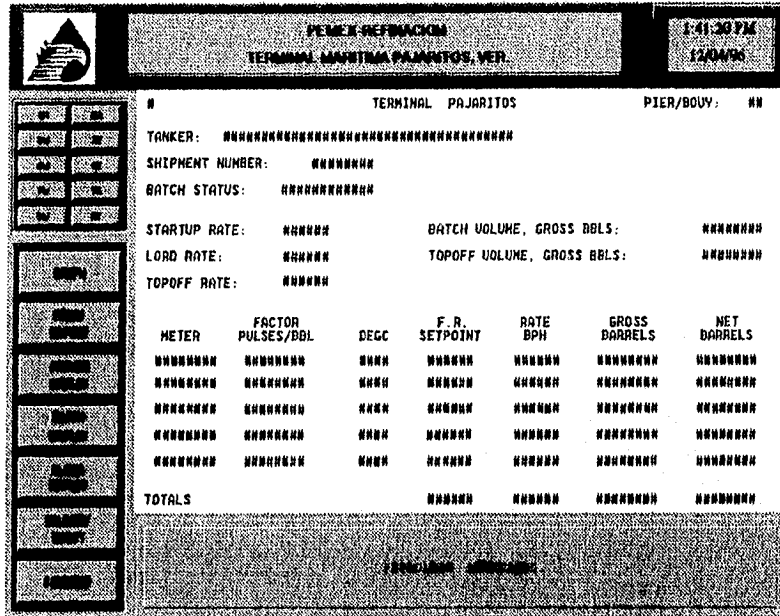


FIG. 3.5 BATCH DISPLAY

e) ALARM REPORT.

La solicitud "Alarm Report" es un gráfico que muestra las alarmas y el estado de todas las estaciones de carga.

El estado y las alarmas de cada estación de carga son desplegadas en pantalla apareciendo en forma consecutiva y agrupadas por estación de carga, es decir, en primer lugar aparece el grupo del estado y las alarmas de una estación de carga determinada, en segundo lugar el grupo del estado y alarmas correspondiente a otra estación de carga y así sucesivamente.

El formato de las alarmas es el siguiente:

- Fecha



- Hora
- Descripción de la alarma.

Adicionalmente en este gráfico se despliegan dos botones para recorrer la información de las estaciones hacia arriba y hacia abajo. Estos dos botones se localizan en la parte inferior de la pantalla. Existe en la misma área otro botón que permite imprimir las alarmas desplegadas en pantalla. Por otro lado el usuario a través del teclado de la computadora puede configurar la cantidad de alarmas que se desplegaran en pantalla, por ejemplo, el usuario puede seleccionar que se despliegue de la alarma No. 35 a la 55. Esta última opción se localiza de igual forma en la parte inferior de la pantalla y en la misma área.

En la figura 3.6 se presenta el desplegado típico de un "Alarm Report "

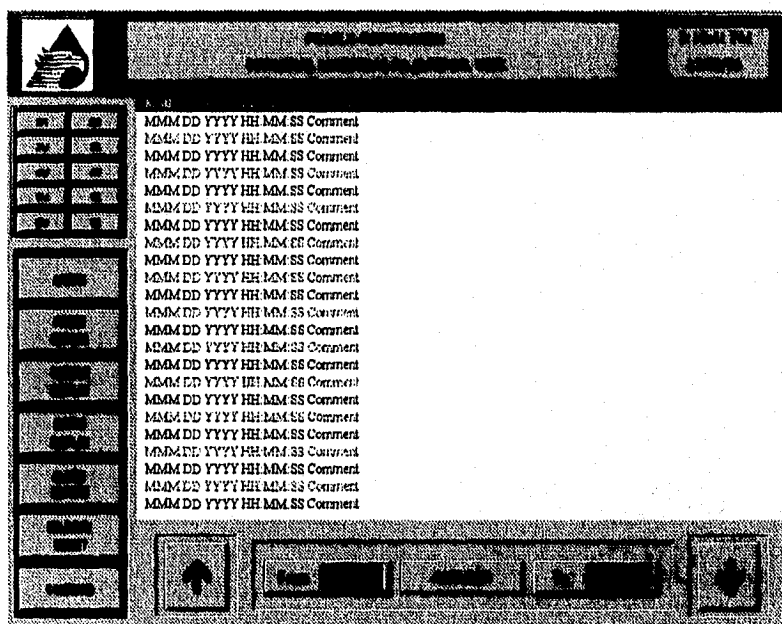


FIG. 3.6 ALARM REPORT

#### f) DELIVERY TICKET

La solicitud "Delivery Ticket" es un gráfico que está disponible para cada estación de carga, el cual muestra un reporte que se genera cada vez que un buque-tanque termina de cargar un lote de petróleo crudo.

La información del reporte de entrega es desplegada al centro de la pantalla en donde se observa el logotipo de PEMEX y los datos de la terminal marítima, posteriormente se presentan los siguientes datos:

- Nombre del buque-tanque (tanker).
- Número de embarque (shipment number).
- Nombre del muelle/boya.
- Fecha.
- Medidor.
- Factor del medidor.
- Temperatura, expresado en Grados Celsius.
- Barriles brutos.
- Barriles netos.
- Porcentaje de Barro, Sal y Agua, expresado en BS&W %.
- Viscosidad, expresado en SSU.
- Presión, expresado en Kg/cm<sup>2</sup>.
- Gravedad específica.
- Total de Barriles Brutos.
- Total de Barriles Netos.

Podrán obtenerse copias de dicho reporte si otro lote (batch) no es configurado en la estación de carga respectiva. Los valores reportados para temperaturas, presiones, BS&W, viscosidad y gravedad específica son promedios del flujo durante el periodo de carga.

En la figura 3.7 se presenta el desplegado típico de un "Delivery Ticket".

**PERMEX - REFINACION**  
**GERENCIA DE TRANSPORTACION MARITIMA**  
**TERMINAL MARITIMA PAJARITOS, VER**

PERMEX - REFINACION  
 GERENCIA DE TRANSPORTACION MARITIMA  
 TERMINAL MARITIMA PAJARITOS, VER

PETROLEOS MEXICANOS      TERMINAL PAJARITOS

TANKER: XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX  
 SHIPMENT NUMBER: XXXXXXXX  
 PIER/BUOY: XX  
 DATE: MM/DD/YY

METER	M.F.	DEGC	GROSS	NET
XXXXXXXX	XXX.XXX	XX.X	XXXXXXXX	XXXXXXXX
XXXXXXXX	XXX.XXX	XX.X	XXXXXXXX	XXXXXXXX
XXXXXXXX	XXX.XXX	XX.X	XXXXXXXX	XXXXXXXX
XXXXXXXX	XXX.XXX	XX.X	XXXXXXXX	XXXXXXXX
XXXXXXXX	XXX.XXX	XX.X	XXXXXXXX	XXXXXXXX

B50M %      X.XX  
 VISCISITY, SSU      XXX.X  
 PRESSURE, KG/CM2      XX.X  
 SPECIFIC GRAVITY      X.XXX  
 TOTAL GROSS BARRELS      XXXXXXX  
 TOTAL NET BARRELS      XXXXXXX

FIG. 3.7 DELIVERY TICKET

**g) COMMAND**

La solicitud "Command" es un gráfico que se proporciona para todas las estaciones de carga. Este gráfico tiene dos campos reservados en la pantalla de la consola de operación para establecer un dialogo pregunta-respuesta entre el usuario y el sistema de supervisión.

El primer campo esta reservado para escribir el comando de entrada hacia el sistema, el segundo campo esta reservado para recibir la repuesta del sistema correspondiente al comando introducido.

En la figura 3.8 se presenta un desplegado típico de la solicitud "Command"

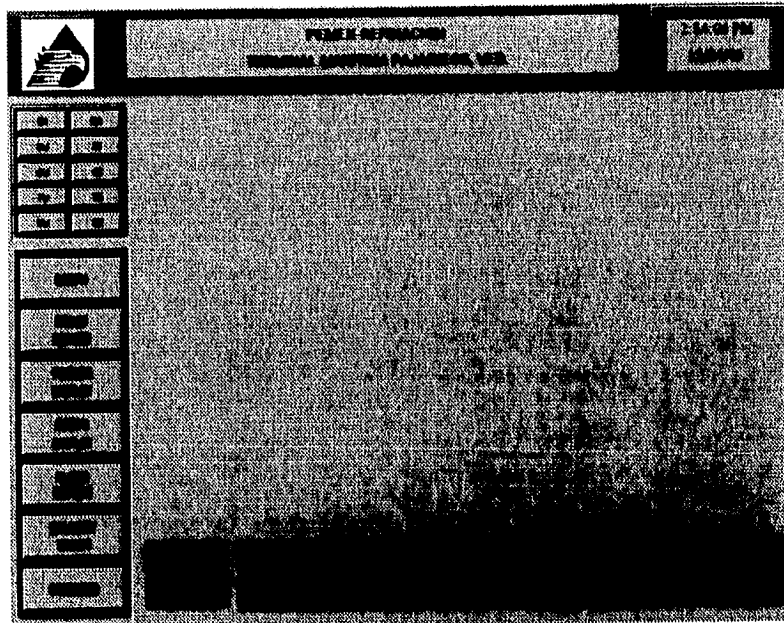


FIG. 3.8 COMMAND

### 3.2.3 PROGRAMACIÓN DE LA INTERFAZ DE OPERACIÓN.

La interfaz de operación Intouch únicamente permite utilizar la sentencia IF-THEN-ELSE para la programación de los scripts (programas en Intouch), además de funciones propias de la interfaz para leer datos, mostrar ventanas, cerrar ventanas, imprimir ventanas, etc., por lo que los programas que a continuación se presentan tienen una lógica de programación básica.

La ventana que presenta la leyenda "PETRÓLEOS MEXICANOS" (alarmas) y la del encabezado no tienen ninguna programación. Esta última presenta la hora y la fecha del sistema por una función de Intouch.

Los programas que se realizaron para la Interfaz de Operación se muestran en el Apéndice A.

### 3.3 DRIVER DE COMUNICACIÓN.

El driver de comunicación se desarrolló en un lenguaje estructurado, debido a que los existentes en el mercado no satisfacían los requerimientos de comunicación entre el equipo PC que contiene la interfaz de operación y el controlador Daniel MM3000.

#### 3.3.1 DISEÑO

Se analizaron las cadenas de texto que el Daniel MM3000 responde a la consola de operación para conocer la forma en que procesa la información y así realizar la programación del driver de comunicación.

Del análisis anterior se determinó que se tiene un funcionamiento diferente para cada solicitud de operación, por lo que se diseñó un Driver de comunicación para cada una.

La solicitud Graphic Display, Proof Report, Prover Display, Batch Display y Delivery Ticket, funcionan de manera similar. Para las solicitudes Alarm Report y Command se tiene un caso especial en el proceso de la información que se recibe y se transmite.

El controlador Daniel MM3000 manda la información de cualquier solicitud en código ASCII por medio del puerto RS-232C. La interfaz de operación Intouch maneja el formato CSV para la recepción de información externa, ya sea de bases de datos o de equipos de control. En este formato la información se encuentra en código ASCII y separada por comas, por lo que el Driver de comunicación que se diseñó tiene la tarea de entregar la información del controlador hacia la interfaz de operación en formato CSV y para la entrega de información de la interfaz de operación hacia el controlador en código ASCII sin ningún formato adicional.

#### Graphic Display

En esta solicitud la cadena de ASCII's que transmite el controlador Daniel MM3000 contiene la información de las dos bandas de medición del patín, el driver de comunicación no procesa la información sino que únicamente le da el formato CSV para que la interfaz de operación la analice e interprete y sea mostrada al operador.



La cadena de información recibida del controlador se compone de caracteres con un significado diferente, dependiendo de la posición en la que se encuentra, estos caracteres se describen a continuación:

EP ENTRADA AL PROBADOR  
 SCE SPARE CABEZAL DE ENTRADA  
 SCS SPARE CABEZAL DE SALIDA  
 RP REGRESO DEL PROBADOR  
 DP ALARMA EN EL FILTRO

GRAPHDIS1	RESPUESTA POSIBLE EN ASCII	SIGNIFICADO
GRAPHDIS1(1)	01 H	• ACCESO AL CONTROLADOR DANIEL MM3000
GRAPHDIS1(2)	36 H	• SOLICITUD GRAPHIC DISPLAY
GRAPHDIS1(3)	0	• FALLA (AMBAS VÁLVULAS)
	1	• ESTACIÓN 1 MEDIDOR 1 ABIERTO Y EP MEDIDOR 1 FALLO
	2	• ESTACIÓN 1 MEDIDOR 1 CERRADO Y EP MEDIDOR 1 FALLO
	3	• ESTACIÓN 1 MEDIDOR 1 ABRIENDO Y EP MEDIDOR 1 FALLO
	4	• ESTACIÓN 1 MEDIDOR 1 FALLO Y EP MEDIDOR 1 ABIERTO
	5	• ESTACIÓN 1 MEDIDOR 1 Y EP MEDIDOR 1 ABIERTO
	6	• ESTACIÓN 1 MEDIDOR 1 CERRADO Y EP MEDIDOR 1 ABIERTO
	7	• ESTACIÓN 1 MEDIDOR 1 ABRIENDO Y EP MEDIDOR 1 ABIERTO
	8	• ESTACIÓN 1 MEDIDOR 1 FALLO Y EP MEDIDOR 1 CERRADO
	9	• ESTACIÓN 1 MEDIDOR 1 ABIERTO Y EP MEDIDOR 1 CERRADO
	:	• CERRADO (AMBAS VÁLVULAS)
	:	• ESTACIÓN 1 MEDIDOR 1 ABRIENDO Y EP MEDIDOR 1 CERRADO
	<	• ESTACIÓN 1 MEDIDOR 1 FALLO Y EP MEDIDOR 1 ABRIENDO
	=	• ESTACIÓN 1 MEDIDOR 1 ABIERTO Y EP MEDIDOR 1 ABRIENDO
>	• ESTACIÓN 1 MEDIDOR 1 CERRADO Y EP MEDIDOR 1 ABRIENDO	
?	• ESTACIÓN 1 MEDIDOR 1 Y EP MEDIDOR 1 ABRIENDO	
	0	• FALLA (AMBAS VÁLVULAS)
	1	• ESTACIÓN 1 MEDIDOR 2 ABIERTO Y EP MEDIDOR 2 FALLO
	2	• ESTACIÓN 1 MEDIDOR 2 CERRADO Y EP MEDIDOR 2 FALLO
	3	• ESTACIÓN 1 MEDIDOR 2 ABRIENDO Y EP MEDIDOR 2 FALLO
	4	• ESTACIÓN 1 MEDIDOR 2 FALLO Y EP MEDIDOR 2

ESTA TESIS NO DEBE  
 SALIR DE LA BIBLIOTECA

GRAPHDIS1(4)	5	ABIERTO
	6	• ESTACIÓN 1 MEDIDOR 2 Y EP MEDIDOR 2 ABIERTO
	7	• ESTACIÓN 1 MEDIDOR 2 CERRADO Y EP MEDIDOR 2 ABIERTO
	8	• ESTACIÓN 1 MEDIDOR 2 ABRIENDO Y EP MEDIDOR 2 ABIERTO
	9	• ESTACIÓN 1 MEDIDOR 2 FALLO Y EP MEDIDOR 2 CERRADO
	:	• ESTACIÓN 1 MEDIDOR 2 ABIERTO Y EP MEDIDOR 2 CERRADO
	;	• CERRADO (AMBAS VÁLVULAS)
	<	• ESTACIÓN 1 MEDIDOR 2 ABRIENDO Y EP MEDIDOR 2 CERRADO
	=	• ESTACIÓN 1 MEDIDOR 2 FALLO Y EP MEDIDOR 2 ABRIENDO
	>	• ESTACIÓN 1 MEDIDOR 2 ABIERTO Y EP MEDIDOR 2 ABRIENDO
	?	• ESTACIÓN 1 MEDIDOR 2 CERRADO Y EP MEDIDOR 2 ABRIENDO
GRAPHDIS1(5)	0	• ESTACIÓN 1 MEDIDOR 2 Y EP MEDIDOR 2 ABRIENDO
	1	• FALLA (AMBAS VÁLVULAS)
	2	• ESTACIÓN 1 MEDIDOR 3 ABIERTO Y EP MEDIDOR 3 FALLO
	3	• ESTACIÓN 1 MEDIDOR 3 CERRADO Y EP MEDIDOR 3 FALLO
	4	• ESTACIÓN 1 MEDIDOR 3 ABRIENDO Y EP MEDIDOR 3 FALLO
	5	• ESTACIÓN 1 MEDIDOR 3 FALLO Y EP MEDIDOR 3 ABIERTO
	6	• ESTACIÓN 1 MEDIDOR 3 Y EP MEDIDOR 3 ABIERTO
	7	• ESTACIÓN 1 MEDIDOR 3 CERRADO Y EP MEDIDOR 3 ABIERTO
	8	• ESTACIÓN 1 MEDIDOR 3 ABRIENDO Y EP MEDIDOR 3 ABIERTO
	9	• ESTACIÓN 1 MEDIDOR 3 FALLO Y EP MEDIDOR 3 CERRADO
	:	• ESTACIÓN 1 MEDIDOR 3 ABIERTO Y EP MEDIDOR 3 CERRADO
	;	• CERRADO (AMBAS VÁLVULAS)
	<	• ESTACIÓN 1 MEDIDOR 3 ABRIENDO Y EP MEDIDOR 3 CERRADO
	=	• ESTACIÓN 1 MEDIDOR 3 FALLO Y EP MEDIDOR 3 ABRIENDO
	>	• ESTACIÓN 1 MEDIDOR 3 ABIERTO Y EP MEDIDOR 3 ABRIENDO
?	• ESTACIÓN 1 MEDIDOR 3 CERRADO Y EP MEDIDOR 3 ABRIENDO	
	0	• ESTACIÓN 1 MEDIDOR 3 Y EP MEDIDOR 3 ABRIENDO
	1	• FALLA (AMBAS VÁLVULAS)
	2	• ESTACIÓN 1 MEDIDOR 4 ABIERTO Y EP MEDIDOR 4 FALLO

GRAPHDIS1(6)	3	FALLO
	4	• ESTACIÓN 1 MEDIDOR 4 ABRIENDO Y EP MEDIDOR 4 FALLO
	5	• ESTACIÓN 1 MEDIDOR 4 FALLO Y EP MEDIDOR 4 ABIERTO
	6	• ESTACIÓN 1 MEDIDOR 4 Y EP MEDIDOR 4 ABIERTO
	7	• ESTACIÓN 1 MEDIDOR 4 CERRADO Y EP MEDIDOR 4 ABIERTO
	8	• ESTACIÓN 1 MEDIDOR 4 ABRIENDO Y EP MEDIDOR 4 ABIERTO
	9	• ESTACIÓN 1 MEDIDOR 4 FALLO Y EP MEDIDOR 4 CERRADO
	:	• ESTACIÓN 1 MEDIDOR 4 ABIERTO Y EP MEDIDOR 4 CERRADO
	;	• CERRADO (AMBAS VÁLVULAS)
	<	• ESTACIÓN 1 MEDIDOR 4 ABRIENDO Y EP MEDIDOR 4 CERRADO
	=	• ESTACIÓN 1 MEDIDOR 4 FALLO Y EP MEDIDOR 4 ABRIENDO
	>	• ESTACIÓN 1 MEDIDOR 4 ABIERTO Y EP MEDIDOR 4 ABRIENDO
	?	• ESTACIÓN 1 MEDIDOR 4 CERRADO Y EP MEDIDOR 4 ABRIENDO
	?	• ESTACIÓN 1 MEDIDOR 4 Y EP MEDIDOR 4 ABRIENDO
GRAPHDIS1(7)	0	• FALLA (AMBAS VÁLVULAS)
	1	• ESTACIÓN 2 MEDIDOR 1 ABIERTO Y EP MEDIDOR 1 FALLO
	2	• ESTACIÓN 2 MEDIDOR 1 CERRADO Y EP MEDIDOR 1 FALLO
	3	• ESTACIÓN 2 MEDIDOR 1 ABRIENDO Y EP MEDIDOR 1 FALLO
	4	• ESTACIÓN 2 MEDIDOR 1 FALLO Y EP MEDIDOR 1 ABIERTO
	5	• ESTACIÓN 2 MEDIDOR 1 Y EP MEDIDOR 1 ABIERTO
	6	• ESTACIÓN 2 MEDIDOR 1 CERRADO Y EP MEDIDOR 1 ABIERTO
	7	• ESTACIÓN 2 MEDIDOR 1 ABRIENDO Y EP MEDIDOR 1 ABIERTO
	8	• ESTACIÓN 2 MEDIDOR 1 FALLO Y EP MEDIDOR 1 CERRADO
	9	• ESTACIÓN 2 MEDIDOR 1 ABIERTO Y EP MEDIDOR 1 CERRADO
	:	• CERRADO (AMBAS VÁLVULAS)
	;	• ESTACIÓN 2 MEDIDOR 1 ABRIENDO Y EP MEDIDOR 1 CERRADO
	<	• ESTACIÓN 2 MEDIDOR 1 FALLO Y EP MEDIDOR 1 ABRIENDO
	=	• ESTACIÓN 2 MEDIDOR 1 ABIERTO Y EP MEDIDOR 1 ABRIENDO
	>	• ESTACIÓN 2 MEDIDOR 1 CERRADO Y EP MEDIDOR 1 ABRIENDO
?	• ESTACIÓN 2 MEDIDOR 1 Y EP MEDIDOR 1 ABRIENDO	
0	• FALLA (AMBAS VÁLVULAS)	

GRAPHDIS1(8)	1	• ESTACIÓN 2 MEDIDOR 2 ABIERTO Y EP MEDIDOR 2 FALLO
	2	• ESTACIÓN 2 MEDIDOR 2 CERRADO Y EP MEDIDOR 2 FALLO
	3	• ESTACIÓN 2 MEDIDOR 2 ABRIENDO Y EP MEDIDOR 2 FALLO
	4	• ESTACIÓN 2 MEDIDOR 2 FALLO Y EP MEDIDOR 2 ABIERTO
	5	• ESTACIÓN 2 MEDIDOR 2 Y EP MEDIDOR 2 ABIERTO
	6	• ESTACIÓN 2 MEDIDOR 2 CERRADO Y EP MEDIDOR 2 ABIERTO
	7	• ESTACIÓN 2 MEDIDOR 2 ABRIENDO Y EP MEDIDOR 2 ABIERTO
	8	• ESTACIÓN 2 MEDIDOR 2 FALLO Y EP MEDIDOR 2 CERRADO
	9	• ESTACIÓN 2 MEDIDOR 2 ABIERTO Y EP MEDIDOR 2 CERRADO
	:	• CERRADO (AMBAS VÁLVULAS)
	;	• ESTACIÓN 2 MEDIDOR 2 ABRIENDO Y EP MEDIDOR 2 CERRADO
	<	• ESTACIÓN 2 MEDIDOR 2 FALLO Y EP MEDIDOR 2 ABRIENDO
	=	• ESTACIÓN 2 MEDIDOR 2 ABIERTO Y EP MEDIDOR 2 ABRIENDO
	>	• ESTACIÓN 2 MEDIDOR 2 CERRADO Y EP MEDIDOR 2 ABRIENDO
?	• ESTACIÓN 2 MEDIDOR 2 Y EP MEDIDOR 2 ABRIENDO	
GRAPHDIS1(9)	0	• FALLA (AMBAS VÁLVULAS)
	1	• ESTACIÓN 2 MEDIDOR 3 ABIERTO Y EP MEDIDOR 3 FALLO
	2	• ESTACIÓN 2 MEDIDOR 3 CERRADO Y EP MEDIDOR 3 FALLO
	3	• ESTACIÓN 2 MEDIDOR 3 ABRIENDO Y EP MEDIDOR 3 FALLO
	4	• ESTACIÓN 2 MEDIDOR 3 FALLO Y EP MEDIDOR 3 ABIERTO
	5	• ESTACIÓN 2 MEDIDOR 3 Y EP MEDIDOR 3 ABIERTO
	6	• ESTACIÓN 2 MEDIDOR 3 CERRADO Y EP MEDIDOR 3 ABIERTO
	7	• ESTACIÓN 2 MEDIDOR 3 ABRIENDO Y EP MEDIDOR 3 ABIERTO
	8	• ESTACIÓN 2 MEDIDOR 3 FALLO Y EP MEDIDOR 3 CERRADO
	9	• ESTACIÓN 2 MEDIDOR 3 ABIERTO Y EP MEDIDOR 3 CERRADO
	:	• CERRADO (AMBAS VÁLVULAS)
	;	• ESTACIÓN 2 MEDIDOR 3 ABRIENDO Y EP MEDIDOR 3 CERRADO
	<	• ESTACIÓN 2 MEDIDOR 3 FALLO Y EP MEDIDOR 3 ABRIENDO
	=	• ESTACIÓN 2 MEDIDOR 3 ABIERTO Y EP MEDIDOR 3 ABRIENDO
>		

CAPITULO 3. DESARROLLO DEL SOFTWARE

	?	<ul style="list-style-type: none"> <li>• ESTACIÓN 2 MEDIDOR 3 CERRADO Y EP MEDIDOR 3 ABRIENDO</li> <li>• ESTACIÓN 2 MEDIDOR 3 Y EP MEDIDOR 3 ABRIENDO</li> </ul>
GRAPHDIS1(10)	0	• FALLA (AMBAS VÁLVULAS)
	1	• ESTACIÓN 2 MEDIDOR 4 ABIERTO Y EP MEDIDOR 4 FALLO
	2	• ESTACIÓN 2 MEDIDOR 4 CERRADO Y EP MEDIDOR 4 FALLO
	3	• ESTACIÓN 2 MEDIDOR 4 ABRIENDO Y EP MEDIDOR 4 FALLO
	4	• ESTACIÓN 2 MEDIDOR 4 FALLO Y EP MEDIDOR 4 ABIERTO
	5	• ESTACIÓN 2 MEDIDOR 4 Y EP MEDIDOR 4 ABIERTO
	6	• ESTACIÓN 2 MEDIDOR 4 CERRADO Y EP MEDIDOR 4 ABIERTO
	7	• ESTACIÓN 2 MEDIDOR 4 ABRIENDO Y EP MEDIDOR 4 ABIERTO
	8	• ESTACIÓN 2 MEDIDOR 4 FALLO Y EP MEDIDOR 4 CERRADO
	9	• ESTACIÓN 2 MEDIDOR 4 ABIERTO Y EP MEDIDOR 4 CERRADO
	:	• CERRADO (AMBAS VÁLVULAS)
	;	• ESTACIÓN 2 MEDIDOR 4 ABRIENDO Y EP MEDIDOR 4 CERRADO
	<	• ESTACIÓN 2 MEDIDOR 4 FALLO Y EP MEDIDOR 4 ABRIENDO
	=	• ESTACIÓN 2 MEDIDOR 4 ABIERTO Y EP MEDIDOR 4 ABRIENDO
>	• ESTACIÓN 2 MEDIDOR 4 CERRADO Y EP MEDIDOR 4 ABRIENDO	
?	• ESTACIÓN 2 MEDIDOR 4 Y EP MEDIDOR 4 ABRIENDO	
GRAPHDIS1(11)	0	• FALLO (AMBAS VÁLVULAS)
	1	• ESTACIÓN 1 SCE ABIERTO Y SCS FALLO
	2	• ESTACIÓN 1 SCE CERRADO Y SCS FALLO
	3	• ESTACIÓN 1 SCE ABRIENDO Y SCS FALLO
	4	• ESTACIÓN 1 SCE FALLO Y SCS ABIERTO
	5	• ESTACIÓN 1 SCE Y SCS ABIERTO
	6	• ESTACIÓN 1 SCE CERRADO Y SCS ABIERTO
	7	• ESTACIÓN 1 SCE ABRIENDO Y SCS ABIERTO
	8	• ESTACIÓN 1 SCE FALLO Y SCS CERRADO
	9	• ESTACIÓN 1 SCE ABIERTO Y SCS CERRADO
	:	• CERRADO (AMBAS VÁLVULAS)
	;	• ESTACIÓN 1 SCE ABRIENDO Y SCS CERRADO
	<	• ESTACIÓN 1 SCE FALLO Y SCS ABRIENDO
	=	• ESTACIÓN 1 SCE ABIERTO Y SCS ABRIENDO
>	• ESTACIÓN 1 SCE CERRADO Y SCS ABRIENDO	
?	• ESTACIÓN 1 SCE Y SCS ABRIENDO	
	0	• FALLO (AMBAS VÁLVULAS)
	1	• ESTACIÓN 2 SCE ABIERTO Y EP FALLO
	2	• ESTACIÓN 2 SCE CERRADO Y EP FALLO
	3	• ESTACIÓN 2 SCE ABRIENDO Y EP FALLO
	4	• ESTACIÓN 2 SCE FALLO Y EP ABIERTO

GRAPHDIS1(12)	5	• ESTACIÓN 2 SCE Y EP ABIERTO
	6	• ESTACIÓN 2 SCE CERRADO Y EP ABIERTO
	7	• ESTACIÓN 2 SCE ABRIENDO Y EP ABIERTO
	8	• ESTACIÓN 2 SCE FALLO Y EP CERRADO
	9	• ESTACIÓN 2 SCE ABIERTO Y EP CERRADO
	:	• CERRADO (AMBAS VÁLVULAS)
	<	• ESTACIÓN 2 SCE ABRIENDO Y EP CERRADO
	=	• ESTACIÓN 2 SCE FALLO Y EP ABRIENDO
	>	• ESTACIÓN 2 SCE CERRADO Y EP ABRIENDO
	?	• ESTACIÓN 2 SCE Y EP ABRIENDO
GRAPHDIS1(13)	0	• FALLO (AMBAS VÁLVULAS)
	1	• ESTACIÓN 2 SCS ABIERTO Y ESTACIÓN 1 RP FALLO
	2	• ESTACIÓN 2 SCS CERRADO Y ESTACIÓN 1 RP FALLO
	3	• ESTACIÓN 2 SCS ABRIENDO Y ESTACIÓN 1 RP FALLO
	4	• ESTACIÓN 2 SCS FALLO Y ESTACIÓN 1 RP ABIERTO
	5	• ESTACIÓN 2 SCS Y ESTACIÓN 1 RP ABIERTO
	6	• ESTACIÓN 2 SCS CERRADO Y ESTACIÓN 1 RP ABIERTO
	7	• ESTACIÓN 2 SCS ABRIENDO Y ESTACIÓN 1 RP ABIERTO
	8	• ESTACIÓN 2 SCS FALLO Y ESTACIÓN 1 RP CERRADO
	9	• ESTACIÓN 2 SCS ABIERTO Y ESTACIÓN 1 RP CERRADO
	:	• CERRADO (AMBAS VÁLVULAS)
	<	• ESTACIÓN 2 SCS ABRIENDO Y ESTACIÓN 1 RP CERRADO
	=	• ESTACIÓN 2 SCS FALLO Y ESTACIÓN 1 ESTACIÓN 1 RP ABRIENDO
	>	• ESTACIÓN 2 SCS ABIERTO Y ESTACIÓN 1 RP ABRIENDO
?	• ESTACIÓN 2 SCS CERRADO Y ESTACIÓN 1 RP ABRIENDO	
GRAPHDIS1(14)	0	• ESTACIÓN 2 RP FALLO
	1	• ESTACIÓN 2 RP ABIERTO
	2	• ESTACIÓN 2 RP CERRADO
	3	• ESTACIÓN 2 RP ABRIENDO
	4	• ESTACIÓN 2 RP FALLO
	5	• ESTACIÓN 2 RP ABIERTO
	6	• ESTACIÓN 2 RP CERRADO
	7	• ESTACIÓN 2 RP ABRIENDO
	8	• ESTACIÓN 2 RP FALLO
	9	• ESTACIÓN 2 RP ABIERTO
	:	• ESTACIÓN 2 RP CERRADO
<	• ESTACIÓN 2 RP ABRIENDO	
=	• ESTACIÓN 2 RP FALLO	
>	• ESTACIÓN 2 RP ABIERTO	
	• ESTACIÓN 2 RP CERRADO	

CAPITULO 3. DESARROLLO DEL SOFTWARE

	?	<ul style="list-style-type: none"> <li>• ESTACIÓN 2 RP ABRIENDO</li> </ul>
GRAPHDIS1(15)	1	<ul style="list-style-type: none"> <li>• ESTACIÓN 1 MEDIDOR 1</li> </ul>
	2	<ul style="list-style-type: none"> <li>• ESTACIÓN 1 MEDIDOR 2</li> </ul>
	3	<ul style="list-style-type: none"> <li>• ESTACIÓN 1 MEDIDOR 1, 2</li> </ul>
	4	<ul style="list-style-type: none"> <li>• ESTACIÓN 1 MEDIDOR 3</li> </ul>
	5	<ul style="list-style-type: none"> <li>• ESTACIÓN 1 MEDIDOR 1, 3</li> </ul>
	6	<ul style="list-style-type: none"> <li>• ESTACIÓN 1 MEDIDOR 2, 3</li> </ul>
	7	<ul style="list-style-type: none"> <li>• ESTACIÓN 1 MEDIDOR 1, 2, 3</li> </ul>
	8	<ul style="list-style-type: none"> <li>• ESTACIÓN 1 MEDIDOR 4</li> </ul>
	9	<ul style="list-style-type: none"> <li>• ESTACIÓN 1 MEDIDOR 1, 4</li> </ul>
	:	<ul style="list-style-type: none"> <li>• ESTACIÓN 1 MEDIDOR 2, 4</li> </ul>
	;	<ul style="list-style-type: none"> <li>• ESTACIÓN 1 MEDIDOR 1, 2, 4</li> </ul>
	<	<ul style="list-style-type: none"> <li>• ESTACIÓN 1 MEDIDOR 3, 4</li> </ul>
	=	<ul style="list-style-type: none"> <li>• ESTACIÓN 1 MEDIDOR 1, 3, 4</li> </ul>
	>	<ul style="list-style-type: none"> <li>• ESTACIÓN 1 MEDIDOR 2, 3, 4</li> </ul>
?	<ul style="list-style-type: none"> <li>• ESTACIÓN 1 MEDIDOR 1, 2, 3, 4</li> </ul>	
GRAPHDIS1(16)	1	<ul style="list-style-type: none"> <li>• ESTACIÓN 2 MEDIDOR 1</li> </ul>
	2	<ul style="list-style-type: none"> <li>• ESTACIÓN 2 MEDIDOR 2</li> </ul>
	3	<ul style="list-style-type: none"> <li>• ESTACIÓN 2 MEDIDOR 1, 2</li> </ul>
	4	<ul style="list-style-type: none"> <li>• ESTACIÓN 2 MEDIDOR 3</li> </ul>
	5	<ul style="list-style-type: none"> <li>• ESTACIÓN 2 MEDIDOR 1, 3</li> </ul>
	6	<ul style="list-style-type: none"> <li>• ESTACIÓN 2 MEDIDOR 2, 3</li> </ul>
	7	<ul style="list-style-type: none"> <li>• ESTACIÓN 2 MEDIDOR 1, 2, 3</li> </ul>
	8	<ul style="list-style-type: none"> <li>• ESTACIÓN 2 MEDIDOR 4</li> </ul>
	9	<ul style="list-style-type: none"> <li>• ESTACIÓN 2 MEDIDOR 1, 4</li> </ul>
	:	<ul style="list-style-type: none"> <li>• ESTACIÓN 2 MEDIDOR 2, 4</li> </ul>
	;	<ul style="list-style-type: none"> <li>• ESTACIÓN 2 MEDIDOR 1, 2, 4</li> </ul>
	<	<ul style="list-style-type: none"> <li>• ESTACIÓN 2 MEDIDOR 3, 4</li> </ul>
	=	<ul style="list-style-type: none"> <li>• ESTACIÓN 2 MEDIDOR 1, 3, 4</li> </ul>
	>	<ul style="list-style-type: none"> <li>• ESTACIÓN 2 MEDIDOR 2, 3, 4</li> </ul>
?	<ul style="list-style-type: none"> <li>• ESTACIÓN 2 MEDIDOR 1, 2, 3, 4</li> </ul>	
GRAPHDIS1(17)	0	<ul style="list-style-type: none"> <li>• COMPUTADORA, FILTRO, DP SELLO</li> </ul>
	1	<ul style="list-style-type: none"> <li>• COMPUTADORA</li> </ul>
	2	<ul style="list-style-type: none"> <li>• COMPUTADORA, DP SELLO</li> </ul>
	3	<ul style="list-style-type: none"> <li>• COMPUTADORA, FILTRO</li> </ul>
	4	<ul style="list-style-type: none"> <li>• MANUAL, FILTRO, DP SELLO</li> </ul>
	5	<ul style="list-style-type: none"> <li>• MANUAL</li> </ul>
	6	<ul style="list-style-type: none"> <li>• MANUAL, DP SELLO</li> </ul>
	7	<ul style="list-style-type: none"> <li>• MANUAL, FILTRO</li> </ul>
	8	<ul style="list-style-type: none"> <li>• CONTROLADOR, FILTRO, DP SELLO</li> </ul>
	9	<ul style="list-style-type: none"> <li>• CONTROLADOR</li> </ul>
:	<ul style="list-style-type: none"> <li>• CONTROLADOR DP SELLO</li> </ul>	
;	<ul style="list-style-type: none"> <li>• CONTROLADOR, FILTRO</li> </ul>	
GRAPHDIS1(18)	0	<ul style="list-style-type: none"> <li>• MODO COMPUTADORA</li> </ul>
	1	<ul style="list-style-type: none"> <li>• MODO COMPUTADORA</li> </ul>
	2	<ul style="list-style-type: none"> <li>• MODO COMPUTADORA</li> </ul>
	3	<ul style="list-style-type: none"> <li>• MODO COMPUTADORA</li> </ul>
	4	<ul style="list-style-type: none"> <li>• MODO MANUAL O CONTROLADOR</li> </ul>
	5	<ul style="list-style-type: none"> <li>• MODO MANUAL O CONTROLADOR</li> </ul>
	6	<ul style="list-style-type: none"> <li>• MODO MANUAL O CONTROLADOR</li> </ul>
	7	<ul style="list-style-type: none"> <li>• MODO MANUAL O CONTROLADOR</li> </ul>
8	<ul style="list-style-type: none"> <li>• MODO MANUAL O CONTROLADOR</li> </ul>	

	9	• MODO MANUAL O CONTROLADOR
	:	• MODO MANUAL O CONTROLADOR
	:	• MODO MANUAL O CONTROLADOR

### Proof Report

En esta solicitud el controlador Daniel MM3000 transmite la cadena de ASCII's a la interfaz de operación. El driver de comunicación no procesa la Información sino que únicamente le da el formato CSV para que la interfaz de operación la muestre al operador.

El significado de los caracteres que son enviados del campo a la sala de control dependen de la posición en que se encuentran dentro de la cadena de información recibida. Su significado se explica a continuación:

PROF1	SIGNIFICADO
1	01 H ASCII ACCESO AL DANIEL MM3000
2	32 H ASCII SOLICITUD PROOF REPORT
3 AL 49	INFORMACIÓN DEL REPORTE DE PRUEBA
50 AL 99	INFORMACIÓN DE LA TERMINAL MARÍTIMA
100 AL 146	20 H ASCII
147 AL 195	INFORMACIÓN DEL MEDIDOR EN PRUEBA
196 AL 243	20 H ASCII
244 AL 291	TÍTULOS DEL NÚMERO DE INTENTO, PULSOS, BARRILES POR HORA, TEMPERATURA, PRESIÓN Y FACTOR DE PRUEBA
292 AL 339	INFORMACIÓN DEL INTENTO, PULSO, BARRILES POR HORA, TEMPERATURA, PRESIÓN Y FACTOR DE PRUEBA 1
340 AL 387	INFORMACIÓN DEL INTENTO, PULSO, BARRILES POR HORA, TEMPERATURA, PRESIÓN Y FACTOR DE PRUEBA 2
388 AL 435	INFORMACIÓN DEL INTENTO, PULSO, BARRILES POR HORA, TEMPERATURA, PRESIÓN Y FACTOR DE PRUEBA 3
436 AL 495	INFORMACIÓN DEL INTENTO, PULSO, BARRILES POR HORA, TEMPERATURA, PRESIÓN Y FACTOR DE PRUEBA 4
496 AL 531	20 H ASCII
532 AL 579	TITULO DEL CALCULO DEL FACTOR DE MEDICIÓN
580 AL 627	20 H ASCII
628 AL 675	INFORMACIÓN DE LA GRAVEDAD ESPECIFICA
676 AL 723	20 H ASCII
724 AL 771	INFORMACIÓN DEL VOLUMEN CERTIFICADO DEL PROBADOR
772 AL 819	INFORMACIÓN DE LA TEMPERATURA PROMEDIO EN EL PROBADOR



820 AL 867	INFORMACIÓN DE LA PRESIÓN PROMEDIO EN EL PROBADOR
868 AL 915	INFORMACIÓN DE LA CORRECCIÓN DE LA TEMPERATURA DEL ACERO
915 AL 963	INFORMACIÓN DE LA CORRECCIÓN DE LA PRESIÓN DEL ACERO
964 AL 1011	INFORMACIÓN DE LA CORRECCIÓN DE LA TEMPERATURA DEL LIQUIDO
1012 AL 1059	INFORMACIÓN DE LA CORRECCIÓN DE LA PRESIÓN DEL LIQUIDO
1060 AL 1107	INFORMACIÓN DEL VOLUMEN DEL PROBADOR CORREGIDO
1108 AL 1155	20 H ASCII
1156 AL 1203	INFORMACIÓN DEL CONTEO DE PULSACIONES DE ENSAYO PROMEDIO
1204 AL 1251	INFORMACIÓN DE LA TEMPERATURA PROMEDIO DEL MEDIDOR
1252 AL 1299	INFORMACIÓN DE LA PRESIÓN PROMEDIO DEL MEDIDOR
1300 AL 1347	INFORMACIÓN DE LA CORRECCIÓN DE LA TEMPERATURA DEL LIQUIDO
1348 AL 1394	INFORMACIÓN DE LA CORRECCIÓN DE LA PRESIÓN DEL LIQUIDO
1395 AL 1443	INFORMACIÓN DEL CONTEO DE PULSACIONES PROMEDIO CORREGIDO
1444 AL 1491	20 H ASCII
1492 AL 1539	INFORMACIÓN DEL FACTOR DEL MEDIDOR
1540 AL 1587	INFORMACIÓN DEL FACTOR DEL MEDIDOR PREVIO
1588 AL 1635	INFORMACIÓN DEL PORCENTAJE DE DESVIACIÓN DEL FACTOR PREVIO
1636 AL 1682	20 H ASCII
1683	00 H ASCII

### Prover Display

En esta solicitud como en la anterior, el driver de comunicación sólo se encarga de dar el formato CSV a la cadena de caracteres ASCII's. La interfaz de operación y el driver de comunicación no realizan ningún análisis ni interpretación de la información que se muestra al operador.

Los caracteres recibidos tienen un significado diferente de acuerdo a la posición en que son recibidos, a continuación se explica este.

PROVER1	SIGNIFICADO
1	01 H ASCII ACCESO AL DANIEL MM3000
2	33 H ASCII SOLICITUD PROVER DISPLAY

3 AL 11	INFORMACIÓN DEL MEDIDOR EN PRUEBA
12 AL 16	INFORMACIÓN DEL VALOR DE FLUJO DEL MEDIDOR
17 AL 30	INFORMACIÓN DEL ESTADO DE LA PRUEBA
31 AL 38	INFORMACIÓN DEL ESTADO DEL LANZADOR
39 AL 44	INFORMACIÓN DE LA TEMPERATURA EN EL MEDIDOR
45 AL 50	PRESIÓN DEL CABEZAL DE ENTRADA AL MEDIDOR
51 AL 56	INFORMACIÓN DE LA TEMPERATURA EN EL PROBADOR
57 AL 62	PRESIÓN DEL CABEZAL DE ENTRADA AL PROBADOR
63 AL 68	GRAVEDAD ESPECIFICA
69 AL 70	INFORMACIÓN DEL INTENTO 1
71 AL 76	INFORMACIÓN DE PULSOS 1
77 AL 82	INFORMACIÓN DE BARRILES POR HORA 1
83 AL 88	INFORMACIÓN DE TEMPERATURA EN DEGC 1
89 AL 94	INFORMACIÓN DE LA PRESIÓN EN KG/CM <sup>2</sup> 1
95 AL 102	INFORMACIÓN DEL FACTOR DE PRUEBA 1
103 AL 104	INFORMACIÓN DEL INTENTO 2
105 AL 110	INFORMACIÓN DE PULSOS 2
111 AL 116	INFORMACIÓN DE BARRILES POR HORA 2
117 AL 122	INFORMACIÓN DE TEMPERATURA EN DEGC 2
123 AL 128	INFORMACIÓN DE LA PRESIÓN EN KG/CM <sup>2</sup> 2
129 AL 136	INFORMACIÓN DEL FACTOR DE PRUEBA 2
137 AL 138	INFORMACIÓN DEL INTENTO 3
139 AL 144	INFORMACIÓN DE PULSOS 3
145 AL 150	INFORMACIÓN DE BARRILES POR HORA 3
151 AL 156	INFORMACIÓN DE TEMPERATURA EN DEGC 3
157 AL 162	INFORMACIÓN DE LA PRESIÓN EN KG/CM <sup>2</sup> 3
163 AL 170	INFORMACIÓN DEL FACTOR DE PRUEBA 3
171 AL 172	INFORMACIÓN DEL INTENTO 4
173 AL 178	INFORMACIÓN DE PULSOS 4
179 AL 184	INFORMACIÓN DE BARRILES POR HORA 4
185 AL 190	INFORMACIÓN DE TEMPERATURA EN DEGC 4
191 AL 196	INFORMACIÓN DE LA PRESIÓN EN KG/CM <sup>2</sup> 4
197 AL 204	INFORMACIÓN DEL FACTOR DE PRUEBA 4
205	OD H ASCII

### Batch Display

El controlador Daniel MM3000 envía la cadena de caracteres ASCII's al driver de comunicación, éste sólo se encarga de dar el formato CSV a la cadena. La interfaz de operación y el driver de comunicación no realizan ningún análisis ni interpretación de la información que se muestra al operador.



A continuación se explica el significado de cada carácter dependiendo de la posición en que se recibe:

BATCH1	SIGNIFICADO
1	01 H ASCII ACCESO AL DANIEL MM3000
2	35 H ASCII SOLICITUD BATCH DISPLAY
3	31 Ó 32 H ASCII BANDA DE MEDICIÓN 1 Ó 2
4	20 H ASCII
5 AL 42	NOMBRE DEL BUQUE-TANQUE
43 AL 50	NÚMERO DEL EMBARQUE
51 AL 62	INFORMACIÓN DEL ESTADO DEL LOTE
63 AL 68	INFORMACIÓN DE LA RAZÓN DE FLUJO DE ARRANQUE
69 AL 76	INFORMACIÓN DEL VOLUMEN DEL LOTE EN BARRILES BRUTOS
77 AL 82	INFORMACIÓN DE LA RAZÓN DE FLUJO DE CARGA
83 AL 90	INFORMACIÓN DEL VOLUMEN DE TERMINADO EN BARRILES BRUTOS
91 AL 96	INFORMACIÓN DE LA RAZÓN DE FLUJO DE TERMINADO
97 AL 104	INFORMACIÓN DEL MEDIDOR 1
105 AL 112	INFORMACIÓN DEL FACTOR DE PULSACIONES/BBLS 1
103 AL 118	INFORMACIÓN DE TEMPERATURA EN DEGC 1
119 AL 124	INFORMACIÓN DEL PUNTO DE AJUSTE FIJADO AL RANGO DE FLUJO 1
125 AL 130	INFORMACIÓN DEL RANGO DE BARRILES POR HORA 1
131 AL 138	INFORMACIÓN DE BARRILES BRUTOS 1
139 AL 146	INFORMACIÓN DE BARRILES NETOS 1
147 AL 154	INFORMACIÓN DEL MEDIDOR 2
155 AL 162	INFORMACIÓN DEL FACTOR DE PULSACIONES/BBLS 2
163 AL 168	INFORMACIÓN DE TEMPERATURA EN DEGC 2
169 AL 174	INFORMACIÓN DEL PUNTO DE AJUSTE FIJADO AL RANGO DE FLUJO 2
175 AL 180	INFORMACIÓN DEL RANGO DE BARRILES POR HORA 2
181 AL 188	INFORMACIÓN DE BARRILES BRUTOS 2
189 AL 196	INFORMACIÓN DE BARRILES NETOS 2
197 AL 204	INFORMACIÓN DEL MEDIDOR 3
205 AL 212	INFORMACIÓN DEL FACTOR DE PULSACIONES/BBLS 3
213 AL 218	INFORMACIÓN DE TEMPERATURA EN DEGC 3
219 AL 224	INFORMACIÓN DEL PUNTO DE AJUSTE FIJADO AL RANGO DE FLUJO 3
225 AL 230	INFORMACIÓN DEL RANGO DE BARRILES POR HORA 3
231 AL 238	INFORMACIÓN DE BARRILES BRUTOS 3

239 AL 246	INFORMACIÓN DE BARRILES NETOS 3
247 AL 254	INFORMACIÓN DEL MEDIDOR 4
255 AL 262	INFORMACIÓN DEL FACTOR DE PULSACIONES/BBLS 4
263 AL 268	INFORMACIÓN DE TEMPERATURA EN DEGC 4
269 AL 274	INFORMACIÓN DEL PUNTO DE AJUSTE FIJADO AL RANGO DE FLUJO 4
275 AL 280	INFORMACIÓN DEL RANGO DE BARRILES POR HORA 4
281 AL 288	INFORMACIÓN DE BARRILES BRUTOS 4
289 AL 296	INFORMACIÓN DE BARRILES NETOS 4
297 AL 304	INFORMACIÓN DEL MEDIDOR 5
305 AL 312	INFORMACIÓN DEL FACTOR DE PULSACIONES/BBLS 5
313 AL 318	INFORMACIÓN DE TEMPERATURA EN DEGC 5
319 AL 324	INFORMACIÓN DEL PUNTO DE AJUSTE FIJADO AL RANGO DE FLUJO 5
325 AL 330	INFORMACIÓN DEL RANGO DE BARRILES POR HORA 5
331 AL 338	INFORMACIÓN DE BARRILES BRUTOS 5
339 AL 346	INFORMACIÓN DE BARRILES NETOS 5
347 AL 352	INFORMACIÓN DEL TOTAL DEL PUNTO DE AJUSTE FIJADO AL RANGO DE FLUJO
353 AL 358	INFORMACIÓN DEL TOTAL DEL RANGO DE BARRILES POR HORA
359 AL 366	INFORMACIÓN DEL TOTAL DE BARRILES BRUTOS
367 AL 374	INFORMACIÓN DEL TOTAL DE BARRILES NETOS
375	OD H ASCII

### Alarm Report

El controlador Daniel MM3000 envía la cadena de caracteres ASCII's al driver de comunicación, éste se encarga de procesarla generando un archivo binario en el cual los unos representan los estados de alarma y los ceros representan una condición normal de operación, esta cadena también tiene formato CSV. La interfaz de operación interpreta la información y la muestra al operador.

El análisis que el driver de comunicación hace a la cadena consiste en comparar cada uno de los caracteres recibidos y asignar ceros y unos colocándolos en la posición que le corresponda, generando una cadena equivalente a la recibida.

Los caracteres que puede recibir el driver y su equivalencia para la colocación de unos en el archivo que presenta a Intouch es la siguiente:

0 = 0	9 = 9
1 = 1	: = 10
2 = 2	; = 11
3 = 3	< = 12
4 = 4	= = 13
5 = 6	> = 14
7 = 7	? = 15
8 = 8	

Supongamos que en la cadena que envia el Daniel MM3000 el carácter (3) es igual a un "7", el carácter (4) igual a un "0" y el carácter (5) es igual a ":", ya que los caracteres 3 y 4 solo pueden de tomar valores de 0 a 9 el driver de comunicación coloca ceros en las primeras seis posiciones de la cadena y un uno en la posición número siete la cual corresponde al carácter "7" enviado en el carácter (3), las posiciones ocho y nueve son llenadas con ceros. Para el carácter (4) el cual contiene un cero, las nueve posiciones que le corresponden son llenadas con ceros. Por su parte el carácter (5) tiene un valor ":" (10), por lo que se colocan ceros de la posición 1 a 9 y en la posición 10 se coloca un "1".

La cadena que entrega el driver de comunicación para estos tres caracteres es la siguiente:

0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,

A continuación se listan los posibles valores de cada carácter y su significado dependiendo de la posición en que se recibe:

HFR RANGO DE FLUJO ALTO  
 LFR RANGO DE FLUJO BAJO  
 PCF FALLA DE COMPARACIÓN DE PULSOS  
 FO FALLO AL ABRIR  
 FC FALLO AL CERRAR  
 CS CABEZAL DE SALIDA  
 EP ENTRADA AL PROBADOR  
 CE CABEZAL DE ENTRADA  
 RP REGRESO AL PROBADOR

ACTUA	RESPUESTA POSIBLE EN ASCII	SIGNIFICADO
ACTUA(1)	01H	• ACCESO AL CONTROLADOR DANIEL MM3000
ACTUA(2)	31 H	• SOLICITUD REGULAR SCAN
ACTUA(3) ESTADO DEL LOTE ESTACIÓN 1	0	• IDLE
	1	• SETUP
	2	• STARTING
	3	• AT START UP
	4	• RAMPING UP
	5	• AT LOAD RATE
	6	• TOPPING OFF
	7	• SUSPENDING
	8	• SUSPENDE
	9	• TERMINATING
ACTUA(4) ESTADO DEL LOTE ESTACIÓN 2	0	• IDLE
	1	• SETUP
	2	• STARTING
	3	• AT START UP
	4	• RAMPING UP
	5	• AT LOAD RATE
	6	• TOPPING OFF
	7	• SUSPENDING
	8	• SUSPENDE
	9	• TERMINATING
ACTUA(5) ESTADO DEL PROBADOR	0	• IDLE
	1	• GOING ON PROOF
	2	• TEMP DELAY
	3	• LAUNCHING
	4	• LAUNCHING
	5	• LAUNCHED
	6	• COUNTING
	7	• BALL RESEATING
	8	• GOING OFF PROOF
	9	• GOING OFF PROOF
:	• COMPLETED	
ACTUA(6) ESTADO DEL MEDIDOR SPARE	0	• FUERA DE SERVICIO (DESELECCIONADO)
	1	• ESTACIÓN 1 SELECCIONADA
	2	• ESTACIÓN 2 SELECCIONADA
ACTUA(7) MODO DE OPERACIÓN	1	• MODO MANUAL
	2	• MODO CONTROLADOR
	4	• MODO COMPUTADORA
ACTUA(8) 2230 FALLA DE COMUNICACIÓN	1	• ESTACIÓN 1 MEDIDOR 1
	2	• ESTACIÓN 1 MEDIDOR 2
	3	• ESTACIÓN 1 MEDIDORES 1, 2
	4	• ESTACIÓN 1 MEDIDOR 3
	5	• ESTACIÓN 1 MEDIDORES 1, 3
	6	• ESTACIÓN 1 MEDIDORES 2, 3
	7	• ESTACIÓN 1 MEDIDORES 1, 2, 3
	8	• ESTACIÓN 1 MEDIDOR 4
	9	• ESTACIÓN 1 MEDIDORES 1, 4

CAPITULO 3. DESARROLLO DEL SOFTWARE

	: ; < = > ?	<ul style="list-style-type: none"> <li>• ESTACIÓN 1 MEDIDORES 2,4</li> <li>• ESTACIÓN 1 MEDIDORES 1, 2, 4</li> <li>• ESTACIÓN 1 MEDIDORES 3, 4</li> <li>• ESTACIÓN 1 MEDIDORES 1, 3, 4</li> <li>• ESTACIÓN 1 MEDIDORES 2, 3, 4</li> <li>• ESTACIÓN 1 MEDIDORES 1, 2, 3, 4</li> </ul>
ACTUA(9) 2230 FALLA DE COMUNICACIÓN	1 2 3 4 5 6 7 8 9 : ; < = > ?	<ul style="list-style-type: none"> <li>• ESTACIÓN 2 MEDIDOR 1</li> <li>• ESTACIÓN 2 MEDIDOR 2</li> <li>• ESTACIÓN 2 MEDIDORES 1, 2</li> <li>• ESTACIÓN 2 MEDIDOR 3</li> <li>• ESTACIÓN 2 MEDIDORES 1, 3</li> <li>• ESTACIÓN 2 MEDIDORES 2, 3</li> <li>• ESTACIÓN 2 MEDIDORES 1, 2, 3</li> <li>• ESTACIÓN 2 MEDIDOR 4</li> <li>• ESTACIÓN 2 MEDIDORES 1, 4</li> <li>• ESTACIÓN 2 MEDIDORES 2, 4</li> <li>• ESTACIÓN 2 MEDIDORES 1, 2, 4</li> <li>• ESTACIÓN 2 MEDIDORES 3, 4</li> <li>• ESTACIÓN 2 MEDIDORES 1, 3, 4</li> <li>• ESTACIÓN 2 MEDIDORES 2, 3, 4</li> <li>• ESTACIÓN 2 MEDIDORES 1, 2, 3, 4</li> </ul>
ACTUA(10)  • 2230 FALLA DE COMUNICACIÓN(A)  • SETPOINT INALCANZABLE(B)	1 2 3 4 5 6 7 8 9 : ; < = > ?	<ul style="list-style-type: none"> <li>• SPARE (A)</li> <li>• ESTACIÓN 1 MEDIDOR 1 (B)</li> <li>• SPARE (A) Y ESTACIÓN 1 MEDIDOR 1 (B)</li> <li>• ESTACIÓN 1 MEDIDOR 2 (B)</li> <li>• SPARE (A) Y ESTACIÓN 1 MEDIDOR 2 (B)</li> <li>• ESTACIÓN 1 MEDIDORES 1, 2 (B)</li> <li>• SPARE (A) Y ESTACIÓN 1 MEDIDORES 1, 2 (B)</li> <li>• ESTACIÓN 1 MEDIDOR 3 (B)</li> <li>• SPARE (A) Y ESTACIÓN 1 MEDIDOR 3 (B)</li> <li>• ESTACIÓN 1 MEDIDORES 1, 3 (B)</li> <li>• SPARE (A) Y ESTACIÓN 1 MEDIDORES 1, 3 (B)</li> <li>• ESTACIÓN 1 MEDIDORES 2,3 (B)</li> <li>• SPARE (A) Y ESTACIÓN 1 MEDIDORES 2, 3 (B)</li> <li>• ESTACIÓN 1 MEDIDORES 1, 2, 3 (B)</li> <li>• SPARE (A) Y ESTACIÓN 1 MEDIDORES 1, 2, 3 (B)</li> </ul>
ACTUA(11) SETPOINT INALCANZABLE	1 2 3 4 5 6 7 8 9 : ; < =	<ul style="list-style-type: none"> <li>• ESTACIÓN 1 MEDIDOR 4</li> <li>• ESTACIÓN 2 MEDIDOR 1</li> <li>• ESTACIÓN 1 MEDIDOR 4 Y ESTACIÓN 2 MEDIDOR 1</li> <li>• ESTACIÓN 2 MEDIDOR 2</li> <li>• ESTACIÓN 1 MEDIDOR 4 Y ESTACIÓN 2 MEDIDOR 2</li> <li>• ESTACIÓN 2 MEDIDORES 1,2</li> <li>• ESTACIÓN 1 MEDIDOR 4 Y ESTACIÓN 2 MEDIDORES 1, 2</li> <li>• ESTACIÓN 2 MEDIDOR 3</li> <li>• ESTACIÓN 1 MEDIDOR 4 Y ESTACIÓN 2 MEDIDOR 3</li> <li>• ESTACIÓN 2 MEDIDORES 1, 3</li> <li>• ESTACIÓN 1 MEDIDOR 4 Y ESTACIÓN 2 MEDIDORES 1, 3</li> <li>• ESTACIÓN 2 MEDIDORES 2, 3</li> <li>• ESTACIÓN 1 MEDIDOR 4 Y ESTACIÓN 2 MEDIDORES 2, 3</li> </ul>

	> ?	<ul style="list-style-type: none"> <li>ESTACIÓN 2 MEDIDORES 1, 2, 3</li> <li>ESTACIÓN 1 MEDIDOR 4 Y ESTACIÓN 2 MEDIDORES 1,2,3</li> </ul>
ACTUA(12)  <ul style="list-style-type: none"> <li>SETPOINT INALCANZABLE (A)</li> <li>ALTA PRESIÓN DEL STRAINER (B)</li> </ul>	1 2 3 4 5 6 7 8 9 : ; < = > ?	<ul style="list-style-type: none"> <li>ESTACIÓN 2 MEDIDOR 4 (A)</li> <li>SPARE (A)</li> <li>ESTACIÓN 2 MEDIDOR 4 (A) Y SPARE (A)</li> <li>ESTACIÓN 1 MEDIDOR 1 (B)</li> <li>ESTACIÓN 2 MEDIDOR 4 (A) Y ESTACIÓN 1 MEDIDOR 1 (B)</li> <li>SPARE (A) Y ESTACIÓN 1 MEDIDOR 1 (B)</li> <li>ESTACIÓN 2 MEDIDOR 4 (A), SPARE (A) Y ESTACIÓN 1 MEDIDOR 1 (B)</li> <li>ESTACIÓN 1 MEDIDOR 2 (B)</li> <li>ESTACIÓN 2 MEDIDOR 4 (A) Y ESTACIÓN 1 MEDIDOR 2 (B)</li> <li>SPARE (A) Y ESTACIÓN 1 MEDIDOR 2 (B)</li> <li>ESTACIÓN 2 MEDIDOR 4 (A), SPARE (A) Y ESTACIÓN 1 MEDIDOR 2 (B)</li> <li>ESTACIÓN 1 MEDIDORES 1, 2 (B)</li> <li>ESTACIÓN 2 MEDIDOR 4 (A) Y ESTACIÓN 1 MEDIDORES 1,2 (B)</li> <li>SPARE (A) Y ESTACIÓN 1 MEDIDORES 1, 2 (B)</li> <li>ESTACIÓN 2 MEDIDOR 4 (A), SPARE (A) Y ESTACIÓN 1 MEDIDORES 1, 2 (B)</li> </ul>
ACTUA(13) ALTA PRESIÓN DEL STRAINER	1 2 3 4 5 6 7 8 9 : ; < = > ?	<ul style="list-style-type: none"> <li>ESTACIÓN 1 MEDIDOR 3</li> <li>ESTACIÓN 1 MEDIDOR 4</li> <li>ESTACIÓN 1 MEDIDORES 3,4</li> <li>ESTACIÓN 2 MEDIDOR 1</li> <li>ESTACIÓN 1 MEDIDOR 3 Y ESTACIÓN 2 MEDIDOR 1</li> <li>ESTACIÓN 1 MEDIDOR 4 Y ESTACIÓN 2 MEDIDOR 1</li> <li>ESTACIÓN 1 MEDIDORES 3,4 Y ESTACIÓN 2 MEDIDOR 1</li> <li>ESTACIÓN 2 MEDIDOR 2</li> <li>ESTACIÓN 1 MEDIDOR 3 Y ESTACIÓN 2 MEDIDOR 2</li> <li>ESTACIÓN 1 MEDIDOR 4 Y ESTACIÓN 2 MEDIDOR 2</li> <li>ESTACIÓN 1 MEDIDOR 3,4 Y ESTACIÓN 2 MEDIDOR 2</li> <li>ESTACIÓN 2 MEDIDORES 1,2</li> <li>ESTACIÓN 1 MEDIDOR 3 Y ESTACIÓN 2 MEDIDORES 1,2</li> <li>ESTACIÓN 1 MEDIDOR 4 Y ESTACIÓN 2 MEDIDORES 1,2</li> <li>ESTACIÓN 1 MEDIDORES 3,4 Y ESTACIÓN 2 MEDIDORES 1,2</li> </ul>
ACTUA(14)  <ul style="list-style-type: none"> <li>ALTA PRESIÓN DEL STRAINER (A)</li> </ul>	1 2 3 4 5 6 7 8 9	<ul style="list-style-type: none"> <li>ESTACIÓN 2 MEDIDOR 3 (A)</li> <li>ESTACIÓN 2 MEDIDOR 4 (A)</li> <li>ESTACIÓN 2 MEDIDORES 3,4 (A)</li> <li>SPARE (A)</li> <li>ESTACIÓN 2 MEDIDOR 3 Y SPARE (A)</li> <li>ESTACIÓN 2 MEDIDOR 4 Y SPARE (A)</li> <li>ESTACIÓN 2 MEDIDORES 3,4 Y SPARE (A)</li> <li>ESTACIÓN 1 MEDIDOR 1 (B)</li> <li>ESTACIÓN 2 MEDIDOR 3 (A) Y ESTACIÓN 1 MEDIDOR 1 (B)</li> </ul>



• MEDIDOR (B) FUERA DE SERVICIO	: ; < = > ?	<ul style="list-style-type: none"> <li>• ESTACIÓN 2 MEDIDOR 4 (A) Y ESTACIÓN 1 MEDIDOR 1 (B)</li> <li>• ESTACIÓN 2 MEDIDORES 3,4 (A) Y ESTACIÓN 1 MEDIDOR 1 (B)</li> <li>• SPARE (A) Y ESTACIÓN 1 MEDIDOR 1 (B)</li> <li>• ESTACIÓN 2 MEDIDOR 3, SPARE (A) Y ESTACIÓN 1 MEDIDOR 1 (B)</li> <li>• ESTACIÓN 2 MEDIDOR 4, SPARE (A) Y ESTACIÓN 1 MEDIDOR 1 (B)</li> <li>• ESTACIÓN 2 MEDIDOR 3,4, SPARE (A) Y ESTACIÓN 1 MEDIDOR 1 (B)</li> </ul>
ACTUA(15) MEDIDOR FUERA DE SERVICIO	1 2 3 4 5 6 7 8 9 : ; < = > ?	<ul style="list-style-type: none"> <li>• ESTACIÓN 1 MEDIDOR 2</li> <li>• ESTACIÓN 1 MEDIDOR 3</li> <li>• ESTACIÓN 1 MEDIDORES 2,3</li> <li>• ESTACIÓN 1 MEDIDOR 4</li> <li>• ESTACIÓN 1 MEDIDORES 2,4</li> <li>• ESTACIÓN 1 MEDIDORES 3,4</li> <li>• ESTACIÓN 1 MEDIDORES 2,3,4</li> <li>• ESTACIÓN 2 MEDIDOR 1</li> <li>• ESTACIÓN 1 MEDIDOR 2 Y ESTACIÓN 2 MEDIDOR 1</li> <li>• ESTACIÓN 1 MEDIDOR 3 Y ESTACIÓN 2 MEDIDOR 1</li> <li>• ESTACIÓN 1 MEDIDORES 2,3 Y ESTACIÓN 2 MEDIDOR 1</li> <li>• ESTACIÓN 1 MEDIDOR 4 Y ESTACIÓN 2 MEDIDOR 1</li> <li>• ESTACIÓN 1 MEDIDORES 2,4 Y ESTACIÓN 2 MEDIDOR 1</li> <li>• ESTACIÓN 1 MEDIDOR 3,4 Y ESTACIÓN 2 MEDIDOR 1</li> <li>• ESTACIÓN 1 MEDIDORES 2,3,4 Y ESTACIÓN 2 MEDIDOR 1</li> </ul>
ACTUA(16) MEDIDOR FUERA DE SERVICIO	1 2 3 4 5 6 7 8 9 : ; < = > ?	<ul style="list-style-type: none"> <li>• ESTACIÓN 2 MEDIDOR 2</li> <li>• ESTACIÓN 2 MEDIDOR 3</li> <li>• ESTACIÓN 2 MEDIDORES 2,3</li> <li>• ESTACIÓN 2 MEDIDOR 4</li> <li>• ESTACIÓN 2 MEDIDORES 2,4</li> <li>• ESTACIÓN 2 MEDIDORES 3,4</li> <li>• ESTACIÓN 2 MEDIDORES 2,3,4</li> <li>• SPARE</li> <li>• ESTACIÓN 2 MEDIDOR 2 Y SPARE</li> <li>• ESTACIÓN 2 MEDIDOR 3 Y SPARE</li> <li>• ESTACIÓN 2 MEDIDORES 2,3 Y SPARE</li> <li>• ESTACIÓN 2 MEDIDOR 4 Y SPARE</li> <li>• ESTACIÓN 2 MEDIDORES 2,4 Y SPARE</li> <li>• ESTACIÓN 2 MEDIDORES 3,4 Y SPARE</li> <li>• ESTACIÓN 2 MEDIDORES 2,3,4 Y SPARE</li> </ul>
ACTUA(17) 2230 DESVIACIÓN BRUTA	1 2 3 4 5 6 7 8 9	<ul style="list-style-type: none"> <li>• ESTACIÓN 1 MEDIDOR 1</li> <li>• ESTACIÓN 1 MEDIDOR 2</li> <li>• ESTACIÓN 1 MEDIDORES 1, 2</li> <li>• ESTACIÓN 1 MEDIDOR 3</li> <li>• ESTACIÓN 1 MEDIDORES 1, 3</li> <li>• ESTACIÓN 1 MEDIDORES 2, 3</li> <li>• ESTACIÓN 1 MEDIDORES 1, 2, 3</li> <li>• ESTACIÓN 1 MEDIDOR 4</li> <li>• ESTACIÓN 1 MEDIDORES 1, 4</li> </ul>

	: ; < = > ?	<ul style="list-style-type: none"> <li>• ESTACIÓN 1 MEDIDORES 2,4</li> <li>• ESTACIÓN 1 MEDIDORES 1, 2, 4</li> <li>• ESTACIÓN 1 MEDIDORES 3, 4</li> <li>• ESTACIÓN 1 MEDIDORES 1, 3, 4</li> <li>• ESTACIÓN 1 MEDIDORES 2, 3, 4</li> <li>• ESTACIÓN 1 MEDIDORES 1, 2, 3, 4</li> </ul>
ACTUA(18) 2230 DESVIACIÓN BRUTA	1 2 3 4 5 6 7 8 9 : ; < = > ?	<ul style="list-style-type: none"> <li>• ESTACIÓN 2 MEDIDOR 1</li> <li>• ESTACIÓN 2 MEDIDOR 2</li> <li>• ESTACIÓN 2 MEDIDORES 1, 2</li> <li>• ESTACIÓN 2 MEDIDOR 3</li> <li>• ESTACIÓN 2 MEDIDORES 1, 3</li> <li>• ESTACIÓN 2 MEDIDORES 2, 3</li> <li>• ESTACIÓN 2 MEDIDORES 1, 2, 3</li> <li>• ESTACIÓN 2 MEDIDOR 4</li> <li>• ESTACIÓN 2 MEDIDORES 1, 4</li> <li>• ESTACIÓN 2 MEDIDORES 2,4</li> <li>• ESTACIÓN 2 MEDIDORES 1, 2, 4</li> <li>• ESTACIÓN 2 MEDIDORES 3, 4</li> <li>• ESTACIÓN 2 MEDIDORES 1, 3, 4</li> <li>• ESTACIÓN 2 MEDIDORES 2, 3, 4</li> <li>• ESTACIÓN 2 MEDIDORES 1, 2, 3, 4</li> </ul>
ACTUA(19)  • 2230 DESVIACIÓN BRUTA (A)  • MEDIDOR (B) ALARMAS	1 2 3 4 5 6 7 8 9 : ; < = > ?	<ul style="list-style-type: none"> <li>• SPARE (A)</li> <li>• ESTACIÓN 1 MEDIDOR 1 HFR (B)</li> <li>• SPARE (A) Y ESTACIÓN 1 MEDIDOR 1 HFR (B)</li> <li>• ESTACIÓN 1 MEDIDOR 1 LFR (B)</li> <li>• SPARE (A) Y ESTACIÓN 1 MEDIDOR 1 LFR (B)</li> <li>• ESTACIÓN 1 MEDIDOR 1 HFR, LFR (B)</li> <li>• SPARE (A) Y ESTACIÓN 1 MEDIDOR 1 HFR, LFR (B)</li> <li>• ESTACIÓN 1 MEDIDOR 1 PCF (B)</li> <li>• SPARE (A) Y ESTACIÓN 1 MEDIDOR 1 PCF (B)</li> <li>• ESTACIÓN 1 MEDIDOR 1 HFR, PCF (B)</li> <li>• SPARE (A) Y ESTACIÓN 1 MEDIDOR 1 HFR, PCF (B)</li> <li>• ESTACIÓN 1 MEDIDOR 1 LFR, PCF (B)</li> <li>• SPARE (A) Y ESTACIÓN 1 MEDIDOR 1 LFR, PCF (B)</li> <li>• ESTACIÓN 1 MEDIDOR 1 HFR, LFR, PCF (B)</li> <li>• SPARE (A) Y ESTACIÓN 1 MEDIDOR 1 HFR, LFR, PCF (B)</li> </ul>
ACTUA(20) • MEDIDOR ALARMAS	1 2 3 4 5 6 7 8 9 : ; < = > ?	<ul style="list-style-type: none"> <li>• ESTACIÓN 1 MEDIDOR 2 HFR</li> <li>• ESTACIÓN 1 MEDIDOR 2 LFR</li> <li>• ESTACIÓN 1 MEDIDOR 2 HFR, LFR</li> <li>• ESTACIÓN 1 MEDIDOR 2 PCF</li> <li>• ESTACIÓN 1 MEDIDOR 2 HFR, PCF</li> <li>• ESTACIÓN 1 MEDIDOR 2 LFR, PCF</li> <li>• ESTACIÓN 1 MEDIDOR 2 HFR, LFR, PCF</li> <li>• ESTACIÓN 1 MEDIDOR 3 HFR</li> <li>• ESTACIÓN 1 MEDIDORES 2,3 HFR</li> <li>• ESTACIÓN 1 MEDIDOR 2 LFR Y MEDIDOR 3 HFR</li> <li>• ESTACIÓN 1 MEDIDOR 2 HFR, LFR Y MEDIDOR 3 HFR</li> <li>• ESTACIÓN 1 MEDIDOR 2 PCF Y MEDIDOR 3 HFR</li> <li>• ESTACIÓN 1 MEDIDOR 2 HFR, LFR Y MEDIDOR 3 HFR</li> <li>• ESTACIÓN 1 MEDIDOR 2 LFR, PCF Y MEDIDOR 3 HFR</li> <li>• ESTACIÓN 1 MEDIDOR 2 HFR, LFR, PCF Y MEDIDOR 3</li> </ul>

		HFR
ACTUA(21) • MEDIDOR ALARMAS	1	• ESTACIÓN 1 MEDIDOR 3 LFR
	2	• ESTACIÓN 1 MEDIDOR 3 PCF
	3	• ESTACIÓN 1 MEDIDOR 3 LFR, PCF
	4	• ESTACIÓN 1 MEDIDOR 4 HFR
	5	• ESTACIÓN 1 MEDIDOR 3 LFR Y MEDIDOR 4 HFR
	6	• ESTACIÓN 1 MEDIDOR 3 PCF Y MEDIDOR 4 HFR
	7	• ESTACIÓN 1 MEDIDOR 3 LFR, PCF Y MEDIDOR 4 HFR
	8	• ESTACIÓN 1 MEDIDOR 4 LFR
	9	• ESTACIÓN 1 MEDIDOR 3 LFR Y MEDIDOR 4 LFR
	:	• ESTACIÓN 1 MEDIDOR 3 PCF Y MEDIDOR 4 LFR
	;	• ESTACIÓN 1 MEDIDOR 3 LFR, PCF Y MEDIDOR 4 LFR
	<	• ESTACIÓN 1 MEDIDOR 4 HFR, LFR
	=	• ESTACIÓN 1 MEDIDOR 3 LFR Y MEDIDOR 4 HFR, LFR
>	• ESTACIÓN 1 MEDIDOR 3 PCF Y MEDIDOR 4 HFR, LFR	
?	• ESTACIÓN 1 MEDIDOR 3 LFR, PCF Y MEDIDOR 4 HFR, LFR	
ACTUA(22) • MEDIDOR ALARMAS	1	• ESTACIÓN 1 MEDIDOR 4 PCF
	2	• ESTACIÓN 2 MEDIDOR 1 HFR
	3	• ESTACIÓN 1 MEDIDOR 4 PCF Y ESTACIÓN 2 MEDIDOR 1 HFR
	4	• ESTACIÓN 2 MEDIDOR 1 LFR
	5	• ESTACIÓN 1 MEDIDOR 4 PCF Y ESTACIÓN 2 MEDIDOR 1 LFR
	6	• ESTACIÓN 2 MEDIDOR 1 HFR, LFR
	7	• ESTACIÓN 1 MEDIDOR 4 PCF Y ESTACIÓN 2 MEDIDOR 1 HFR, LFR
	8	• ESTACIÓN 2 MEDIDOR 1 PCF
	9	• ESTACIÓN 1 MEDIDOR 4 PCF Y ESTACIÓN 2 MEDIDOR 1 PCF
	:	• ESTACIÓN 2 MEDIDOR 1 HFR, PCF
	;	• ESTACIÓN 1 MEDIDOR 4 PCF Y ESTACIÓN 2 MEDIDOR 1 HFR, PCF
	<	• ESTACIÓN 2 MEDIDOR 1 LFR, PCF
	=	• ESTACIÓN 1 MEDIDOR 4 PCF Y ESTACIÓN 2 MEDIDOR 1 LFR, PCF
>	• ESTACIÓN 2 MEDIDOR 1 HFR, LFR, PCF	
?	• ESTACIÓN 1 MEDIDOR 4 PCF Y ESTACIÓN 2 MEDIDOR 1 HFR, LFR, PCF	
ACTUA(23) • MEDIDOR ALARMAS	1	• ESTACIÓN 2 MEDIDOR 2 HFR
	2	• ESTACIÓN 2 MEDIDOR 2 LFR
	3	• ESTACIÓN 2 MEDIDOR 2 HFR, LFR
	4	• ESTACIÓN 2 MEDIDOR 2 PCF
	5	• ESTACIÓN 2 MEDIDOR 2 HFR, PCF
	6	• ESTACIÓN 2 MEDIDOR 2 LFR, PCF
	7	• ESTACIÓN 2 MEDIDOR 2 HFR, LFR, PCF
	8	• ESTACIÓN 2 MEDIDOR 3 HFR
	9	• ESTACIÓN 2 MEDIDORES 2,3 HFR
	:	• ESTACIÓN 2 MEDIDOR 2 LFR Y MEDIDOR 3 HFR
	;	• ESTACIÓN 2 MEDIDOR 2 HFR, LFR Y MEDIDOR 3 HFR
	<	• ESTACIÓN 2 MEDIDOR 2 PCF Y MEDIDOR 3 HFR
	=	• ESTACIÓN 2 MEDIDOR 2 HFR, PCF Y MEDIDOR 3 HFR
>	• ESTACIÓN 2 MEDIDOR 2 LFR, PCF Y MEDIDOR 3 HFR	

	?	<ul style="list-style-type: none"> <li>• ESTACIÓN 2 MEDIDOR 2 HFR, LFR, PCF Y MEDIDOR 3 HFR</li> </ul>
ACTUA(24) • MEDIDOR A • ALARMAS	1	<ul style="list-style-type: none"> <li>• ESTACIÓN 2 MEDIDOR 3 LFR</li> </ul>
	2	<ul style="list-style-type: none"> <li>• ESTACIÓN 2 MEDIDOR 3 PCF</li> </ul>
	3	<ul style="list-style-type: none"> <li>• ESTACIÓN 2 MEDIDOR 3 LFR, PCF</li> </ul>
	4	<ul style="list-style-type: none"> <li>• ESTACIÓN 2 MEDIDOR 4 HFR</li> </ul>
	5	<ul style="list-style-type: none"> <li>• ESTACIÓN 2 MEDIDOR 3 LFR Y MEDIDOR 4 HFR</li> </ul>
	6	<ul style="list-style-type: none"> <li>• ESTACIÓN 2 MEDIDOR 3 PCF Y MEDIDOR 4 HFR</li> </ul>
	7	<ul style="list-style-type: none"> <li>• ESTACIÓN 2 MEDIDOR 3 LFR, PCF Y MEDIDOR 4 HFR</li> </ul>
	8	<ul style="list-style-type: none"> <li>• ESTACIÓN 2 MEDIDOR 4 LFR</li> </ul>
	9	<ul style="list-style-type: none"> <li>• ESTACIÓN 2 MEDIDOR 3 LFR Y MEDIDOR 4 LFR</li> </ul>
	:	<ul style="list-style-type: none"> <li>• ESTACIÓN 2 MEDIDOR 3 PCF Y MEDIDOR 4 LFR</li> </ul>
	;	<ul style="list-style-type: none"> <li>• ESTACIÓN 2 MEDIDOR 3 LFR, PCF Y MEDIDOR 4 LFR</li> </ul>
<	<ul style="list-style-type: none"> <li>• ESTACIÓN 2 MEDIDOR 4 HFR, LFR</li> </ul>	
=	<ul style="list-style-type: none"> <li>• ESTACIÓN 2 MEDIDOR 3 LFR Y MEDIDOR 4 HFR, LFR</li> </ul>	
>	<ul style="list-style-type: none"> <li>• ESTACIÓN 2 MEDIDOR 3 PCF Y MEDIDOR 4 HFR, LFR</li> </ul>	
?	<ul style="list-style-type: none"> <li>• ESTACIÓN 2 MEDIDOR 3 LFR, PCF Y MEDIDOR 4 HFR, LFR</li> </ul>	
ACTUA(25) • MEDIDOR ALARMAS	1	<ul style="list-style-type: none"> <li>• ESTACIÓN 2 MEDIDOR 4 PCF</li> </ul>
	2	<ul style="list-style-type: none"> <li>• SPARE HFR</li> </ul>
	3	<ul style="list-style-type: none"> <li>• ESTACIÓN 2 MEDIDOR 4 PCF Y SPARE HFR</li> </ul>
	4	<ul style="list-style-type: none"> <li>• SPARE LFR</li> </ul>
	5	<ul style="list-style-type: none"> <li>• ESTACIÓN 2 MEDIDOR 4 PCF Y SPARE LFR</li> </ul>
	6	<ul style="list-style-type: none"> <li>• SPARE HFR, LFR</li> </ul>
	7	<ul style="list-style-type: none"> <li>• ESTACIÓN 2 MEDIDOR 4 PCF Y SPARE HFR, LFR</li> </ul>
	8	<ul style="list-style-type: none"> <li>• SPARE PCF</li> </ul>
	9	<ul style="list-style-type: none"> <li>• ESTACIÓN 2 MEDIDOR 4 PCF Y SPARE PCF</li> </ul>
	:	<ul style="list-style-type: none"> <li>• SPARE HFR, PCF</li> </ul>
	;	<ul style="list-style-type: none"> <li>• ESTACIÓN 2 MEDIDOR 4 PCF Y SPARE HFR, PCF</li> </ul>
<	<ul style="list-style-type: none"> <li>• SPARE LFR, PCF</li> </ul>	
=	<ul style="list-style-type: none"> <li>• ESTACIÓN 2 MEDIDOR 4 PCF Y SPARE LFR, PCF</li> </ul>	
>	<ul style="list-style-type: none"> <li>• SPARE HFR, LFR, PCF</li> </ul>	
?	<ul style="list-style-type: none"> <li>• ESTACIÓN 2 MEDIDOR 4 PCF Y SPARE HFR, LFR, PCF</li> </ul>	
ACTUA(26) • VÁLVULA ALARMAS	1	<ul style="list-style-type: none"> <li>• ESTACIÓN 1 MEDIDOR 1 FO</li> </ul>
	2	<ul style="list-style-type: none"> <li>• ESTACIÓN 1 MEDIDOR 1 FC</li> </ul>
	3	<ul style="list-style-type: none"> <li>• ESTACIÓN 1 MEDIDOR 1 FO, FC</li> </ul>
	4	<ul style="list-style-type: none"> <li>• ESTACIÓN 1 MEDIDOR 2 FO</li> </ul>
	5	<ul style="list-style-type: none"> <li>• ESTACIÓN 1 MEDIDOR 1 FO Y MEDIDOR 2 FO</li> </ul>
	6	<ul style="list-style-type: none"> <li>• ESTACIÓN 1 MEDIDOR 1 FC Y MEDIDOR 2 FO</li> </ul>
	7	<ul style="list-style-type: none"> <li>• ESTACIÓN 1 MEDIDOR 1 FO, FC Y MEDIDOR 2 FO</li> </ul>
	8	<ul style="list-style-type: none"> <li>• ESTACIÓN 1 MEDIDOR 2 FC</li> </ul>
	9	<ul style="list-style-type: none"> <li>• ESTACIÓN 1 MEDIDOR 1 FO Y MEDIDOR 2 FC</li> </ul>
	:	<ul style="list-style-type: none"> <li>• ESTACIÓN 1 MEDIDOR 1 FC Y MEDIDOR 2 FC</li> </ul>
;	<ul style="list-style-type: none"> <li>• ESTACIÓN 1 MEDIDOR 1 FO, FC Y MEDIDOR 2 FC</li> </ul>	
<	<ul style="list-style-type: none"> <li>• ESTACIÓN 1 MEDIDOR 2 FO, FC</li> </ul>	
=	<ul style="list-style-type: none"> <li>• ESTACIÓN 1 MEDIDOR 1 FO Y MEDIDOR 2 FO, FC</li> </ul>	
>	<ul style="list-style-type: none"> <li>• ESTACIÓN 1 MEDIDOR 1 FC Y MEDIDOR 2 FO, FC</li> </ul>	
?	<ul style="list-style-type: none"> <li>• ESTACIÓN 1 MEDIDOR 1 FO, FC Y MEDIDOR 2 FO, FC</li> </ul>	
	1	<ul style="list-style-type: none"> <li>• ESTACIÓN 1 MEDIDOR 3 FO</li> </ul>
	2	<ul style="list-style-type: none"> <li>• ESTACIÓN 1 MEDIDOR 3 FC</li> </ul>
	3	<ul style="list-style-type: none"> <li>• ESTACIÓN 1 MEDIDOR 3 FO, FC</li> </ul>

<p>ACTUA(27) • VÁLVULA ALARMAS</p>	<p>4 5 6 7 8 9 : ; &lt; = &gt; ?</p>	<ul style="list-style-type: none"> <li>• ESTACIÓN 1 MEDIDOR 4 FO</li> <li>• ESTACIÓN 1 MEDIDOR 3 FO Y MEDIDOR 4 FO</li> <li>• ESTACIÓN 1 MEDIDOR 3 FC Y MEDIDOR 4 FO</li> <li>• ESTACIÓN 1 MEDIDOR 3 FO, FC Y MEDIDOR 4 FO</li> <li>• ESTACIÓN 1 MEDIDOR 4 FC</li> <li>• ESTACIÓN 1 MEDIDOR 3 FO Y MEDIDOR 4 FC</li> <li>• ESTACIÓN 1 MEDIDOR 3 FC Y MEDIDOR 4 FC</li> <li>• ESTACIÓN 1 MEDIDOR 3 FO, FC Y MEDIDOR 4 FC</li> <li>• ESTACIÓN 1 MEDIDOR 4 FO, FC</li> <li>• ESTACIÓN 1 MEDIDOR 3 FO Y MEDIDOR 4 FO, FC</li> <li>• ESTACIÓN 1 MEDIDOR 3 FC Y MEDIDOR 4 FO, FC</li> <li>• ESTACIÓN 1 MEDIDOR 3 FO, FC Y MEDIDOR 4 FO, FC</li> </ul>
<p>ACTUA(28) • VÁLVULA ALARMAS</p>	<p>1 2 3 4 5 6 7 8 9 : ; &lt; = &gt; ?</p>	<ul style="list-style-type: none"> <li>• ESTACIÓN 2 MEDIDOR 1 FO</li> <li>• ESTACIÓN 2 MEDIDOR 1 FC</li> <li>• ESTACIÓN 2 MEDIDOR 1 FO, FC</li> <li>• ESTACIÓN 2 MEDIDOR 2 FO</li> <li>• ESTACIÓN 2 MEDIDOR 1 FO Y MEDIDOR 2 FO</li> <li>• ESTACIÓN 2 MEDIDOR 1 FC Y MEDIDOR 2 FO</li> <li>• ESTACIÓN 2 MEDIDOR 1 FO, FC Y MEDIDOR 2 FO</li> <li>• ESTACIÓN 2 MEDIDOR 2 FC</li> <li>• ESTACIÓN 2 MEDIDOR 1 FO Y MEDIDOR 2 FC</li> <li>• ESTACIÓN 2 MEDIDOR 1 FC Y MEDIDOR 2 FC</li> <li>• ESTACIÓN 2 MEDIDOR 1 FO, FC Y MEDIDOR 2 FC</li> <li>• ESTACIÓN 2 MEDIDOR 2 FO, FC</li> <li>• ESTACIÓN 2 MEDIDOR 1 FO Y MEDIDOR 2 FO, FC</li> <li>• ESTACIÓN 2 MEDIDOR 1 FC Y MEDIDOR 2 FO, FC</li> <li>• ESTACIÓN 2 MEDIDOR 1 FO, FC Y MEDIDOR 2 FO, FC</li> </ul>
<p>ACTUA(29) • VÁLVULA ALARMAS</p>	<p>1 2 3 4 5 6 7 8 9 : ; &lt; = &gt; ?</p>	<ul style="list-style-type: none"> <li>• ESTACIÓN 2 MEDIDOR 3 FO</li> <li>• ESTACIÓN 2 MEDIDOR 3 FC</li> <li>• ESTACIÓN 2 MEDIDOR 3 FO, FC</li> <li>• ESTACIÓN 2 MEDIDOR 4 FO</li> <li>• ESTACIÓN 2 MEDIDOR 3 FO Y MEDIDOR 4 FO</li> <li>• ESTACIÓN 2 MEDIDOR 3 FC Y MEDIDOR 4 FO</li> <li>• ESTACIÓN 2 MEDIDOR 3 FO, FC Y MEDIDOR 4 FO</li> <li>• ESTACIÓN 2 MEDIDOR 4 FC</li> <li>• ESTACIÓN 2 MEDIDOR 3 FO Y MEDIDOR 4 FC</li> <li>• ESTACIÓN 2 MEDIDOR 3 FC Y MEDIDOR 4 FC</li> <li>• ESTACIÓN 2 MEDIDOR 3 FO, FC Y MEDIDOR 4 FC</li> <li>• ESTACIÓN 2 MEDIDOR 4 FO, FC</li> <li>• ESTACIÓN 2 MEDIDOR 3 FO Y MEDIDOR 4 FO, FC</li> <li>• ESTACIÓN 2 MEDIDOR 3 FC Y MEDIDOR 4 FO, FC</li> <li>• ESTACIÓN 2 MEDIDOR 3 FO, FC Y MEDIDOR 4 FO, FC</li> </ul>
<p>ACTUA(30) • VÁLVULA ALARMAS</p>	<p>1 2 3 4 5 6 7 8 9 :</p>	<ul style="list-style-type: none"> <li>• SPARE CS ESTACIÓN 1 FO</li> <li>• SPARE CS ESTACIÓN 1 FC</li> <li>• SPARE CS ESTACIÓN 1 FO, FC</li> <li>• SPARE CS ESTACIÓN 2 FO</li> <li>• SPARE CS ESTACIÓN 1 FO Y ESTACIÓN 2 FO</li> <li>• SPARE CS ESTACIÓN 1 FC Y ESTACIÓN 2 FO</li> <li>• SPARE CS ESTACIÓN 1 FO, FC Y ESTACIÓN 2 FO</li> <li>• SPARE CS ESTACIÓN 2 FC</li> <li>• SPARE CS ESTACIÓN 1 FO Y ESTACIÓN 2 FC</li> <li>• SPARE CS ESTACIÓN 1 FC Y ESTACIÓN 2 FC</li> </ul>

	; < = > ?	<ul style="list-style-type: none"> <li>• SPARE CS ESTACIÓN 1 FO, FC Y ESTACIÓN 2 FC</li> <li>• SPARE CS ESTACIÓN 2 FO, FC</li> <li>• SPARE CS ESTACIÓN 1 FO Y ESTACIÓN 2 FO, FC</li> <li>• SPARE CS ESTACIÓN 1 FC Y ESTACIÓN 2 FO, FC</li> <li>• SPARE CS ESTACIÓN 1 FO, FC Y ESTACIÓN 2 FO, FC</li> </ul>
ACTUA(31) • VÁLVULA ALARMAS	1 2 3 4 5 6 7 8 9 : ; < = > ?	<ul style="list-style-type: none"> <li>• ESTACIÓN 1 EP MEDIDOR 1 FO</li> <li>• ESTACIÓN 1 EP MEDIDOR 1 FC</li> <li>• ESTACIÓN 1 EP MEDIDOR 1 FO, FC</li> <li>• ESTACIÓN 1 EP MEDIDOR 2 FO</li> <li>• ESTACIÓN 1 EP MEDIDOR 1 FO Y MEDIDOR 2 FO</li> <li>• ESTACIÓN 1 EP MEDIDOR 1 FC Y MEDIDOR 2 FO</li> <li>• ESTACIÓN 1 EP MEDIDOR 1 FO, FC Y MEDIDOR 2 FO</li> <li>• ESTACIÓN 1 EP MEDIDOR 2 FC</li> <li>• ESTACIÓN 1 EP MEDIDOR 1 FO Y MEDIDOR 2 FC</li> <li>• ESTACIÓN 1 EP MEDIDOR 1 FC Y MEDIDOR 2 FC</li> <li>• ESTACIÓN 1 EP MEDIDOR 1 FO, FC Y MEDIDOR 2 FC</li> <li>• ESTACIÓN 1 EP MEDIDOR 2 FO, FC</li> <li>• ESTACIÓN 1 EP MEDIDOR 1 FO Y MEDIDOR 2 FO, FC</li> <li>• ESTACIÓN 1 EP MEDIDOR 1 FC Y MEDIDOR 2 FO, FC</li> <li>• ESTACIÓN 1 EP MEDIDOR 1 FO, FC Y MEDIDOR 2 FO, FC</li> </ul>
ACTUA(32) • VÁLVULA ALARMAS	1 2 3 4 5 6 7 8 9 : ; < = > ?	<ul style="list-style-type: none"> <li>• ESTACIÓN 1 EP MEDIDOR 3 FO</li> <li>• ESTACIÓN 1 EP MEDIDOR 3 FC</li> <li>• ESTACIÓN 1 EP MEDIDOR 3 FO, FC</li> <li>• ESTACIÓN 1 EP MEDIDOR 4 FO</li> <li>• ESTACIÓN 1 EP MEDIDOR 3 FO Y MEDIDOR 4 FO</li> <li>• ESTACIÓN 1 EP MEDIDOR 3 FC Y MEDIDOR 4 FO</li> <li>• ESTACIÓN 1 EP MEDIDOR 3 FO, FC Y MEDIDOR 4 FO</li> <li>• ESTACIÓN 1 EP MEDIDOR 4 FC</li> <li>• ESTACIÓN 1 EP MEDIDOR 3 FO Y MEDIDOR 4 FC</li> <li>• ESTACIÓN 1 EP MEDIDOR 3 FC Y MEDIDOR 4 FC</li> <li>• ESTACIÓN 1 EP MEDIDOR 3 FO, FC Y MEDIDOR 4 FC</li> <li>• ESTACIÓN 1 EP MEDIDOR 4 FO, FC</li> <li>• ESTACIÓN 1 EP MEDIDOR 3 FO Y MEDIDOR 4 FO, FC</li> <li>• ESTACIÓN 1 EP MEDIDOR 3 FC Y MEDIDOR 4 FO, FC</li> <li>• ESTACIÓN 1 EP MEDIDOR 3 FO, FC Y MEDIDOR 4 FO, FC</li> </ul>
ACTUA(33) • VÁLVULA ALARMAS	1 2 3 4 5 6 7 8 9 : ;	<ul style="list-style-type: none"> <li>• ESTACIÓN 2 EP MEDIDOR 1 FO</li> <li>• ESTACIÓN 2 EP MEDIDOR 1 FC</li> <li>• ESTACIÓN 2 EP MEDIDOR 1 FO, FC</li> <li>• ESTACIÓN 2 EP MEDIDOR 2 FO</li> <li>• ESTACIÓN 2 EP MEDIDOR 1 FO Y MEDIDOR 2 FO</li> <li>• ESTACIÓN 2 EP MEDIDOR 1 FC Y MEDIDOR 2 FO</li> <li>• ESTACIÓN 2 EP MEDIDOR 1 FO, FC Y MEDIDOR 2 FO</li> <li>• ESTACIÓN 2 EP MEDIDOR 2 FC</li> <li>• ESTACIÓN 2 EP MEDIDOR 1 FO Y MEDIDOR 2 FC</li> <li>• ESTACIÓN 2 EP MEDIDOR 1 FC Y MEDIDOR 2 FC</li> <li>• ESTACIÓN 2 EP MEDIDOR 1 FO, FC Y MEDIDOR 2 FC</li> </ul>

CAPITULO 3. DESARROLLO DEL SOFTWARE

	<ul style="list-style-type: none"> <li>&lt;</li> <li>=</li> <li>&gt;</li> <li>?</li> </ul>	<ul style="list-style-type: none"> <li>• ESTACIÓN 2 EP MEDIDOR 2 FO, FC</li> <li>• ESTACIÓN 2 EP MEDIDOR 1 FO Y MEDIDOR 2 FO, FC</li> <li>• ESTACIÓN 2 EP MEDIDOR 1 FC Y MEDIDOR 2 FO, FC</li> <li>• ESTACIÓN 2 EP MEDIDOR 1 FO, FC Y MEDIDOR 2 FO, FC</li> </ul>
<ul style="list-style-type: none"> <li>• ACTUA(34)</li> <li>• VÁLVULA ALARMAS</li> </ul>	<ul style="list-style-type: none"> <li>1</li> <li>2</li> <li>3</li> <li>4</li> <li>5</li> <li>6</li> <li>7</li> <li>8</li> <li>9</li> <li>:</li> <li>:</li> <li>&lt;</li> <li>=</li> <li>&gt;</li> <li>?</li> </ul>	<ul style="list-style-type: none"> <li>• ESTACIÓN 2 EP MEDIDOR 3 FO</li> <li>• ESTACIÓN 2 EP MEDIDOR 3 FC</li> <li>• ESTACIÓN 2 EP MEDIDOR 3 FO, FC</li> <li>• ESTACIÓN 2 EP MEDIDOR 4 FO</li> <li>• ESTACIÓN 2 EP MEDIDOR 3 FO Y MEDIDOR 4 FO</li> <li>• ESTACIÓN 2 EP MEDIDOR 3 FC Y MEDIDOR 4 FO</li> <li>• ESTACIÓN 2 EP MEDIDOR 3 FO, FC Y MEDIDOR 4 FO</li> <li>• ESTACIÓN 2 EP MEDIDOR 4 FC</li> <li>• ESTACIÓN 2 EP MEDIDOR 3 FO Y MEDIDOR 4 FC</li> <li>• ESTACIÓN 2 EP MEDIDOR 3 FC Y MEDIDOR 4 FC</li> <li>• ESTACIÓN 2 EP MEDIDOR 3 FO, FC Y MEDIDOR 4 FC</li> <li>• ESTACIÓN 2 EP MEDIDOR 4 FO, FC</li> <li>• ESTACIÓN 2 EP MEDIDOR 3 FO Y MEDIDOR 4 FO, FC</li> <li>• ESTACIÓN 2 EP MEDIDOR 3 FC Y MEDIDOR 4 FO, FC</li> <li>• ESTACIÓN 2 EP MEDIDOR 3 FO, FC Y MEDIDOR 4 FO, FC</li> </ul>
<ul style="list-style-type: none"> <li>• ACTUA(35)</li> <li>• VÁLVULA ALARMAS</li> </ul>	<ul style="list-style-type: none"> <li>1</li> <li>2</li> <li>3</li> <li>4</li> <li>5</li> <li>6</li> <li>7</li> <li>8</li> <li>9</li> <li>:</li> <li>:</li> <li>&lt;</li> <li>=</li> <li>&gt;</li> <li>?</li> </ul>	<ul style="list-style-type: none"> <li>• SPARE EP FO</li> <li>• SPARE EP FC</li> <li>• SPARE EP FO, FC</li> <li>• SPARE CE ESTACIÓN 1 FO</li> <li>• SPARE EP FO Y CE ESTACIÓN 1 FO</li> <li>• SPARE EP FC Y CE ESTACIÓN 1 FO</li> <li>• SPARE EP FO, FC Y CE ESTACIÓN 1 FO</li> <li>• SPARE CE ESTACIÓN 1 FC</li> <li>• SPARE EP FO Y CE ESTACIÓN 1 FC</li> <li>• SPARE EP FC Y CE ESTACIÓN 1 FC</li> <li>• SPARE EP FO, FC Y CE ESTACIÓN 1 FC</li> <li>• SPARE CE ESTACIÓN 1 FO, FC</li> <li>• SPARE EP FO Y CE ESTACIÓN 1 FO, FC</li> <li>• SPARE EP FC Y CE ESTACIÓN 1 FO, FC</li> <li>• SPARE EP FO, FC Y CE ESTACIÓN 1 FO, FC</li> </ul>
<ul style="list-style-type: none"> <li>• ACTUA(36)</li> <li>• VÁLVULA ALARMAS</li> </ul>	<ul style="list-style-type: none"> <li>1</li> <li>2</li> <li>3</li> <li>4</li> <li>5</li> <li>6</li> <li>7</li> <li>8</li> <li>9</li> <li>:</li> <li>:</li> <li>:</li> </ul>	<ul style="list-style-type: none"> <li>• SPARE CE ESTACIÓN 2 FO</li> <li>• SPARE CE ESTACIÓN 2 FC</li> <li>• SPARE CE ESTACIÓN 2 FO, FC</li> <li>• RP ESTACIÓN 1 FO</li> <li>• SPARE CE ESTACIÓN 2 FO Y RP ESTACIÓN 1 FO</li> <li>• SPARE CE ESTACIÓN 2 FC Y RP ESTACIÓN 1 FO</li> <li>• SPARE CE ESTACIÓN 2 FO, FC Y RP ESTACIÓN 1 FO</li> <li>• RP ESTACIÓN 1 FC</li> <li>• SPARE CE ESTACIÓN 2 FO Y RP ESTACIÓN 1 FC</li> <li>• SPARE CE ESTACIÓN 2 FC Y RP ESTACIÓN 1 FC</li> <li>• SPARE CE ESTACIÓN 2 FO, FC Y RP ESTACIÓN 1 FC</li> </ul>

	< = > ?	<ul style="list-style-type: none"> <li>• RP ESTACIÓN 1 E FO, FC</li> <li>• SPARE CE ESTACIÓN 2 FO Y RP ESTACIÓN 1 FO, FC</li> <li>• SPARE CE ESTACIÓN 2 FC Y RP ESTACIÓN 1 FO, FC</li> <li>• SPARE CE ESTACIÓN 2 FO, FC Y RP ESTACIÓN 1 FO, FC</li> </ul>
<p>ACTUA(37)</p> <ul style="list-style-type: none"> <li>• VÁLVULA ALARMAS/MEDIDOR TEMPERATURA</li> </ul>	1 2 3 4 5 6 7 8 9 : ; < = > ?	<ul style="list-style-type: none"> <li>• RP ESTACIÓN 2 FO</li> <li>• RP ESTACIÓN 2 FC</li> <li>• RP ESTACIÓN 2 FO, FC</li> <li>• ESTACIÓN 1 MEDIDOR 1 ALTA</li> <li>• RP ESTACIÓN 2 FO Y ESTACIÓN 1 MEDIDOR 1 ALTA</li> <li>• RP ESTACIÓN 2 FC Y ESTACIÓN 1 MEDIDOR 1 ALTA</li> <li>• RP ESTACIÓN 2 FO, FC Y ESTACIÓN 1 MEDIDOR 1 ALTA</li> <li>• ESTACIÓN 1 MEDIDOR 1 BAJA</li> <li>• RP ESTACIÓN 2 FO Y ESTACIÓN 1 MEDIDOR 1 BAJA</li> <li>• RP ESTACIÓN 2 FC Y ESTACIÓN 1 MEDIDOR 1 BAJA</li> <li>• RP ESTACIÓN 2 FO, FC Y ESTACIÓN 1 MEDIDOR 1 BAJA</li> <li>• ESTACIÓN 1 MEDIDOR 1 ALTA, BAJA</li> <li>• RP ESTACIÓN 2 FO Y ESTACIÓN 1 MEDIDOR 1 ALTA, BAJA</li> <li>• RP ESTACIÓN 2 FC Y ESTACIÓN 1 MEDIDOR 1 ALTA, BAJA</li> <li>• RP ESTACIÓN 2 FO, FC Y ESTACIÓN 1 MEDIDOR 1 ALTA, BAJA</li> </ul>
<p>ACTUA(38)</p> <ul style="list-style-type: none"> <li>• MEDIDOR TEMPERATURA</li> </ul>	1 2 3 4 5 6 7 8 9 : ; < = > ?	<ul style="list-style-type: none"> <li>• ESTACIÓN 1 MEDIDOR 2 ALTA</li> <li>• ESTACIÓN 1 MEDIDOR 2 BAJA</li> <li>• ESTACIÓN 1 MEDIDOR 2 ALTA, BAJA</li> <li>• ESTACIÓN 1 MEDIDOR 3 ALTA</li> <li>• ESTACIÓN 1 MEDIDOR 2 ALTA Y MEDIDOR 3 ALTA</li> <li>• ESTACIÓN 1 MEDIDOR 2 BAJA Y MEDIDOR 3 ALTA</li> <li>• ESTACIÓN 1 MEDIDOR 2 ALTA, BAJA Y MEDIDOR 3 ALTA</li> <li>• ESTACIÓN 1 MEDIDOR 3 BAJA</li> <li>• ESTACIÓN 1 MEDIDOR 2 ALTA Y MEDIDOR 3 BAJA</li> <li>• ESTACIÓN 1 MEDIDOR 2 BAJA Y MEDIDOR 3 BAJA</li> <li>• ESTACIÓN 1 MEDIDOR 2 ALTA, BAJA Y MEDIDOR 3 BAJA</li> <li>• ESTACIÓN 1 MEDIDOR 3 ALTA, BAJA</li> <li>• ESTACIÓN 1 MEDIDOR 2 ALTA Y MEDIDOR 3 ALTA, BAJA</li> <li>• ESTACIÓN 1 MEDIDOR 2 BAJA Y MEDIDOR 3 ALTA, BAJA</li> <li>• ESTACIÓN 1 MEDIDOR 2 ALTA, BAJA Y MEDIDOR 3 ALTA, BAJA</li> </ul>
	1 2 3 4 5	<ul style="list-style-type: none"> <li>• ESTACIÓN 1 MEDIDOR 4 ALTA</li> <li>• ESTACIÓN 1 MEDIDOR 4 BAJA</li> <li>• ESTACIÓN 1 MEDIDOR 4 ALTA, BAJA</li> <li>• ESTACIÓN 2 MEDIDOR 1 ALTA</li> <li>• ESTACIÓN 1 MEDIDOR 4 ALTA Y ESTACIÓN 2 MEDIDOR 1 ALTA</li> </ul>



ACTUA(39) • MEDIDOR TEMPERATUR A	6	• ESTACIÓN 1 MEDIDOR 4 BAJA Y ESTACIÓN 2 MEDIDOR 1 ALTA
	7	• ESTACIÓN 1 MEDIDOR 4 ALTA, BAJA Y ESTACIÓN 2 MEDIDOR 1 ALTA
	8	• ESTACIÓN 2 MEDIDOR 1 BAJA
	9	• ESTACIÓN 1 MEDIDOR 4 ALTA Y ESTACIÓN 2 MEDIDOR 1 BAJA
	:	• ESTACIÓN 1 MEDIDOR 4 BAJA Y ESTACIÓN 2 MEDIDOR 1 BAJA
	;	• ESTACIÓN 1 MEDIDOR 4 ALTA, BAJA Y ESTACIÓN 2 MEDIDOR 1 BAJA
	<	• ESTACIÓN 2 MEDIDOR 1 ALTA, BAJA
	=	• ESTACIÓN 1 MEDIDOR 4 ALTA Y ESTACIÓN 2 MEDIDOR 1 ALTA, BAJA
	>	• ESTACIÓN 1 MEDIDOR 4 BAJA Y ESTACIÓN 2 MEDIDOR 1 ALTA, BAJA
?	• ESTACIÓN 1 MEDIDOR 4 ALTA, BAJA Y ESTACIÓN 2 MEDIDOR 1 ALTA, BAJA	
ACTUA(40) • MEDIDOR TEMPERATUR A	1	• ESTACIÓN 2 MEDIDOR 2 ALTA
	2	• ESTACIÓN 2 MEDIDOR 2 BAJA
	3	• ESTACIÓN 2 MEDIDOR 2 ALTA, BAJA
	4	• ESTACIÓN 2 MEDIDOR 3 ALTA
	5	• ESTACIÓN 2 MEDIDOR 2 ALTA Y MEDIDOR 3 ALTA
	6	• ESTACIÓN 2 MEDIDOR 2 BAJA Y MEDIDOR 3 ALTA
	7	• ESTACIÓN 2 MEDIDOR 2 ALTA, BAJA Y MEDIDOR 3 ALTA
	8	• ESTACIÓN 2 MEDIDOR 3 BAJA
	9	• ESTACIÓN 2 MEDIDOR 2 ALTA Y MEDIDOR 3 BAJA
	:	• ESTACIÓN 2 MEDIDOR 2 BAJA Y MEDIDOR 3 BAJA
	;	• ESTACIÓN 2 MEDIDOR 2 ALTA, BAJA Y MEDIDOR 3 BAJA
	<	• ESTACIÓN 2 MEDIDOR 3 ALTA, BAJA
=	• ESTACIÓN 2 MEDIDOR 2 ALTA Y MEDIDOR 3 ALTA, BAJA	
>	• ESTACIÓN 2 MEDIDOR 2 BAJA Y MEDIDOR 3 ALTA, BAJA	
?	• ESTACIÓN 2 MEDIDOR 2 ALTA, BAJA Y MEDIDOR 3 ALTA, BAJA	
ACTUA(41) • MEDIDOR TEMPERATUR A	1	• ESTACIÓN 2 MEDIDOR 4 ALTA
	2	• ESTACIÓN 2 MEDIDOR 4 BAJA
	3	• ESTACIÓN 2 MEDIDOR 4 ALTA, BAJA
	4	• SPARE ALTA
	5	• ESTACIÓN 2 MEDIDOR 4 ALTA Y SPARE ALTA
	6	• ESTACIÓN 2 MEDIDOR 4 BAJA Y SPARE ALTA
	7	• ESTACIÓN 2 MEDIDOR 4 ALTA, BAJA Y SPARE ALTA
	8	• SPARE BAJA
	9	• ESTACIÓN 2 MEDIDOR 4 ALTA Y SPARE BAJA
	:	• ESTACIÓN 2 MEDIDOR 4 BAJA Y SPARE BAJA
	;	• ESTACIÓN 2 MEDIDOR 4 ALTA, BAJA Y SPARE BAJA
	<	• SPARE ALTA, BAJA
=	• ESTACIÓN 2 MEDIDOR 4 ALTA Y SPARE ALTA, BAJA	
>	• ESTACIÓN 2 MEDIDOR 4 BAJA Y SPARE ALTA, BAJA	
?	• ESTACIÓN 2 MEDIDOR 4 ALTA, BAJA Y SPARE ALTA,	

		BAJA
ACTUA(42) • PRESIÓN DEL CABEZAL DE ENTRADA	1	• ESTACIÓN 1 ALTA
	2	• ESTACIÓN 1 BAJA
	3	• ESTACIÓN 1 ALTA, BAJA
	4	• ESTACIÓN 2 ALTA
	5	• ESTACIÓN 1 ALTA Y ESTACIÓN 2 ALTA
	6	• ESTACIÓN 1 BAJA Y ESTACIÓN 2 ALTA
	7	• ESTACIÓN 1 ALTA, BAJA Y ESTACIÓN 2 ALTA
	8	• ESTACIÓN 2 BAJA
	9	• ESTACIÓN 1 ALTA Y ESTACIÓN 2 BAJA
	:	• ESTACIÓN 1 BAJA Y ESTACIÓN 2 BAJA
	;	• ESTACIÓN 1 ALTA, BAJA Y ESTACIÓN 2 BAJA
	<	• ESTACIÓN 2 ALTA, BAJA
=	• ESTACIÓN 1 ALTA Y ESTACIÓN 2 ALTA, BAJA	
>	• ESTACIÓN 1 BAJA Y ESTACIÓN 2 ALTA, BAJA	
?	• ESTACIÓN 1 ALTA, BAJA Y ESTACIÓN 2 ALTA, BAJA	
ACTUA(43) • PRESIÓN DEL CABEZAL DE SALIDA	1	• ESTACIÓN 1 ALTA
	2	• ESTACIÓN 1 BAJA
	3	• ESTACIÓN 1 ALTA, BAJA
	4	• ESTACIÓN 2 ALTA
	5	• ESTACIÓN 1 ALTA Y ESTACIÓN 2 ALTA
	6	• ESTACIÓN 1 BAJA Y ESTACIÓN 2 ALTA
	7	• ESTACIÓN 1 ALTA, BAJA Y ESTACIÓN 2 ALTA
	8	• ESTACIÓN 2 BAJA
	9	• ESTACIÓN 1 ALTA Y ESTACIÓN 2 BAJA
	:	• ESTACIÓN 1 BAJA Y ESTACIÓN 2 BAJA
	;	• ESTACIÓN 1 ALTA, BAJA Y ESTACIÓN 2 BAJA
	<	• ESTACIÓN 2 ALTA, BAJA
=	• ESTACIÓN 1 ALTA Y ESTACIÓN 2 ALTA, BAJA	
>	• ESTACIÓN 1 BAJA Y ESTACIÓN 2 ALTA, BAJA	
?	• ESTACIÓN 1 ALTA, BAJA Y ESTACIÓN 2 ALTA, BAJA	
ACTUA(44) • VISCOSIDAD	1	• ESTACIÓN 1 ALTA
	2	• ESTACIÓN 1 BAJA
	3	• ESTACIÓN 1 ALTA, BAJA
	4	• ESTACIÓN 2 ALTA
	5	• ESTACIÓN 1 ALTA Y ESTACIÓN 2 ALTA
	6	• ESTACIÓN 1 BAJA Y ESTACIÓN 2 ALTA
	7	• ESTACIÓN 1 ALTA, BAJA Y ESTACIÓN 2 ALTA
	8	• ESTACIÓN 2 BAJA
	9	• ESTACIÓN 1 ALTA Y ESTACIÓN 2 BAJA
	:	• ESTACIÓN 1 BAJA Y ESTACIÓN 2 BAJA
	;	• ESTACIÓN 1 ALTA, BAJA Y ESTACIÓN 2 BAJA
	<	• ESTACIÓN 2 ALTA, BAJA
=	• ESTACIÓN 1 ALTA Y ESTACIÓN 2 ALTA, BAJA	
>	• ESTACIÓN 1 BAJA Y ESTACIÓN 2 ALTA, BAJA	
?	• ESTACIÓN 1 ALTA, BAJA Y ESTACIÓN 2 ALTA, BAJA	
ACTUA(45) • BS&W	1	• ESTACIÓN 1 ALTA
	2	• ESTACIÓN 1 BAJA
	3	• ESTACIÓN 1 ALTA, BAJA
	4	• ESTACIÓN 2 ALTA
	5	• ESTACIÓN 1 ALTA Y ESTACIÓN 2 ALTA
	6	• ESTACIÓN 1 BAJA Y ESTACIÓN 2 ALTA

	7 8 9 : ; < = > ?	<ul style="list-style-type: none"> <li>• ESTACIÓN 1 ALTA, BAJA Y ESTACIÓN 2 ALTA</li> <li>• ESTACIÓN 2 BAJA</li> <li>• ESTACIÓN 1 ALTA Y ESTACIÓN 2 BAJA</li> <li>• ESTACIÓN 1 BAJA Y ESTACIÓN 2 BAJA</li> <li>• ESTACIÓN 1 ALTA, BAJA Y ESTACIÓN 2 BAJA</li> <li>• ESTACIÓN 2 ALTA, BAJA</li> <li>• ESTACIÓN 1 ALTA Y ESTACIÓN 2 ALTA, BAJA</li> <li>• ESTACIÓN 1 BAJA Y ESTACIÓN 2 ALTA, BAJA</li> <li>• ESTACIÓN 1 ALTA, BAJA Y ESTACIÓN 2 ALTA, BAJA</li> </ul>
<p>ACTUA(46)</p> <ul style="list-style-type: none"> <li>• TEMPERATURA Y PRESIÓN PROBADOR</li> </ul>	1 2 3 4 5 6 7 8 9 : ; < = > ?	<ul style="list-style-type: none"> <li>• PROBADOR TEMPERATURA ALTA</li> <li>• PROBADOR TEMPERATURA BAJA</li> <li>• PROBADOR TEMPERATURA ALTA, BAJA</li> <li>• PROBADOR PRESIÓN ALTA</li> <li>• PROBADOR TEMPERATURA ALTA Y PROBADOR PRESIÓN ALTA</li> <li>• PROBADOR TEMPERATURA BAJA Y PROBADOR PRESIÓN ALTA</li> <li>• PROBADOR TEMPERATURA ALTA, BAJA Y PROBADOR PRESIÓN ALTA</li> <li>• PROBADOR PRESIÓN BAJA</li> <li>• PROBADOR TEMPERATURA ALTA Y PROBADOR PRESIÓN BAJA</li> <li>• PROBADOR TEMPERATURA BAJA Y PROBADOR PRESIÓN BAJA</li> <li>• PROBADOR TEMPERATURA ALTA, BAJA Y PROBADOR PRESIÓN BAJA</li> <li>• PROBADOR PRESIÓN ALTA, BAJA</li> <li>• PROBADOR TEMPERATURA ALTA Y PROBADOR PRESIÓN ALTA, BAJA</li> <li>• PROBADOR TEMPERATURA BAJA Y PROBADOR PRESIÓN ALTA, BAJA</li> <li>• PROBADOR TEMPERATURA ALTA, BAJA Y PROBADOR PRESIÓN ALTA, BAJA</li> </ul>
<p>ACTUA(47)</p> <ul style="list-style-type: none"> <li>• GRAVEDAD ESPECIFICA</li> </ul>	1 2 3 4 5 6 7 8 9 : ; < = > ?	<ul style="list-style-type: none"> <li>• ESTACIÓN 1 ALTA</li> <li>• ESTACIÓN 1 BAJA</li> <li>• ESTACIÓN 1 ALTA, BAJA</li> <li>• ESTACIÓN 2 ALTA</li> <li>• ESTACIÓN 1 ALTA Y ESTACIÓN 2 ALTA</li> <li>• ESTACIÓN 1 BAJA Y ESTACIÓN 2 ALTA</li> <li>• ESTACIÓN 1 ALTA, BAJA Y ESTACIÓN 2 ALTA</li> <li>• ESTACIÓN 2 BAJA</li> <li>• ESTACIÓN 1 ALTA Y ESTACIÓN 2 BAJA</li> <li>• ESTACIÓN 1 BAJA Y ESTACIÓN 2 BAJA</li> <li>• ESTACIÓN 1 ALTA, BAJA Y ESTACIÓN 2 BAJA</li> <li>• ESTACIÓN 2 ALTA, BAJA</li> <li>• ESTACIÓN 1 ALTA Y ESTACIÓN 2 ALTA, BAJA</li> <li>• ESTACIÓN 1 BAJA Y ESTACIÓN 2 ALTA, BAJA</li> <li>• ESTACIÓN 1 ALTA, BAJA Y ESTACIÓN 2 ALTA, BAJA</li> </ul>

De acuerdo a esta tabla para el ejemplo anterior se tiene que:

La estación uno esta pasando al estado de suspensión, sin llegar a él todavía, la estación dos se encuentra inactiva y las pruebas de la estación uno han sido completadas.

### Delivery Ticket

Cuando se elige esta solicitud el controlador Daniel MM3000 manda la cadena con la información correspondiente a lo que se visualizará y/o imprimirá. La cadena de respuesta puede tener cuatro tamaños diferentes:

- 109 si el ticket esta en proceso. Esto sucede cuando la carga del buque-tanque se ha completado, pero aún se están contabilizando los totales.
- 292 si la prueba fue suspendida. Se necesita concluir un ciclo de cuatro pruebas para presentar la información correspondiente a éstas.
- 1012 si el spare no se encuentra seleccionado y se concluyeron las pruebas para la banda de medición a imprimir.
- 1060 si se completaron las pruebas y se tiene seleccionado el spare en la banda de medición.

El driver de comunicación le da el formato CSV a la cadena de caracteres recibidos y la envía a la interfaz de operación. En el caso de la cadena de 109 caracteres el driver de comunicación genera un mensaje indicando que el ticket esta en proceso, éste se entrega a la interfaz de operación para ser mostrado al operador.

Los caracteres recibidos tienen un significado diferente de acuerdo a la posición en que son recibidos, a continuación se explica este.

DELTIC1	SIGNIFICADO
1	01 H ASCII ACCESO AL DANIEL MM3000
2	34 H ASCII SOLICITUD DELIVERY TICKET
3	31 H Ó 32 H ASCII BANDA DE MEDICIÓN 1 Ó 2
4	20 H ASCII
5 AL 52	INFORMACIÓN DE LA TERMINAL MARÍTIMA
53 AL 100	20 H ASCII
101 AL 148	NOMBRE DEL BUQUE-TANQUE
149 AL 196	NÚMERO DEL EMBARQUE
197 AL 244	NÚMERO DEL MUELLE

La estación uno esta pasando al estado de suspensión, sin llegar a él todavía, la estación dos se encuentra inactiva y las pruebas de la estación uno han sido completadas.

#### Delivery Ticket

Cuando se elige esta solicitud el controlador Daniel MM3000 manda la cadena con la información correspondiente a lo que se visualizará y/o imprimirá, La cadena de respuesta puede tener cuatro tamaños diferentes:

- 109 si el ticket esta en proceso. Esto sucede cuando la carga del buque-tanque se ha completado, pero aún se están contabilizando los totales.
- 292 si la prueba fue suspendida. Se necesita concluir un ciclo de cuatro pruebas para presentar la información correspondiente a éstas.
- 1012 si el spare no se encuentra seleccionado y se concluyeron las pruebas para la banda de medición a imprimir.
- 1060 si se completaron las pruebas y se tiene seleccionado el spare en la banda de medición.

El driver de comunicación le da el formato CSV a la cadena de caracteres recibidos y la envía a la interfaz de operación. En el caso de la cadena de 109 caracteres el driver de comunicación genera un mensaje indicando que el ticket esta en proceso, éste se entrega a la interfaz de operación para ser mostrado al operador.

Los caracteres recibidos tienen un significado diferente de acuerdo a la posición en que son recibidos, a continuación se explica este.

DELTIC1	SIGNIFICADO
1	01 H ASCII ACCESO AL DANIEL MM3000
2	34 H ASCII SOLICITUD DELIVERY TICKET
3	31 H Ó 32 H ASCII BANDA DE MEDICIÓN 1 Ó 2
4	20 H ASCII
5 AL 52	INFORMACIÓN DE LA TERMINAL MARÍTIMA
53 AL 100	20 H ASCII
101 AL 148	NOMBRE DEL BUQUE-TANQUE
149 AL 196	NÚMERO DEL EMBARQUE
197 AL 244	NÚMERO DEL MUELLE

245 AL 292	FECHA DE EMBARQUE
293 AL 340	20 H ASCII
341 AL 388	TÍTULOS DEL MEDIDOR, FACTOR DE MEDICIÓN, TEMPERATURA, TOTAL BRUTO Y TOTAL NETO
389 AL 436	20 H ASCII
437 AL 484	INFORMACIÓN DEL MEDIDOR, FACTOR DE MEDICIÓN, TEMPERATURA, TOTAL BRUTO Y TOTAL NETO 1
485 AL 532	INFORMACIÓN DEL MEDIDOR, FACTOR DE MEDICIÓN, TEMPERATURA, TOTAL BRUTO Y TOTAL NETO 2
533 AL 580	INFORMACIÓN DEL MEDIDOR, FACTOR DE MEDICIÓN, TEMPERATURA, TOTAL BRUTO Y TOTAL NETO 3
581 AL 628	INFORMACIÓN DEL MEDIDOR, FACTOR DE MEDICIÓN, TEMPERATURA, TOTAL BRUTO Y TOTAL NETO 4
629 AL 676	INFORMACIÓN DEL MEDIDOR, FACTOR DE MEDICIÓN, TEMPERATURA, TOTAL BRUTO Y TOTAL NETO 5 (SPARE)
677 AL 724	20 H ASCII
725 AL 772	20 H ASCII
773 AL 820	INFORMACIÓN DE BS&W
821 AL 868	INFORMACIÓN DE LA VISCOSIDAD
869 AL 916	INFORMACIÓN DE LA PRESIÓN
917 AL 964	INFORMACIÓN DE LA GRAVEDAD ESPECIFICA
965 AL 1011	INFORMACIÓN DEL TOTAL DE BARRILES BRUTOS
1012 AL 1059	INFORMACIÓN DEL TOTAL DE BARRILES NETOS
1060	00 H ASCII

### Command

Quando el operador introduce un comando, Intouch genera un archivo de texto y el driver de comunicación se encarga de transmitir la cadena de información contenida en este archivo al controlador Daniel MM3000.

La respuesta del controlador Daniel MM3000 al comando solicitado es recibida por el driver de comunicación, el cual se encarga de separar el 01H y 37H ASCII de la cadena del resto de la respuesta y grabarla en un archivo de texto. Debido a que la respuesta se considera como un sólo dato no necesita tener formato CSV para poder ser leída por Intouch y presentarla al operador.

El formato típico de la cadena de respuesta es el siguiente:

BUF1	SIGNIFICADO
1	01 H ASCII ACCESO AL DANIEL MM3000
2	37 H ASCII SOLICITUD COMMAND
3 AL 39	RESPUESTA AL COMANDO
40	0D H ASCII

### 3.3.2 PROGRAMACIÓN DEL DRIVER DE COMUNICACIÓN.

A continuación se presentan los programas que permiten la comunicación entre el Daniel MM3000 y la interfaz de operación Intouch, estos están realizados en lenguaje "C++" y compilados en ambiente Windows para lograr una respuesta más rápida a las solicitudes y a los comandos.

En ellos se utilizan librerías de "C++" y librerías propias de la tarjeta multipuertos PC-LAB y se muestran en el Apéndice B en el mismo orden en que aparecen las pantallas de la interfaz.

### 3.4 OPERACIÓN DEL NUEVO SISTEMA DE MONITOREO Y CONTROL.

Un equipo de cómputo y el software de interfaz hombre-máquina Intouch versión 5.0 de Wonderware constituyen la interfaz principal entre el operador y el sistema de medición y control.

Las operaciones de carga de buques-tanque y pruebas de medidores son completamente automáticas. El control es desarrollado por los controladores de estación Daniel MM3000 actuando en respuesta a los comandos que le son transmitidos por medio del teclado de la consola de operación (PC HP-Vectra).

Los controladores imprimen reportes de entrega (Delivery Tickets) pero no imprimen reportes de pruebas y alarmas, estas funciones son asumidas por la consola de operación.

Cada estación de carga muelle/boya tiene un gráfico que muestra los estados actuales de las válvulas (abierto/cerrado), flujos instantáneos, temperaturas y presiones de cada subsistema, así como, todas las líneas de medición de la estación y el probador asociado a ellas. Un desplegado de lote (batch), un desplegado de prueba de medidor

(proof report) y un desplegado para las alarmas que se presenten en el sistema son proporcionados para cada estación.

El operador configura lotes de entrega, modifica parámetros y realiza pruebas sobre las líneas de medición introduciendo comandos a través del teclado de la consola de operación. Un campo de diálogo en el monitor de la consola de operación está reservado para este propósito, la introducción de estos comandos se realiza de forma interactiva (pregunta- respuesta).

Se cuenta con una impresora donde se imprimen reportes de entrega (Delivery Tickets), reportes de pruebas (Proof Reports) y reportes de alarma.

Los controladores Daniel MM3000 son las unidades de control, la consola de operación exclusivamente transmite parámetros a ellos y monitorea el progreso de las operaciones de entrega y de pruebas de medidor.

Las válvulas son controladas automáticamente y las líneas de medición son abiertas o cerradas por el controlador Daniel MM3000 como sea requerido, con el fin de conseguir la cantidad exacta de crudo en una entrega determinada. Los 2233's desarrollan funciones de control de flujo sobre líneas individuales de medición, usando puntos de ajuste para el flujo instantáneo, mismos que son recibidos desde el controlador Daniel MM3000.

Las pruebas de líneas de medición se desarrollan automáticamente a través del controlador Daniel MM3000, pero dichas operaciones son iniciadas por comandos transmitidos desde la consola de operación.

#### **COMANDOS DE LA CONSOLA DE OPERACIÓN.**

Un campo sobre la pantalla de video de la consola de operación está reservado para los diálogos (pregunta-respuesta), a través del cual el operador introduce parámetros y controla operaciones de prueba y de entrega de lote, la entrada del parámetro es permitido solamente si la llave de modo manual/controlador/computadora del tablero de control se encuentra en la posición de modo computadora. Para acceder este campo, se dispone de un botón (command) en la pantalla principal.



Los comandos de entrada permitidos en la consola de operación son:

- 1) Abrir o cerrar una línea de medición.
- 2) Poner una línea de medición en o fuera de servicio.
- 3) Prueba de un medidor.
- 4) Prueba de todos los medidores de flujo de una estación.
- 5) Abortar una prueba.
- 6) Introducción de un factor de medición a un medidor.
- 7) Introducción de un valor para el volumen del probador.
- 8) Introducción del número máximo de ensayos de prueba.
- 9) Introducción de la tolerancia de repetibilidad de prueba.
- 10) Restablecimiento de los 2233's y del 3000's de una estación.
- 11) Introducción de la razón de muestreo de una estación.
- 12) Introducción del punto de ajuste del gasto de una estación.
- 13) Configuración e Inicialización de un lote desde una estación.
- 14) Cambio de parámetros del lote de una estación.
- 15) Ir por la razón de carga.
- 16) Ir por la razón topoff.
- 17) Suspensión de entrega.
- 18) Concluir un lote.
- 19) Seleccionar la línea de medición de reserva de una estación.
- 20) Introducción de los límites de alarma alto y bajo para un valor analógico.
- 21) Introducción de un valor de sobreposición para una entrada analógica.
- 22) Solicitud de despliegue del lote (batch) de una estación.
- 23) Solicitud de despliegue de un gráfico de una estación.
- 24) Solicitud de despliegue de la prueba de una línea de medición.
- 25) Solicitud de despliegue de alarmas.

Los parámetros necesarios para la configuración de un lote son los siguientes:

- 1) Nombre del barco.
- 2) Número del embarque.
- 3) Volumen del lote.
- 4) Volumen topoff.
- 5) Razón de arranque.
- 6) Razón de carga.
- 7) Razón topoff.

Los comandos introducidos a través de la consola de operación son requeridos para iniciar un lote y conseguir la razón de carga. Algunos comandos opcionales son proporcionados para llevar el lote a la razón topoff y para suspender el lote. Normalmente, estas dos últimas funciones son desarrolladas por el controlador y no requieren que el operador ejecute algún comando en particular para ordenar su ejecución.

#### **PROCEDIMIENTO DE CARGA DE UN LOTE (BATCH).**

La consola de operación transmite los comandos ordenados por el operador hacia los controladores Daniel MM3000, este efectúa las operaciones de entrega de lotes y pruebas de medidores.

El controlador Daniel MM3000 abre y cierra las líneas de medición y asigna puntos de ajuste al flujo instantáneo automáticamente, de acuerdo con los parámetros del lote transmitidos a él durante el tiempo que se realiza la configuración del lote.

El operador configura la entrega y la inicia a través de la consola de operación introduciendo diferentes comandos. Cuando el comando de Inicio es introducido la consola transmite los datos de la configuración del lote hacia el controlador Daniel MM3000 para su ejecución, éste empieza a abrir los medidores que son requeridos y consigue la razón de arranque (startup rate), ésta es mantenida hasta que otro comando sea introducido para pasar posteriormente a otra etapa. Cuando esta razón está lista, el operador ordena al controlador Daniel MM3000 que inicie la rampa de subida hasta conseguir la razón de carga introducida por el operador. Este comando es transmitido al controlador Daniel MM3000 para su ejecución, en un ciclo de procesamiento, el controlador Daniel MM3000 determina cuantos medidores son requeridos de acuerdo al punto de ajuste del flujo instantáneo actual de la estación. Para un flujo abajo de 8000 bph un medidor es abierto, arriba de 7999 bph dos medidores son abiertos, arriba de 15999 bph tres medidores son abiertos, arriba de 23999 bph cuatro medidores son abiertos. Los medidores son abiertos uno a la vez y el punto de ajuste del flujo instantáneo de la estación es distribuido eventualmente entre los medidores abiertos, sin embargo, a ningún medidor se le asigna un punto de ajuste menor a 1130 bph o mayor a 11400 bph.

Cuando la razón de carga ha sido conseguida, a menos que el operador intervenga, el flujo continua en esa razón hasta que el volumen "topoff" de flujo ha sido alcanzado. En este punto la razón topoff automáticamente se convierte en el punto de

ajuste del flujo instantáneo de la estación y los medidores se cerraran como sea necesario para iniciar la disminución de la razón al nuevo punto de ajuste.

El flujo permanece en la razón "topoff" hasta que el volumen del lote ha sido alcanzado, en este punto los medidores de flujo restantes son cerrados uno a la vez. El operador termina una entrega a través de un comando que indica la conclusión. El comando de conclusión es transmitido al controlador Daniel MM3000 el cual finaliza la entrega. También transmite los datos a la consola de operación en donde si el operador lo desea se imprime un reporte de la entrega en la impresora conectada a ella.

En cualquier instante durante una entrega, el operador puede introducir un comando en la consola para suspender el flujo. La consola transmite el comando al controlador Daniel MM3000 para suspender el flujo, el cual cierra las líneas de medición, una a la vez. Después de que todas las líneas han sido cerradas, la entrega puede ser iniciada de nuevo o esta puede ser concluida. Si es necesario incrementar el tamaño de un lote (batch) después de que el volumen del lote se ha conseguido, el operador puede entonces cambiar los parámetros del lote desde el teclado de la consola y reiniciar el lote. Los parámetros del lote pueden también ser cambiados en cualquier tiempo mientras la entrega este en progreso.

El punto de ajuste del flujo instantáneo es establecido por el controlador Daniel MM3000 de acuerdo con los parámetros del lote enviados a él desde la consola de operación. Sin embargo, el punto de ajuste actual puede ser sobrepuesto por el operador en cualquier tiempo durante la entrega a través de comandos transmitidos desde la consola. Si el punto de ajuste es cambiado mientras la entrega esta en la razón de carga o en la razón topoff, el correspondiente parámetro de lote también será cambiado. El control de flujo actual en este caso es realizado por los 2233's y no por el controlador Daniel MM3000.

La razón de muestreo de la estación es establecida por el controlador Daniel MM3000 cuando una entrega es iniciada. Esta es obtenida de dividir el volumen del lote entre 6300, el resultado es el número de muestras al cual se llenara el contenedor. Después del arranque, la razón puede ser cambiada por el operador introduciendo un comando en la consola. Si la entrega es suspendida y posteriormente reiniciada, no se calcula una nueva razón de muestreo.

### **PRUEBA DE MEDIDOR.**

Existen dos comandos diferentes para comenzar las pruebas de los medidores, uno origina que un solo medidor sea probado, mientras que el otro comando ordena probar todos los medidores de la estación. En ambos casos la consola simplemente transmite los comandos al controlador Daniel MM3000. Este inicia y controla todas las operaciones de la prueba y transmite los datos de la misma hacia la consola, por solicitud del operador.

Los datos son los mismos que se muestran en el desplegado del probador después de que la prueba ha sido completada. El controlador pone a disposición de la computadora los datos de la prueba para imprimir el reporte.

### **SELECCIÓN DE UN MEDIDOR DE RESERVA (SPARE).**

Un comando de entrada introducido por medio del teclado de la consola de operación indica que un medidor de reserva se asignara a una de las estaciones de carga muelle/boya. La selección es transmitida al controlador Daniel MM3000 apropiado.

El nuevo sistema de supervisión de la Terminal Maritima Pajaritos atiende las siguientes solicitudes: Graphic Display, Proof Report, Prover Display, Batch Display, Alarm Report, Delivery Ticket y Commands.

### **GRAPHIC DISPLAY.**

Para operar la solicitud "Graphic Display" no importando el gráfico que esté presente en pantalla es necesario realizar las siguientes acciones para que dicha opción sea desplegada:

- 1) Seleccionar la estación de carga en la pantalla principal, haciendo un click con el mouse sobre el botón correspondiente.
- 2) Seleccionar el botón "Graph" con el mouse y automáticamente es desplegada en pantalla la información correspondiente a la estación seleccionada.

**PROOF REPORT.**

Para operar la solicitud "Proof Report" no importando el gráfico que esté presente en pantalla, es necesario realizar las siguientes acciones para que esta solicitud sea atendida:

- 1) Seleccionar la estación de carga haciendo un click con el mouse sobre el botón correspondiente.
- 2) Seleccionar el botón "Proof Report" con el mouse y automáticamente es desplegada en pantalla la información correspondiente a la estación seleccionada.

Si se desea imprimir un reporte de esta solicitud se deberá dar un click con el mouse sobre la tecla "imprimir" desplegada a la derecha de este gráfico.

El reporte de prueba (Proof Report) es desplegado en pantalla o impreso a petición del usuario.

**PROVER DISPLAY**

Para operar la solicitud "Prover Display" no importando el gráfico que esté presente en pantalla, es necesario realizar las siguientes acciones para que esta solicitud sea atendida:

- 1) Seleccionar la estación de carga haciendo un click con el mouse sobre el botón correspondiente.
- 2) Seleccionar el botón "Prover Display" con el mouse y automáticamente es desplegada en pantalla la información correspondiente a la estación seleccionada.

**BATCH DISPLAY**

Para operar la solicitud "Batch Display" no importando el gráfico que esté presente en pantalla, es necesario realizar las siguientes acciones para que esta solicitud sea atendida:

- 1) Seleccionar la estación deseada haciendo un click con el mouse sobre el botón correspondiente.
- 2) Seleccionar el botón "Batch Display" con el mouse y automáticamente es desplegada en pantalla la información correspondiente a la estación seleccionada.

**ALARM REPORT.**

Para operar la solicitud "Alarm Report" no importando el gráfico que esté presente en pantalla, es necesario realizar la siguiente acción para que esta solicitud sea atendida:

- 1) Seleccionar el botón "Alarm Report" con el mouse y automáticamente es desplegada en pantalla la información correspondiente a todas estaciones.

Si se desea imprimir un reporte del estado y de las alarmas de todas las estaciones de carga (muelle/bolla) se deberá dar un click con el mouse sobre la tecla "imprimir" desplegada en la parte inferior de este gráfico.

A continuación se listan todos los mensajes y estados posibles de alarma, que son desplegados por una solicitud de "Alarm Report":

valve failed to open
valve failed to close
high flow rate
low flow rate
setpoint unattainable
high strainer pressure
2230 comm failure
2230 gross deviation
pulse comparison failure
meter out of service
meter high temperature
meter low temperature
inlet hdr high pressure
inlet hdr low pressure
Prover high temperature

Prover low temperature
Prover high pressure
Prover low pressure
prf abtd, operator
prf abtd, interlocks
prf abtd, valve failed
prf abtd, temperatures
prf abtd, detector 1
prf abtd, detector 2
prf abtd, counter
prf abtd, launch fail
prf abtd, manual mode
prf abtd, suspended
prf abtd, dp seal
high viscosity
low viscosity
high BS&W
low BS&W
outlet hdr high pressure
outlet hdr low pressure
high specific gravity
low specific gravity
delivery starting
going to load rate
delivery topping off
delivery suspended
batch volume reached
controller failure
station in computer mode
station in controller mode
station in manual mode
batch status
prove status
status meter spare

**DELIVERY TICKET.**

Para operar la solicitud "Delivery Ticket" no importando el gráfico que esté presente en pantalla, es necesario realizar las siguientes acciones para que esta solicitud sea atendida:

- 1) Seleccionar la estación de carga haciendo un click con el mouse sobre el botón correspondiente.
- 2) Seleccionar el botón "Delivery Ticket" con el mouse y automáticamente es desplegada en pantalla la información correspondiente a la estación seleccionada.

El reporte de entrega es desplegado en pantalla o impreso a petición del usuario.

### COMMAND

El usuario selecciona esta opción cuando requiera introducir parámetros y controlar operaciones de prueba y carga de lotes.

Para operar la solicitud "Command" no importando el gráfico que esté presente en pantalla, es necesario realizar las siguientes acciones para que esta solicitud sea atendida:

- 1) Seleccionar el botón "Command" haciendo un click con el mouse y automáticamente es desplegado en pantalla el gráfico correspondiente a "Command".
- 2) Con el mouse hacer un click sobre el símbolo (>) localizado junto a la leyenda "comando" o en su defecto basta oprimir la barra espaciadora del teclado de la computadora (ambos desplegados en este gráfico), para habilitar el campo de comandos de entrada al sistema.
- 3) Una vez que se ha escrito el comando en el campo reservado para tal fin, se oprime la tecla "enter" del teclado de la computadora para transmitir dicho comando hacia el controlador de estación Daniel MM3000 correspondiente, automáticamente la respuesta enviada por el controlador es desplegada en el campo reservado para tal fin en la pantalla de la consola de operación.
- 4) Si es requerido por el tipo de comando establecer un diálogo entre el usuario y el sistema, cuando éste último solicite un dato o alguna confirmación, el usuario presionará la barra espaciadora del teclado de la computadora para habilitar el campo reservado para la introducción de



comandos, y entonces escribirá el dato o la confirmación solicitada por el sistema.

Es importante señalar que cuando se escriba un comando éste necesariamente tendrá que ir acompañado de un número de estación de carga, de otro modo el sistema no enviara respuesta alguna hacia la consola de operación.

Cuando un comando mal escrito es transmitido hacia el controlador Daniel MM3000 se despliega la leyenda "entry invalid" (entrada invalida) en el campo reservado para la respuesta del sistema.

Los comandos de entrada hacia el sistema son permitidos exclusivamente si la llave de modo de operación manual / controlador / computadora, localizada en el tablero de control, se encuentra en la posición "modo computadora."

A continuación se listan los comandos que pueden ser utilizados en el sistema:

comando del teclado	función del comando
open 7e-1	abrir el medidor 1 de la estación 7e.
close 7e-1	cerrar el medidor 1 de la estación 7e.
enable 7e-1	colocar el medidor 1 de la estación 7e en servicio.
disable 7e-1	colocar el medidor 1 de la estación 7e fuera de servicio.
prove 7e	probar todos los medidores de flujo de la estación 7e.
prove 7e-1	probar el medidor 1 de la estación 7e.
abort	aborta la prueba actual del medidor.
max trials	despliega el número máximo de ensayos de prueba permitidos.
max trials nn	introducir el valor nn para el número máximo de ensayos de prueba (1-20).
repeat %	despliega la tolerancia de repetibilidad de la prueba.
repeat % XX	introducir el valor .XX para la tolerancia de repetibilidad de la prueba (.02-.05).
bv	despliega el volumen base del probador.
bv xx.xxxxx	introducir el valor xx.xxxxx para el volumen base del probador (41.0-43.0).
reset 7e	restablecer los contadores de la estación 7e.
setup 7e	configurar el lote para la estación 7e.
change 7e	cambiar los datos de configuración de lote de la estación 7e.
start 7e	arranque del lote de la estación 7e.
load 7e	orden para ir por la razón de carga de la estación 7e.

CAPITULO 3. DESARROLLO DEL SOFTWARE

<b>topoff 7e</b>	orden para ir por la razón topoff de la estación 7e.
<b>suspend 7e</b>	suspende el flujo en la estación 7e.
<b>terminate 7e</b>	obtiene los totales de la estación 7e e imprime el reporte de la entrega
<b>spare 7e</b>	asigna el medidor de reserva a la estación 7e.
<b>sample 7e</b>	despliega la razón de muestreo para la estación 7e.
<b>sample 7e nnnnn</b>	introducir la razón de muestreo de nnnnn bbls para la estación 7e.
<b>setpt 7e</b>	despliega el punto de ajuste del gasto de la estación 7e.
<b>setpt 7e nnnnn</b>	introducir el punto de ajuste de gasto nnnnn bph para la estación 7e.
<b>setpt 7e-1</b>	despliega el punto de ajuste del gasto para el medidor 1 de la estación 7e.
<b>mf-7e</b>	despliega el factor de medición actual para el medidor 1 de la estación 7e.
<b>mf-7e xxx.xxxx</b>	introducir el nuevo factor de medición de xxx.xxxx para el medidor de la estación 7e.
<b>kgcm2 7e</b>	despliega la presión del cabezal de entrada de la estación 7e.
<b>kgcm2 7eov</b>	despliega el valor de sobreposición de la presión del cabezal de entrada (-1= ninguno).
<b>kgcm2 7e ovxx.x</b>	introducir el valor de sobreposición de xx.x (0-21) para la presión del cabezal de entrada, o introducir -1.0 para cancelar un valor de sobreposición.
<b>kgcm2 7e hi</b>	despliega el límite de alarma de la presión de entrada de la estación 7e.
<b>kgcm2 7e hixx.x</b>	introducir el valor xx.x como el límite de alarma alto de la presión de entrada de la estación 7e.
<b>kgcm2 7e lo</b>	despliega el límite de alarma bajo de presión de entrada de la estación 7e.
<b>kgcm2 loxx.x</b>	introducir el valor xx.x como el límite de alarma bajo de la presión de entrada de la estación 7e.
<b>okgcm2 7e</b>	despliega la presión del cabezal de salida.
<b>okgcm2 7e ov</b>	despliega el valor de sobreposición de la presión de salida de la estación 7e (-1 = ninguno)
<b>okgcm2 7e ovxx.x</b>	Introducir un valor de sobreposición de xx.x (0-21) para la presión del cabezal de salida de la estación 7e, o Introducir -1.0 para cancelar el valor de sobreposición.
<b>okgcm2 7e hi</b>	despliega el límite de alarma alta de la presión de salida para la estación 7e.
<b>okgcm2 7e hixx.x</b>	Introducir el valor xx.x como el límite de alarma alta de la presión de salida de la estación 7e.

okgcm2 7e lo	despliega el límite de alarma baja de la presión de salida de la estación 7e.
okgcm2 7e loxx.x	Introducir el valor xx.x como límite de alarma baja de la presión de salida de la estación 7e.
pkgcm2	despliega la presión del probador.
pkgcm2 ov	despliega el valor de sobreposición de la presión del probador (-1.0 = ninguno).
pkgcm2 ovxx.x	Introducir un valor de sobreposición de xx.x (0-21) para la presión del probador, o introducir -1.0 para cancelar el valor de sobreposición.
pkgcm2 hi	despliega el valor xx.x como el límite de alarma de alta de la presión del probador.
pkgcm2 hixx.x	Introducir el valor xx.x como el límite de alarma alta de la presión del probador.
pkgcm2 lo	despliega el límite de alarma baja de la presión del probador.
pkgcm2 loxx.x	Introducir el valor xx.x como límite de alarma baja de la presión del probador.
pdegc	despliega la temperatura del probador.
pdegc ov	despliega el valor de sobreposición de la temperatura del probador (-1 = ninguno).
pdegc ovxx.x	Introducir el valor de sobreposición de xx.x (0-100 para la temperatura del probador, o introducir -1.0 para cancelar el valor de sobreposición.
pdegc hi	despliega el límite de alarma alta de la temperatura del probador.
pdegc hixx.x	Introducir el valor xx.x como límite de alarma alta de la temperatura del probador.
pdegc lo	despliega el límite de alarma baja de la temperatura del probador.
pdegc loxx.x	Introducir el valor xx.x como límite de alarma baja de la temperatura del probador.
visc 7e	despliega la viscosidad del producto.
visc 7e ov	despliega el valor de sobreposición de la viscosidad (-1 = ninguno).
visc 7e ovxxx.x	Introducir un valor de sobreposición de la viscosidad de xx.x (20-500), o introducir -1.0 para cancelar un valor de sobreposición.
visc 7e hi	despliega el límite de alarma alta de la viscosidad.
visc 7e hixxx.x	Introducir el valor xx.x como límite de alarma alta de la viscosidad.
visc 7e lo	despliega el límite de alarma baja de la viscosidad.

visc 7e loxxx.x	introducir el valor xx.x como límite de alarma baja de la viscosidad.
bsw 7e	despliega el contenido BS&W del producto.
bsw 7e ov	despliega el valor de sobreposición del BS&W (-1 = ninguno).
bsw 7e ovxxx.x	introducir un valor de sobreposición para el BS&W de xx.x, o introducir -1.0 para cancelar el valor de sobreposición.
bsw 7e hl	despliega el límite de alarma alta del BS&W.
bsw 7e hixxx.x	introducir el valor xx.x como el límite de alarma alto del BS&W.
bsw 7e lo	despliega el límite de alarma bajo del BS&W.
bsw /e loxxx.x	introducir el valor xx.x como límite de alarma bajo del BS&W.
sg 7e	despliega la gravedad específica del producto.
sg 7e ov	despliega el valor de sobreposición de la gravedad específica (-1 = ninguno).
sg 7e ovx.x	introducir un valor de sobreposición para la gravedad específica de x.x (0.7- 1.0), o introducir -1.0 para cancelar el valor de sobreposición.
sg 7e hl	despliega el límite de alarma alta de la gravedad específica.
sg 7e hix.x	introducir el valor x.x como el límite de alarma alta de la gravedad específica.
sg 7e lo	despliega el límite de alarma baja de la gravedad específica.
sg 7e lox.x	introducir el valor x.x como el límite de alarma baja de la gravedad específica.
degc 7e-1	despliega la temperatura del medidor 1 de la estación 7e.
degc 7e-1 ov	despliega el valor de sobreposición de la temperatura del medidor 1 en la estación 7e (-1 = ninguno)
degc 7e-1 ovxx.x	introducir un valor de sobreposición de temperatura de xx.x (0-100) para la temperatura del medidor 1 en la estación 7e, o introducir -1.0 para cancelar un valor de sobreposición.
degc 7e-1 hl	despliega el límite de alarma alta de temperatura del medidor 1 en la estación 7e.
degc 7e-1 hixx.x	introducir el valor xx.x como límite de alarma alta de la temperatura del medidor 1 en la estación 7e.
degc 7e-1 lo	despliega el límite de alarma baja de la temperatura del medidor 1 en la estación 7e.

Estos comandos son similares para la estaciones 3E, 3W, 4E, 4W, 7W, 9E, 9W, B1 y B2

Adicional a las botoneras desplegadas en la pantalla principal, destinadas para la selección de las diferentes estaciones de carga muelle/boya y solicitudes de usuario, el usuario puede operar las teclas de números y de función dedicadas (teclado de la

Adicional a las botoneras desplegadas en la pantalla principal, destinadas para la selección de las diferentes estaciones de carga muelle/boya y solicitudes de usuario, el usuario puede operar las teclas de números y de función dedicadas (teclado de la consola de operación), para seleccionar directamente dichas estaciones y solicitudes, con la siguiente correspondencia.

TECLA NUMÉRICA	MUELLE/ BOYA
1	B1
2	B2
3	3W
4	3E
5	4W
6	4E
7	7W
8	7E
9	9W
0	9E
**CTRL + P	IMPRIMIR
TECLA DE FUNCIÓN	SOLICITUD
F1	GRAPHIC DISPLAY
F2	PROOF REPORT
F3	PROVER DISPLAY
F4	BATCH DISPLAY
F5	ALARM REPORT
F6	DELIVERY TICKET
F7	COMMAND

**\*\* NOTA:** La secuencia de teclas Ctrl + P permite al usuario enviar a imprimir un Proof Report, Alarm Report o un Delivery Ticket.

Es importante señalar que actualmente el embarque de petróleo crudo en la Terminal Marítima Pajaritos, es realizado en las estaciones de carga (muelles): 4e, 4w, 3e, 3w (m-6), 7e y 7w. El muelle 9w es utilizado para recibir y medir el petróleo crudo enviado desde la estación de bombeo de Nuevo Teapa, Ver., mientras que las estaciones de carga 9e, b-1 y b-2 actualmente se encuentran fuera de servicio.

Por lo anterior el nuevo sistema de supervisión desarrollado para la Terminal Marítima Pajaritos, quedó configurado para supervisar las operaciones de carga y realizar operaciones de prueba de medidor de las estaciones de carga( muelles) 4e, 4w,

3e, 3w (m-6), 7e y 7w. El muelle 9w quedó exclusivamente configurado para realizar la supervisión del petróleo crudo recibido de la estación de bombeo Nuevo Teapa, Ver.

Cabe mencionar que las solicitudes mencionadas anteriormente son atendidas exclusivamente a petición del usuario, de otro modo el sistema de supervisión no despliega ningún tipo de información.

Por otro lado, la información presentada en pantalla para cada una de las anteriores solicitudes es constantemente actualizada, ésto se hace cada tres segundos. Es importante señalar que el nuevo sistema de supervisión sólo puede atender una solicitud a la vez.

## CAPITULO 4. REQUERIMIENTOS DE HARDWARE Y SOFTWARE

Los requerimientos de hardware y software que se presentan a continuación son los mínimos que el sistema necesita para funcionar adecuadamente en cuanto a velocidad de respuesta y calidad en el despliegue de gráficos.

Primeramente se presenta el diagrama general del sistema y posteriormente se enuncian los elementos de hardware y software que componen al nuevo sistema de monitoreo y control.

### 4.1 DIAGRAMA GENERAL DEL SISTEMA

En el diagrama 4.1 se muestra la estructura del nuevo sistema de monitoreo y control. En la parte superior del diagrama se muestran los controladores Daniel MM3000 los cuales están conectados al patrón de medición que les corresponde.

Los Daniel MM3000 se conectan a la computadora supervisora, la cual se encarga de distribuir la información ya procesada al monitor o a la impresora si el operador así lo ordena desde el teclado de la computadora supervisora.

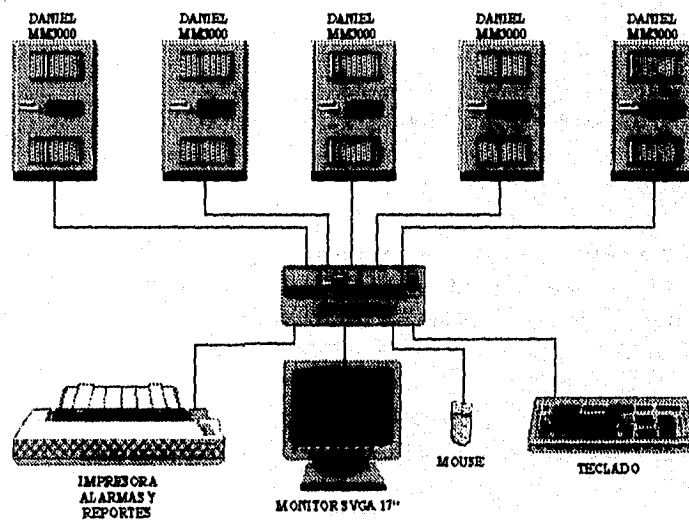


FIGURA 4.1 DIAGRAMA GENERAL DEL SISTEMA

## 4.2 SOFTWARE

El Software es una serie de instrucciones de computadora que realizan una tarea en particular, a esta tarea se le llama programa de software. Existen dos categorías principales que son: *software de sistemas* y *software de aplicaciones*.

El *Software de Sistemas* se compone de programas de control, incluyendo el sistema operativo, software de comunicaciones y administrador de base de datos.

El *Software de Aplicaciones* es cualquier programa que procesa datos para el usuario.

El sistema de supervisión que sustituye a la computadora Hewlett-Packard A600, esta constituido por:

### Software de Sistema:

- Sistema operativo: MS-DOS versión 6.2 de Microsoft.
- Windows versión 3.1 o Windows for Work Groups versión 3.11 de Microsoft.

### Software de Aplicación:

- Interfaz de Operación: Intouch de Wonderware versión 5.0.

## 4.3 HARDWARE

El Hardware es el mundo del almacenamiento y la transmisión de datos. Cuánta más memoria y almacenamiento en disco tiene un sistema informático, más trabajo puede hacer. Cuánto más rápidos sean la memoria y los discos, para transmitir datos e instrucciones entre elio y el CPU, más rápido se hará el trabajo.

Los requerimientos de Hardware se basan en el tamaño de los archivos y las bases de datos que serán creadas y el número de usuarios simultáneos en las terminales.



En base a las especificaciones dadas por el fabricante de la interfaz de operación se determino que los componentes mínimos del sistema de supervisión que sustituye a la computadora Hewlett-Packard A600, son los siguientes:

- IBM PC o Compatible
- Procesador 80486 DX a 33MHz.
- 8 Megabytes de memoria RAM.
- Un disco duro de 120 megabytes.
- Una unidad de disco flexible de 3.5" de 1.44 megabytes.
- Una tarjeta inteligente PC-LAB CARD PCLS-802 de 8 puertos seriales marca Advantech.
- Un monitor VGA a color.
- Un puerto paralelo.
- Un puertos serie.
- Un mouse.
- Una impresora de matriz de puntos con resolución de 600 DPI, 12 CPI, 8 PPM.
- Llave de software de aplicación (RUNTIME).

En base al volumen de datos que maneja el sistema y a la configuración realizada en la interfaz de operación se recomienda un equipo de computo con las siguientes características:

- IBM PC o Compatible
- Procesador 80486 DX/2 a 66MHz.
- 16 Megabytes de memoria RAM.
- Un disco duro de 540 megabytes.
- Una unidad de disco flexible de 3.5" de 1.44 megabytes.
- Una tarjeta inteligente PC-LAB CARD PCLS-802 de 8 puertos seriales marca Advantech.
- Un monitor ultra VGA de 21" a color.
- Un puerto paralelo.
- Dos puertos serie.
- Un mouse.
- Una impresora de inyección de tinta con una resolución de 600 x 300 DPI en negro y 300 x 300 DPI en color, con opción de impresión en blanco y negro y 16 millones de colores.



- Llave de software de aplicación (RUNTIME).

Debido a restricciones en el presupuesto, Petroleos Mexicanos adquirió el equipo de cómputo que se describe a continuación:

- Una PC HP Vectra.
- Procesador 80486 DX/2 a 66MHz.
- 8 Megabytes de memoria RAM.
- Un disco duro de 320 megabytes.
- Una unidad de disco flexible de 3.5" de 1.44 megabytes.
- Una tarjeta inteligente PC-LAB CARD PCLS-802 de 8 puertos seriales marca Advantech.
- Un monitor ultra VGA de 17" a color.
- Un puerto paralelo.
- Dos puertos serie.
- Un mouse.
- Una impresora de matriz de puntos con resolución de 600 DPI, 12 CPI, 8 PPM.
- Llave de software de aplicación (RUNTIME).

#### 4.4 DIAGRAMA DE CONEXIONES

La comunicación entre cada uno de los Daniel MM3000 y la computadora supervisora se realiza por medio de un cable serial que cumple con la norma RS-232-C. Esta es una norma de la EIA (Electronic Industries Association) para transmisiones en serie que utiliza conectores DB-25, de 25 "pins"(clavijas), o DB-9, de 9 pins. Su limitación normal de cable de 50 pies (15 m) puede extenderse a varios cientos de pies mediante un cable de alta calidad.

La norma RS-232 define los propósitos, características eléctricas y temporización de las señales en el cable. Sin embargo, no siempre se usan los 25 conectores del cable; muchas aplicaciones utilizan menos de una docena.

Para esta aplicación se emplean conectores DB-25 de los cuales se utilizan los pines de recepción, transmisión y tierra únicamente.

En la figura 2.2 se muestra el diagrama de conexión entre la computadora supervisora y los controladores Daniel MM3000.

La tarjeta multipuertos esta preconfigurada para referirse a sus puertos como 10, 11, 12, 13 y 14 respectivamente, éstos son conectados a los controladores Daniel MM3000 de acuerdo a la siguiente tabla:

PUERTO	DANIEL MM3000
10	B1-B2
11	3E-3W
12	4E-4W
13	7E-7W
14	9E-9W

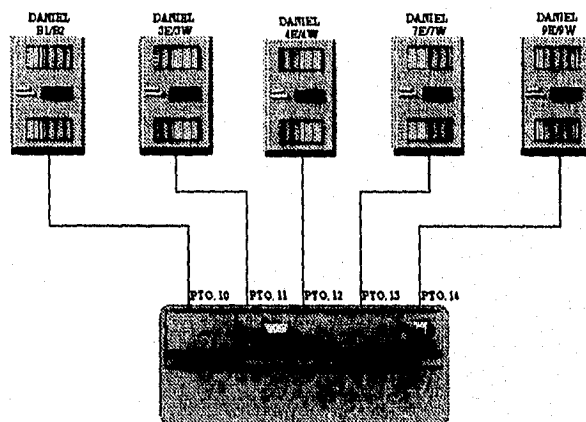


FIGURA 2.2 DIAGRAMA DE CONEXION

Los requerimientos de hardware y diagramas de conexión pueden cambiar en caso de que la aplicación sea modificada, se incrementen las unidades Daniel MM3000 que se monitorean o se habilite el monitoreo histórico de las variables o alarmas.

Una vez que se conocen y se cubren los requerimientos de software y de hardware, así como la manera de conectar la computadora con los Daniel MM3000, es posible comenzar la instalación del sistema.

---

## CAPITULO 5. INSTALACIÓN, ARRANQUE Y MANTENIMIENTO DEL SISTEMA.

La instalación de un nuevo sistema consiste en preparar a un programa o programas en un paquete de Software para su ejecución en la computadora. Se copian los archivos de los disquetes de distribución al disco rígido y, si es necesario, también se descomprimen. En algunas ocasiones se solicita al usuario que identifique los tipos de dispositivos periféricos que están conectados para poder enlazar los controladores apropiados para una determinada presentación por pantalla, impresora, explorador, etc.

EL arranque de cualquier sistema consiste en una rutina que se ejecuta al ser inicializada la computadora o cuando se carga un programa para el software asociado a él.

El mantenimiento de programas o software es la actualización de los programas de aplicación para satisfacer los cambiantes requerimientos de información. El mantenimiento de discos o archivos es la reorganización periódica de los archivos en discos duros que han sufrido fragmentaciones debido a la actualización continua.

A continuación, se darán los pasos a seguir para la instalación del nuevo sistema, el proceso a seguir para el arranque del mismo, así como su mantenimiento.

### 5.1 INSTALACIÓN.

El sistema de supervisión de la terminal marítima Pajaritos Ver., requiere para sus instalación de los siguientes paquetes de software:

- Sistema Operativo MS-DOS versión 6.2 de Microsoft.
- Windows Versión 3.1 o Windows for Groups Versión 3.11 de Microsoft.
- Interfaz de operación Intouch Versión 5.0 de Wonderware.

Para la instalación del sistema de supervisión se deben seguir los siguientes pasos:

- 1) Instalar el sistema operativo MS-DOS V. 6.2 en su propio directorio, es decir, en el directorio DOS, esto se realizará si el sistema operativo no está instalado, de otra manera no será necesario volverlo a instalar. Deberá

añadir una línea al archivo CONFIG.SYS para crear una unidad de disco ram en el sistema, la instrucción es la siguiente:

```
device = ramdrive.sys [1024] [/e]
```

- 2) Instalar Windows V. 3.1 o Windows for Groups V. 3.11 en su propio directorio, es decir, en el directorio de Windows, esto se realizará si Windows no está instalado, de otra forma no será necesario volverlo a instalar.
- 3) Instalar el Windowviewer o Runtime de la interfaz de Operación Intouch Ver. 5.0 de Wonderware. Para realizar este paso se deberá utilizar los discos de instalación de Intouch, al estar instalando el paquete se presentará la alternativa de instalar el Runtime o el WindowViewer debiéndose seleccionar la opción de instalar el Runtime.
- 4) Instalación del software de la tarjeta multipuertos serial. Hacer una copia del disco que contiene el software de la tarjeta al disco duro. Modificar el archivo AUTOEXEC.BAT aumentando la siguiente línea:

```
PATH C:\PCL-844
```

- 5) Por último se instala el sistema de supervisión de la Terminal Marítima Pajaritos, Ver.

El sistema de supervisión consta de cuatro discos, que son:

- a) Disco No. 1., que contiene todos los archivos con extensión .EXE.
- b) Disco No. 2., que contiene el sistema de supervisión de forma compactada, este disco contendrá todos los tags (variables) y Scripts (Programas realizados en Intouch), esta información estará contenida en un archivo DAN3000 con extensión .LZH.
- c) Discos 3 y 4 que contienen todos los archivos con extensión .DEF, .OBJ, .PRJ, .C y .CPP, los cuales permiten generar los archivos .EXE que son los programas de la aplicación del sistema de supervisión.

Para instalar el nuevo sistema de supervisión será necesario realizar las siguientes acciones:

- I) Copiar todo el contenido del disco No. 1 en el directorio de Windows. En este directorio se copiarán todos los archivos con extensión .EXE.
- II) Crear un directorio con el nombre de DAN3000. Descompactar el contenido del disco No. 2, cuya información será copiada en el directorio DAN3000, esta operación se realizará ejecutando el siguiente comando:

C:\LHA X A:\DAN3000.LZH C:\DAN3000

- III) Los discos No. 3 y 4 sólo se utilizarán en caso de que se efectúen modificaciones en los archivos con extensión .C y .CPP contenidos en ellos. Estos cambios implicarán la modificación de la programación y configuración del sistema de supervisión, para actualizar estas modificaciones en el sistema de supervisión será necesario compilar nuevamente los archivos con extensión .PRJ. Si no se realiza ninguna modificación al sistema de supervisión de la terminal marítima no será necesario realizar la operación anterior, por lo que no se utilizarán los discos durante la instalación.

## 5.2. ARRANQUE

El arranque del nuevo sistema de supervisión requiere en primer lugar que se ejecute el archivo DRVDAN.EXE el cual reconoce la tarjeta multipuertos en el sistema.

Una vez ejecutado este archivo, será necesario que se ejecute el Runtime del paquete Intouch de Wonderware, esto se conseguirá dando un doble clic con el mouse sobre el icono identificado con el nombre WindowViewer desplegado en el administrador de programas.

Estos archivos se ejecutan de manera automática si los iconos que corresponden a cada uno son introducidos dentro de la ventana de inicio de Windows.

Una vez realizado lo anterior, se desplegará en el monitor de la consola de operación la pantalla principal del nuevo sistema de supervisión.

### 5.3 MANTENIMIENTO

Además del mantenimiento normal que necesita la PC como lo es la limpieza general y periódica del hardware, sólo debe darse mantenimiento al disco duro. Esto se hace mediante un procedimiento en el que se reescriben todos los archivos del disco duro para que todas las partes de cada archivo se inscriban en sectores contiguos. El resultado es un mejoramiento significativo -hasta del 75 por ciento más- en la velocidad del disco en operaciones de recuperación. Durante una operación normal, los archivos del disco duro quedan fragmentados de tal forma que partes de uno se inscriben en distintos lugares del disco, lo que provoca que las operaciones de recuperación sean lentas, la instrucción del sistema operativo con la cual se da este mantenimiento es:

#### DEFRAG C:

Esta operación debe de realizarse una vez por mes para que el sistema funcione de manera óptima.

Las acciones necesarias para la instalación y el amaque del sistema de supervisión de la Terminal Marítima Pajaritos Ver., no son muy complicadas ni requieren de grandes conocimientos de software y hardware. El mantenimiento por ser únicamente al disco duro no requiere más que conocer la instrucción del sistema operativo que lleva a cabo este proceso.

---

## CONCLUSIONES

El controlar implica dirigir, ordenar o mandar mientras que monitorear significa que el operador supervisa de manera local (en el campo) o remota (en la sala de control) las variables más importantes del proceso para la generación de nuevos puntos de ajuste con la finalidad de reducir errores o mejorar la respuesta del sistema.

Un sistema de monitoreo recibe señales físicas del campo y las muestra al operador, en cambio un sistema de control dirige o regula un proceso para obtener la respuesta deseada del mismo.

Por lo expuesto en este trabajo se determina que el sistema realiza un control centralizado, el cual se lleva a cabo en el Microcontrolador Daniel MM3000. Cuando el sistema trabaja en modo controlador la computadora realiza la función de monitoreo ya que no puede alterar el proceso. Si el sistema se encuentra en modo computadora entonces se realiza un control supervisorio, ya que le permite al operador configurar lotes de carga, fijar nuevos puntos de ajuste (setpoint) durante el proceso y abrir o cerrar válvulas de manera directa.

El sistema de monitoreo y control puede trabajar en otras plataformas ya que la interfaz de operación seleccionada es compatible con Windows for Work Groups, Windows 95 y Windows NT, cumpliendo con el estándar DDE y NetDDE, lo cual le permite comunicarse con cualquier aplicación de Windows que cumpla con este estándar como: Excel, Word, FoxBase, etc. Además cuenta con acceso a manejadores de bases de datos como: Oracle, Dbase, Microsoft SQL Server entre otros.

El sistema de monitoreo y control tiene la posibilidad de trabajar en ambientes de redes Ethernet, Novell, Token Ring, ArcNet, entre otras, utilizando protocolos abiertos como el estándar NetBIOS con conectividad DecNet, Serial y TCP/IP. Para esto únicamente se tiene que adicionar una tarjeta de interfaz de red (NIC "Network Interface Card") y el sistema operativo de red en caso de que no se encuentre integrado en la plataforma en que se ejecute la aplicación.

La computadora posee ocho puertos seriales de los cuales el sistema ocupa seis de ellos, permitiendo con esto adicionar cuatro estaciones de medición controladas por dos Microcontroladores Daniel MM3000 sin que sea necesario adicionar hardware. En cuanto al software sólo se deben generar los programas ejecutables para dichas estaciones, cambiando únicamente:



- a) El número de puerto por el cual se transmitirá la información.
- b) El nombre de los archivos a los cuales se direccionará la información resultante.

En la interfaz de operación es necesario generar las pantallas para el monitoreo del Graphic Display de las estaciones adicionadas y direccionar la lectura de información hacia los nuevos archivos.

Para mejoras en el sistema de monitoreo es posible habilitar el monitoreo histórico de variables como temperatura, presión, totales de carga, etc., además del monitoreo histórico de alarmas.

Si se desea una modernización en el sistema de control no es necesario modificar el sistema de monitoreo ya que la interfaz de operación puede monitorear PLC's y DPC's de las marcas más reconocidas como Telemecanique, GE-Fanuc, Allen Bradley, Texas Instruments, Siemens, Bristol Babcock, etc.

El sistema no necesita de una UPS, ya que el cuarto de control cuenta con un banco de baterías que garantiza el suministro de energía por cuatro horas operando a toda su capacidad (todas las estaciones de carga y sistema de monitoreo operando).

Para el manejo del sistema de monitoreo y control no es necesaria una capacitación especializada ya que éste se ejecuta en ambiente Windows, sin embargo todo el personal que operó este sistema debe ser experto en el proceso de carga de los buques-tanque.

Este sistema puede operar en otras instalaciones de PEMEX ya que existen sistemas de medición similares al de la Terminal Marítima Pajaritos, en Coatzacoalcos Veracruz, como el de la estación de bombeo Las Palomas en el estado de Tabasco o la Terminal Marítima Nuevo Teapa en Veracruz.

---

APENDICE A.- PROGRAMAS DE LA INTERFAZ DE OPERACION.

La programación de la ventana que muestra los botones de las estaciones de carga y las solicitudes de operación es la siguiente:

```
IF Window == "B-1" THEN
BOTEST1 = 1;
BOTEST2 = 0;
BOTEST3 = 0;
BOTEST4 = 0;
BOTEST5 = 0;
BOTEST6 = 0;
BOTEST7 = 0;
BOTEST8 = 0;
BOTEST9 = 0;
BOTEST10 = 0;
ELSE
  IF Window == "B-2" THEN
BOTEST1 = 0;
BOTEST2 = 1;
BOTEST3 = 0;
BOTEST4 = 0;
BOTEST5 = 0;
BOTEST6 = 0;
BOTEST7 = 0;
BOTEST8 = 0;
BOTEST9 = 0;
BOTEST10 = 0;
  ELSE
    IF Window == "M-3W" THEN
BOTEST1 = 0;
BOTEST2 = 0;
BOTEST3 = 1;
BOTEST4 = 0;
BOTEST5 = 0;
BOTEST6 = 0;
BOTEST7 = 0;
BOTEST8 = 0;
BOTEST9 = 0;
BOTEST10 = 0;
    ELSE
      IF Window == "M-3E" THEN
BOTEST1 = 0;
BOTEST2 = 0;
BOTEST3 = 0;
BOTEST4 = 1;
BOTEST5 = 0;
BOTEST6 = 0;
BOTEST7 = 0;
BOTEST8 = 0;
BOTEST9 = 0;
BOTEST10 = 0;
      ELSE
        IF Window == "M-4W" THEN
BOTEST1 = 0;
```

```
BOTEST2 = 0;
BOTEST3 = 0;
BOTEST4 = 0;
BOTEST5 = 1;
BOTEST6 = 0;
BOTEST7 = 0;
BOTEST8 = 0;
BOTEST9 = 0;
BOTEST10 = 0;
ELSE
IF Window == "M-4E" THEN
  BOTEST1 = 0;
  BOTEST2 = 0;
  BOTEST3 = 0;
  BOTEST4 = 0;
  BOTEST5 = 0;
  BOTEST6 = 1;
  BOTEST7 = 0;
  BOTEST8 = 0;
  BOTEST9 = 0;
  BOTEST10 = 0;
ELSE
IF Window == "M-7W" THEN
  BOTEST1 = 0;
  BOTEST2 = 0;
  BOTEST3 = 0;
  BOTEST4 = 0;
  BOTEST5 = 0;
  BOTEST6 = 0;
  BOTEST7 = 1;
  BOTEST8 = 0;
  BOTEST9 = 0;
  BOTEST10 = 0;
ELSE
IF Window == "M-7E" THEN
  BOTEST1 = 0;
  BOTEST2 = 0;
  BOTEST3 = 0;
  BOTEST4 = 0;
  BOTEST5 = 0;
  BOTEST6 = 0;
  BOTEST7 = 0;
  BOTEST8 = 1;
  BOTEST9 = 0;
  BOTEST10 = 0;
ELSE
IF Window == "M-9W" THEN
  BOTEST1 = 0;
  BOTEST2 = 0;
  BOTEST3 = 0;
  BOTEST4 = 0;
  BOTEST5 = 0;
  BOTEST6 = 0;
  BOTEST7 = 0;
  BOTEST8 = 0;
  BOTEST9 = 1;
```



```
Hide "CONTROLALARMAS";
Show "PROVER DISPLAY";
Hide "COMANDO";
Show "ENCABEZADO";
ENDIF;
IF BOTON4 == 1 THEN
  BOTON1 = 0;
  Hide "ALARMA";
  Hide "CONTROLALARMAS";
  Show "BATCH DISPLAY";
  Hide "COMANDO";
  Show "ENCABEZADO";
ENDIF;
IF BOTON5 == 1 THEN
  BOTON1 = 0;
  Show "ALARMA";
  Hide "COMANDO";
  Show "ENCABEZADO";
  Show "CONTROLALARMAS";
ENDIF;
IF BOTON6 == 1 THEN
  BOTON1 = 0;
  Hide "ALARMA";
  Hide "CONTROLALARMAS";
  Show "DELIVERYTICKET";
  Hide "COMANDO";
  Show "ENCABEZADO";
ENDIF;
IF BOTON7 == 1 THEN
  Show "COMANDO";
  Hide "B-1";
  Hide "B-2";
  Hide "M-3E";
  Hide "M-3W";
  Hide "M-4E";
  Hide "M-4W";
  Hide "M-7E";
  Hide "M-7W";
  Hide "M-9E";
  Hide "M-9W";
  Hide "BATCH DISPLAY";
  Hide "PROVER DISPLAY";
  Hide "DELIVERYTICKET";
  Hide "PROVER DISPLAY";
  Hide "ALARMA";
  Hide "CONTROLALARMAS";
ENDIF;
```

La ventana que se presenta a la solicitud Graphic Display contiene la siguiente programación:

```
Show "ALARMAS";
g = g + 1;
IF g == 1 THEN
```

```

StartApp "g_db-1.exe";
ENDIF;
IF g == 2 THEN
FileReadFields( "D:\g_db-1.TXT", 0,"GRAPHICDISPB1",B1);

IF GRAPHICDISPB3 == "." THEN
    B1401A = 0;
    B1401B = 0;
ELSE
    IF GRAPHICDISPB3 == "," THEN
        B1401A = 1;
        B1401B = 0;
    ELSE
        IF GRAPHICDISPB3 == "9" THEN
            B1401A = 2;
            B1401B = 0;
        ELSE
            IF GRAPHICDISPB3 == ">" THEN
                B1401A = 0;
                B1401B = 1;
            ELSE
                IF GRAPHICDISPB3 == "6" THEN
                    B1401A = 0;
                    B1401B = 2;
                ELSE
                    IF GRAPHICDISPB3 == "=" THEN
                        B1401A = 2;
                        B1401B = 1;
                    ELSE
                        IF GRAPHICDISPB3 == "5" THEN
                            B1401A = 2;
                            B1401B = 2;
                        ELSE
                            IF GRAPHICDISPB3 == "7" THEN
                                B1401A = 1;
                                B1401B = 2;
                            ELSE
                                IF GRAPHICDISPB3 == "?" THEN
                                    B1401A = 1;
                                    B1401B = 1;
                                ELSE
                                    IF GRAPHICDISPB3 == "8" AND B1401B == 0 THEN
                                        B1401A = 3;
                                        B1401B = 0;
                                    ELSE
                                        IF GRAPHICDISPB3 == "8" AND B1401B == 1 THEN
                                            B1401A = 3;
                                            B1401B = 1;
                                        ELSE
                                            IF GRAPHICDISPB3 == "2" AND B1401A == 0 THEN
                                                B1401A = 0;
                                                B1401B = 3;
                                            ELSE
                                                IF GRAPHICDISPB3 == "2" AND B1401A == 1 THEN
                                                    B1401A = 1;
                                                    B1401B = 3;
                                                ENDIF;
                                            ENDIF;
                                        ENDIF;
                                    ENDIF;
                                ENDIF;
                            ENDIF;
                        ENDIF;
                    ENDIF;
                ENDIF;
            ENDIF;
        ENDIF;
    ENDIF;
ENDIF;

```



```

        B1402B = 0;
ELSE
IF GRAPHICDISPB4 == "8" AND B1402B == 1 THEN
        B1402A = 3;
        B1402B = 1;
ELSE
IF GRAPHICDISPB4 == "2" AND B1402A == 0 THEN
        B1402A = 0;
        B1402B = 3;
ELSE
IF GRAPHICDISPB4 == "2" AND B1402A == 1 THEN
        B1402A = 1;
        B1402B = 3;
ELSE
IF GRAPHICDISPB4 == "0" THEN
        B1402A = 3;
        B1402B = 3;
ENDIF;
ENDIF;
ENDIF;
ENDIF;
ENDIF;
ENDIF;
ENDIF;
ENDIF;
ENDIF;
ENDIF;
ENDIF;
ENDIF;
ENDIF;
ENDIF;
ENDIF;
ENDIF;
IF GRAPHICDISPB5 == "." THEN
        B1403A = 0;
        B1403B = 0;
ELSE
IF GRAPHICDISPB5 == "," THEN
        B1403A = 1;
        B1403B = 0;
ELSE
IF GRAPHICDISPB5 == "9" THEN
        B1403A = 2;
        B1403B = 0;
ELSE
IF GRAPHICDISPB5 == ">" THEN
        B1403A = 0;
        B1403B = 1;
ELSE
IF GRAPHICDISPB5 == "6" THEN
        B1403A = 0;
        B1403B = 2;
ELSE
IF GRAPHICDISPB5 == "=" THEN
        B1403A = 2;
        B1403B = 1;
ELSE
IF GRAPHICDISPB5 == "5" THEN

```







```
ENDIF;
IF GRAPHICDISPB7 == ":" THEN
  B2401A = 0;
  B2401B = 0;
ELSE
  IF GRAPHICDISPB7 == ";" THEN
    B2401A = 1;
    B2401B = 0;
  ELSE
    IF GRAPHICDISPB7 == "9" THEN
      B2401A = 2;
      B2401B = 0;
    ELSE
      IF GRAPHICDISPB7 == ">" THEN
        B2401A = 0;
        B2401B = 1;
      ELSE
        IF GRAPHICDISPB7 == "6" THEN
          B2401A = 0;
          B2401B = 2;
        ELSE
          IF GRAPHICDISPB7 == "=" THEN
            B2401A = 2;
            B2401B = 1;
          ELSE
            IF GRAPHICDISPB7 == "5" THEN
              B2401A = 2;
              B2401B = 2;
            ELSE
              IF GRAPHICDISPB7 == "7" THEN
                B2401A = 1;
                E7101B = 2;
              ELSE
                IF GRAPHICDISPB7 == "?" THEN
                  B2401A = 1;
                  B2401B = 1;
                ELSE
                  IF GRAPHICDISPB7 == "8" AND B2401B == 0 THEN
                    B2401A = 3;
                    B2401B = 0;
                  ELSE
                    IF GRAPHICDISPB7 == "8" AND B2401B == 1 THEN
                      B2401A = 3;
                      B2401B = 1;
                    ELSE
                      IF GRAPHICDISPB7 == "2" AND B2401A == 0 THEN
                        B2401A = 0;
                        B2401B = 3;
                      ELSE
                        IF GRAPHICDISPB7 == "2" AND B2401A == 1 THEN
                          B2401A = 1;
                          B2401B = 3;
                        ELSE
                          IF GRAPHICDISPB7 == "0" THEN
                            B2401A = 3;
                            B2401B = 3;
```



```
B2402B = 1;
ELSE
  IF GRAPHICDISPB8 == "2" AND B2402A == 0 THEN
    B2402A = 0;
    B2402B = 3;
  ELSE
    IF GRAPHICDISPB8 == "2" AND B2402A == 1 THEN
      B2402A = 1;
      B2402B = 3;
    ELSE
      IF GRAPHICDISPB8 == "0" THEN
        B2402A = 3;
        B2402B = 3;
      ENDIF;
    ENDIF;
  ENDIF;
ENDIF;
ENDIF;
ENDIF;
ENDIF;
ENDIF;
ENDIF;
ENDIF;
ENDIF;
ENDIF;
ENDIF;
ENDIF;
ENDIF;
ENDIF;
ENDIF;
ENDIF;
ENDIF;
IF GRAPHICDISPB9 == "." THEN
  B2403A = 0;
  B2403B = 0;
ELSE
  IF GRAPHICDISPB9 == "," THEN
    B2403A = 1;
    B2403B = 0;
  ELSE
    IF GRAPHICDISPB9 == "9" THEN
      B2403A = 2;
      B2403B = 0;
    ELSE
      IF GRAPHICDISPB9 == ">" THEN
        B2403A = 0;
        B2403B = 1;
      ELSE
        IF GRAPHICDISPB9 == "6" THEN
          B2403A = 0;
          B2403B = 2;
        ELSE
          IF GRAPHICDISPB9 == "=" THEN
            B2403A = 2;
            B2403B = 1;
          ELSE
            IF GRAPHICDISPB9 == "5" THEN
              B2403A = 2;
              B2403B = 2;
            ELSE
              IF GRAPHICDISPB9 == "7" THEN
```





```

STRAINERB13 = 0;
STRAINERB14 = 0;
ELSE
  IF GRAPHICDISPB15 == "2" THEN
    STRAINERB11 = 0;
    STRAINERB12 = 1;
    STRAINERB13 = 0;
    STRAINERB14 = 0;
  ELSE
    IF GRAPHICDISPB15 == "4" THEN
      STRAINERB11 = 0;
      STRAINERB12 = 0;
      STRAINERB13 = 1;
      STRAINERB14 = 0;
    ELSE
      IF GRAPHICDISPB15 == "8" THEN
        STRAINERB11 = 0;
        STRAINERB12 = 0;
        STRAINERB13 = 0;
        STRAINERB14 = 1;
      ELSE
        IF GRAPHICDISPB15 == "3" THEN
          STRAINERB11 = 1;
          STRAINERB12 = 1;
          STRAINERB13 = 0;
          STRAINERB14 = 0;
        ELSE
          IF GRAPHICDISPB15 == "5" THEN
            STRAINERB11 = 1;
            STRAINERB12 = 0;
            STRAINERB13 = 1;
            STRAINERB14 = 0;
          ELSE
            IF GRAPHICDISPB15 == "9" THEN
              STRAINERB11 = 1;
              STRAINERB12 = 0;
              STRAINERB13 = 0;
              STRAINERB14 = 1;
            ELSE
              IF GRAPHICDISPB15 == "6" THEN
                STRAINERB11 = 0;
                STRAINERB12 = 1;
                STRAINERB13 = 1;
                STRAINERB14 = 0;
              ELSE
                IF GRAPHICDISPB15 == "." THEN
                  STRAINERB11 = 0;
                  STRAINERB12 = 1;
                  STRAINERB13 = 0;
                  STRAINERB14 = 1;
                ELSE
                  IF GRAPHICDISPB15 == "<" THEN
                    STRAINERB11 = 0;
                    STRAINERB12 = 0;
                    STRAINERB13 = 1;
                    STRAINERB14 = 1;

```







```
B2401D = 0;
ELSE
  IF GRAPHICDISPB13 == "6" THEN
    B1401E = 2;
    B2401D = 0;
  ELSE
    IF GRAPHICDISPB13 == "=" THEN
      B1401E = 1;
      B2401D = 2;
    ELSE
      IF GRAPHICDISPB13 == "5" THEN
        B1401E = 2;
        B2401D = 2;
      ELSE
        IF GRAPHICDISPB13 == "7" THEN
          B1401E = 2;
          B2401D = 1;
        ELSE
          IF GRAPHICDISPB13 == "7" THEN
            B1401E = 1;
            B2401D = 1;
          ELSE
            IF GRAPHICDISPB13 == "8" AND B1401E == 0 THEN
              B1401E = 0;
              B2401D = 3;
            ELSE
              IF GRAPHICDISPB13 == "8" AND B1401E == 1 THEN
                B1401E = 1;
                B2401D = 3;
              ELSE
                IF GRAPHICDISPB13 == "8" AND B1401E == 2 THEN
                  B1401E = 2;
                  B2401D = 3;
                ELSE
                  IF GRAPHICDISPB13 == "2" AND B2401D == 0 THEN
                    B1401E = 3;
                    B2401D = 0;
                  ELSE
                    IF GRAPHICDISPB13 == "2" AND B2401D == 1 THEN
                      B1401E = 3;
                      B2401D = 1;
                    ELSE
                      IF GRAPHICDISPB13 == "2" AND B2401D == 2 THEN
                        B1401E = 3;
                        B2401D = 2;
                      ENDIF;
                    ENDIF;
                  ENDIF;
                ENDIF;
              ENDIF;
            ENDIF;
          ENDIF;
        ENDIF;
      ENDIF;
    ENDIF;
  ENDIF;
ENDIF;
```

```

ENDIF;
ENDIF;
ENDIF;
ENDIF;
IF GRAPHICDISPB11 == "." THEN
    B1401C = 0;
    B1401D = 0;
ELSE
    IF GRAPHICDISPB11 == "," THEN
        B1401C = 1;
        B1401D = 0;
    ELSE
        IF GRAPHICDISPB11 == "9" THEN
            B1401C = 2;
            B1401D = 0;
        ELSE
            IF GRAPHICDISPB11 == ">" THEN
                B1401C = 0;
                B1401D = 1;
            ELSE
                IF GRAPHICDISPB11 == "6" THEN
                    B1401C = 0;
                    B1401D = 2;
                ELSE
                    IF GRAPHICDISPB11 == "7" THEN
                        B1401C = 1;
                        B1401D = 1;
                    ELSE
                        IF GRAPHICDISPB11 == "7" THEN
                            B1401C = 1;
                            B1401D = 2;
                        ELSE
                            IF GRAPHICDISPB11 == "=" THEN
                                B1401C = 2;
                                B1401D = 1;
                            ELSE
                                IF GRAPHICDISPB11 == "5" THEN
                                    B1401C = 5;
                                    B1401D = 5;
                                ELSE
                                    IF GRAPHICDISPB11 == "8" AND B1401D == 0 THEN
                                        B1401C = 3;
                                        B1401D = 0;
                                    ELSE
                                        IF GRAPHICDISPB11 == "8" AND B1401D == 2 THEN
                                            B1401C = 3;
                                            B1401D = 2;
                                        ELSE
                                            IF GRAPHICDISPB11 == "2" AND B1401C == 0 THEN
                                                B1401C = 0;
                                                B1401D = 3;
                                            ELSE
                                                IF GRAPHICDISPB11 == "2" AND B1401C == 1 THEN
                                                    B1401C = 1;
                                                    B1401D = 3;
                                                ELSE

```

```
IF GRAPHICDISPB11 == "0" THEN
  B1401C = 3;
  B1401D = 3;
ENDIF;
ENDIF;
ENDIF;
ENDIF;
ENDIF;
ENDIF;
ENDIF;
ENDIF;
ENDIF;
ENDIF;
ENDIF;
ENDIF;
ENDIF;
ENDIF;
ENDIF;
IF GRAPHICDISPB12 == "." THEN
  B2401C = 0;
  B401D = 0;
ELSE
  IF GRAPHICDISPB12 == "," THEN
    B2401C = 1;
    B401D = 0;
  ELSE
    IF GRAPHICDISPB12 == "9" THEN
      B2401C = 2;
      B401D = 0;
    ELSE
      IF GRAPHICDISPB12 == ">" THEN
        B2401C = 0;
        B401D = 1;
      ELSE
        IF GRAPHICDISPB12 == "6" THEN
          B2401C = 0;
          B401D = 2;
        ELSE
          IF GRAPHICDISPB12 == "7" THEN
            B2401C = 1;
            B401D = 1;
          ELSE
            IF GRAPHICDISPB12 == "7" THEN
              B2401C = 1;
              B401D = 2;
            ELSE
              IF GRAPHICDISPB12 == "=" THEN
                B2401C = 2;
                B401D = 1;
              ELSE
                IF GRAPHICDISPB12 == "5" THEN
                  B2401C = 5;
                  B401D = 5;
                ELSE
                  IF GRAPHICDISPB12 == "8" AND B401D == 0 THEN
                    B2401C = 3;
                    B401D = 0;
```



```
B2401E = 0;
ALARMPROBAB = 1;
ENDIF;
ENDIF;
ENDIF;
ENDIF;
ENDIF;
ENDIF;
IF B2401B == 2 OR B2402B == 2 OR B2403B == 2 OR B2404B == 2 THEN
    VALGRALB2 = 2;
ELSE
    VALGRALB2 = 0;
ENDIF;
ENDIF;
IF g == 3 THEN
    g = 0;
ENDIF;
```

La ventana que se presenta a la solicitud Proof Report funciona conjuntamente con una ventana que contiene un botón para la impresión del reporte de prueba. La programación de estas ventanas se muestra a continuación:

#### PROOF REPORT

```
Show "botonera3";
g = g + 1;

IF g == 1 THEN
    PREXE = "P_R" + Window + ".EXE";
    StartApp PREXE;
ENDIF;

IF g == 4 THEN
    PRTXT = "D:P_R" + Window + ".TXT";
    FileReadFields( PRTXT, 0, "proofreport1", 50 );
    g = 0;
ENDIF;
```

#### BOTON DE IMPRESIÓN (botonera3)

```
PrintWindow("PROOFREPORT",1,1,,0,0,0);
```

La programación de la ventana que se despliega al hacer la solicitud Prover Display se muestra a continuación:

```
Show "ALARMAS";
g = g + 1;

IF g == 1 THEN
    prodis = "P_D" + Window + ".EXE";
    StartApp prodis;
ENDIF;
```

```

IF g == 2 THEN
PDTXT = "D:\P_D" + Window + ".TXT";
FileReadFields(PDTXT, 0, "PROVERDISPLAY1", 40 );
ENDIF;

```

```

IF g == 3 THEN
g = 0;
ENDIF;

```

La ventana que despliega la solicitud Batch Display se tiene la siguiente programación:

```

Show "ALARMAS";
g = g + 1;

```

```

IF g == 1 THEN
BDEXE = "B_D" + Window + ".EXE";
StartApp BDEXE;
ENDIF;
IF g == 2 THEN
BDTXT = "D:\DIS" + Window + ".TXT";
FileReadFields( BDTXT, 0, "BATCHDISP1", 56 );
ENDIF;

```

```

IF g == 3 THEN
g = 0;
ENDIF;

```

Para la solicitud Alarm Report no se requiere de ninguna programación debido a que dentro de la interfaz Intouch es posible declarar variables del tipo alarma, y siempre que tengan un valor diferente de cero aparecerán en esta pantalla. Esta solicitud funciona conjuntamente con una ventana de impresión (controlalarmas), la programación para esta ventana se muestra a continuación:

BOTON DE IMPRESION (controlalarmas)

```
PrintWindow("ALARMA",1,1,,0,0,0);
```

La ventana que proporciona la solicitud Delivery Ticket funciona conjuntamente con una ventana de impresión (botonera2). La programación para estas ventanas se muestra a continuación:

```
g = g + 1;
```

```
IF g == 1 THEN
Show "ALARMAS";
```



```
dellic = "d_l" + Window + ".exe";
StartApp dellic;
ENDIF;

IF g == 3 THEN
PRTXT = "D:\D_T" + Window + ".TXT";
FileReadFields(PRTXT, 0, "DELTICKET1",28 );
g = 0;
ENDIF;

ON SHOW

g = 0;
Show "botonera2";

BOTON DE IMPRESION (botonera2)

PrintWindow("DELIVERYTICKET",1,1,,0,0,0);
```

La programación para la ventana desplegada por la solicitud Command es la siguiente:

```
I = I + 1;
ESTACION = StringInString(COMANDO,"B1",1,1 );
PUERTO = StringMid( COMANDO,ESTACION, 2 );
IF ESTACION == 0 THEN
ESTACION = StringInString(COMANDO,"B2",1,1 );
PUERTO = StringMid( COMANDO,ESTACION, 2 );
IF ESTACION == 0 THEN
ESTACION = StringInString(COMANDO,"3E",1,1 );
PUERTO = StringMid( COMANDO,ESTACION, 2 );
IF ESTACION == 0 THEN
ESTACION = StringInString(COMANDO,"3W",1,1 );
PUERTO = StringMid( COMANDO,ESTACION, 2 );
IF ESTACION == 0 THEN
ESTACION = StringInString(COMANDO,"4E",1,1 );
PUERTO = StringMid( COMANDO,ESTACION, 2 );
IF ESTACION == 0 THEN
ESTACION = StringInString(COMANDO,"4W",1,1 );
PUERTO = StringMid( COMANDO,ESTACION, 2 );
IF ESTACION == 0 THEN
ESTACION = StringInString(COMANDO,"7E",1,1 );
PUERTO = StringMid( COMANDO,ESTACION, 2 );
IF ESTACION == 0 THEN
ESTACION = StringInString(COMANDO,"7W",1,1 );
PUERTO = StringMid( COMANDO,ESTACION, 2 );
IF ESTACION == 0 THEN
ESTACION = StringInString(COMANDO,"9E",1, 2 );
IF ESTACION == 0 THEN
ESTACION = StringInString(COMANDO,"9W",1,1 );
PUERTO = StringMid( COMANDO,ESTACION, 2 );
ENDIF;
```

```

ENDIF;
ENDIF;
ENDIF;
ENDIF;
ENDIF;
ENDIF;
ENDIF;
ENDIF;
IF PUERTO <> "" THEN
  PUERTO2 = PUERTO;
  ENDF;
  COMRESP1 = StringInString(COMANDO,"START",1,1);
  COMRESP2 = StringMid( COMANDO,COMRESP1, 5 );
IF COMRESP1 == 0 THEN
  COMRESP1 = StringInString(COMANDO,"LOAD",1,1);
  COMRESP2 = StringMid( COMANDO,COMRESP1, 4 );
IF COMRESP1 == 0 THEN
  COMRESP1 = StringInString(COMANDO,"TOPOFF",1,1);
  COMRESP2 = StringMid( COMANDO,COMRESP1, 6 );
IF COMRESP1 == 0 THEN
  COMRESP1 = StringInString(COMANDO,"SUSPEND",1,1);
  COMRESP2 = StringMid( COMANDO,COMRESP1, 7 );
IF COMRESP1 == 0 THEN
  COMRESP1 = StringInString(COMANDO,"TERMINATE",1,1);
  COMRESP2 = StringMid( COMANDO,COMRESP1, 9 );
IF COMRESP1 == 0 THEN
  COMRESP1 = StringInString(COMANDO,"OPEN",1,1);
  COMRESP2 = StringMid( COMANDO,COMRESP1, 4 );
IF COMRESP1 == 0 THEN
  COMRESP1 = StringInString(COMANDO,"CLOSE",1,1);
  COMRESP2 = StringMid( COMANDO,COMRESP1, 5 );
IF COMRESP1 == 0 THEN
  COMRESP1 = StringInString(COMANDO,"DISABLE",1,1);
  COMRESP2 = StringMid( COMANDO,COMRESP1, 7 );
IF COMRESP1 == 0 THEN
  COMRESP1 = StringInString(COMANDO,"ENABLE",1,1);
  COMRESP2 = StringMid( COMANDO,COMRESP1, 6 );
IF COMRESP1 == 0 THEN
  COMRESP1 = StringInString(COMANDO,"PROVE",1,1);
  COMRESP2 = StringMid( COMANDO,COMRESP1, 5 );
IF COMRESP1 == 0 THEN
  COMRESP1 = StringInString(COMANDO,"ABORT",1,1);
  COMRESP2 = StringMid( COMANDO,COMRESP1, 5 );
IF COMRESP1 == 0 THEN
  COMRESP1 = StringInString(COMANDO,"RESET",1,1);
  COMRESP2 = StringMid( COMANDO,COMRESP1, 5 );
IF COMRESP1 == 0 THEN
  COMRESP1 = StringInString(COMANDO,"MF",1,1);
  COMRESP2 = StringMid( COMANDO,COMRESP1, 2 );
ENDIF;
ENDIF;
ENDIF;
ENDIF;
ENDIF;
ENDIF;

```

```
ENDIF;
ENDIF;
ENDIF;
ENDIF;
ENDIF;
IF COMANDO1 <> ">" THEN
IF COMRESP2 == "SETUP" THEN
g = 40;
ENDIF;
IF g == 41 THEN
COMANDO3 = StringUpper( COMANDO1 );
COMANDO = StringChar( 1 ) + StringChar( 55 ) + COMANDO3;
FileWriteMessage("D:\COMANDO1.TXT",0,COMANDO,1);
ENDIF;
IF g == 42 THEN
COMEXE = "COM" + PUERTO2 + ".EXE";
StartApp COMEXE;
ENDIF;
IF g == 45 THEN
FileReadMessage("D:\COMANDO2.TXT",0,RESPUESTA1,38);
ENDIF;
IF g == 47 THEN
FileDelete( "D:\COMANDO1.TXT" );
FileDelete( "D:\COMANDO2.TXT" );
COMANDO1 = ">";
g = 40;
ENDIF;
g = g + 1;
IF COMRESP2 <> "START" AND COMRESP2 <> "LOAD" AND COMRESP2 <> "TOPOFF"
AND COMRESP2 <> "SUSPEND" AND COMRESP2 <> "TERMINATE" AND COMRESP2 <>
"OPEN" AND COMRESP2 <> "CLOSE" AND COMRESP2 <> "DISABLE" AND COMRESP2 <>
"ENABLE" AND COMRESP2 <> "PROVE" AND COMRESP2 <> "ABORT" AND COMRESP2 <>
"RESET" AND COMRESP2 <> "MF" AND COMANDO1 <> "Y" AND COMANDO1 <> "N" AND
COMANDO1 <> "<" THEN
COMANDO3 = StringUpper( COMANDO1 );
COMANDO = StringChar( 1 ) + StringChar( 55 ) + COMANDO3;
IF g == 21 THEN
FileWriteMessage("D:\COMANDO1.TXT",0,COMANDO,1);
ENDIF;
IF g == 22 THEN
COMEXE = "COM" + PUERTO2 + ".EXE";
StartApp COMEXE;
ENDIF;
IF g == 25 THEN
FileReadMessage("D:\COMANDO2.TXT",0,RESPUESTA1,38);
ENDIF;
IF g == 27 THEN
FileDelete( "D:\COMANDO1.TXT" );
FileDelete( "D:\COMANDO2.TXT" );
COMANDO1 = ">";
g = 20;
ENDIF;
ENDIF;
IF COMRESP2 == "START" OR COMRESP2 == "LOAD" OR COMRESP2 == "TOPOFF" OR
COMRESP2 == "SUSPEND" OR COMRESP2 == "TERMINATE" OR COMRESP2 == "OPEN" OR
COMRESP2 == "CLOSE" OR COMRESP2 == "DISABLE" OR COMRESP2 == "ENABLE" OR
```

```

COMRESP2 == "PROVE" OR COMRESP2 == "ABORT" OR COMRESP2 == "RESET" OR
COMRESP2 == "MF" OR COMANDO1 == "Y" OR COMANDO1 == "N" THEN
  COMANDO3 = StringUpper( COMANDO1 );
COMANDO = StringChar( 1 ) + StringChar( 55 ) + COMANDO3;
IF g == 31 THEN
FileWriteMessage("D:\COMANDO1.TXT",0,COMANDO,1);
ENDIF;
IF g == 32 THEN
COMEXE = "COM" + PUERTO2 + ".EXE";
StartApp COMEXE;
ENDIF;
IF g == 35 THEN
FileReadMessage("D:\COMANDO2.TXT",0,RESPUESTA1,38);
ENDIF;
IF g == 37 THEN
FileDelete( "D:\COMANDO1.TXT" );
FileDelete( "D:\COMANDO2.TXT" );
  COMANDO1 = ">";
IF ANTONIO == 2 THEN
  COMANDO1 = ">";
ENDIF;
g =20;
ENDIF;
ENDIF;
ENDIF;

ON SHOW

g =20;
COMANDO1 = ">";
RESPUESTA1 = " ";
PUERTO2 = "";
Show "botonera1";
Show "ENCABEZADO";
Hide "botonera2";

```

## APENDICE B.- PROGRAMACIÓN DEL DRIVER DE COMUNICACIÓN.

### Graphic Display

<b>DECLARACIÓN DE LIBRERIAS</b>	
#include<windows.h>	Librería estandar para Windows
#include<stdio.h>	Librería estandar de "C"
#include"mx_win.h"	Librería de la tarjeta multipuertos
<b>DECLARACIÓN DE CONSTANTES</b>	
#define len1 298	Define la longitud de la cadena de respuesta del controlador
#define len2 378	Define la longitud de la cadena que se grabara en el archivo G_D-B1.TXT
<b>DECLARACIÓN DE VARIABLES</b>	
int lener1 = 300;	Contiene el valor de la longitud de la cadena de respuesta del controlador
int lener2 = 380;	Contiene el valor de la longitud de la cadena que se grabara en el archivo G_D-B1.TXT
int port = 10;	Define el puerto por el cual se transmiten y reciben los datos
int term = 13;	Define el caracter terminador de la cadena (0D H ASCII)
int lener = 3;	Define la longitud de la cadena con la solicitud Graphic Display
int i,j;	Declaración de variables enteras
char graphdis1[len1];	Declaración de la cadena que recibe la respuesta original.
char grapdis2[len2];	Declaración de la cadena que recibe el formato CSV
char bufer[4]={1,54,13};	Declaración de la cadena con la solicitud Graphic Display
FILE *ft;	Declaración del puntero al archivo
<b>PROGRAMA</b>	
void main() {	
sio_init();	Lee el archivo de configuración de la tarjeta
sio_ioctl(port,B9600,CS7 Stop2 ParityEven);	Configura el puerto para la comunicación
sio_open(port);	Habilita el puerto para recibir/transmitir
sio_flush(port,2);	Limpia el buffer de entrada y salida de datos
sio_write(port,bufer,lener);	Envia la solicitud de datos hacia el Daniel MM3000
for(i=0;i<150;i++){for(j=0;j<30000;j++);	Retardo de tiempo para esperar respuesta del Daniel MM3000
sio_read(port,graphdis1,lener1);	Lee la cadena del buffer de entrada de datos
sio_close(port);	Deshabilita el puerto para recibir/transmitir
<b>INCERSION DE COMAS EN LA CADENA ORIGINAL PARA DARLE FORMATO CSV</b>	
grapdis2[0]=graphdis1[0];	
grapdis2[1]='.';	
grapdis2[2]=graphdis1[1];	
grapdis2[3]='.';	

```

grapdis2[4]=graphdis1[2];
grapdis2[5]=';';
grapdis2[6]=graphdis1[3];
grapdis2[7]=';';
grapdis2[8]=graphdis1[4];
grapdis2[9]=';';
grapdis2[10]=graphdis1[5];
grapdis2[11]=';';
grapdis2[12]=graphdis1[6];
grapdis2[13]=';';
grapdis2[14]=graphdis1[7];
grapdis2[15]=';';
grapdis2[16]=graphdis1[8];
grapdis2[17]=';';
grapdis2[18]=graphdis1[9];
grapdis2[19]=';';
grapdis2[20]=graphdis1[10];
grapdis2[21]=';';
grapdis2[22]=graphdis1[11];
grapdis2[23]=';';
grapdis2[24]=graphdis1[12];
grapdis2[25]=';';
grapdis2[26]=graphdis1[13];
grapdis2[27]=';';
grapdis2[28]=graphdis1[14];
grapdis2[29]=';';
grapdis2[30]=graphdis1[15];
grapdis2[31]=';';
grapdis2[32]=graphdis1[16];
grapdis2[33]=';';
grapdis2[34]=graphdis1[17];
grapdis2[35]=';';
grapdis2[36]=graphdis1[18];
grapdis2[37]=';';
for (i=38;i<=42;i++)
grapdis2[i]=grapdis1[i-19];
grapdis2[43]=';';
grapdis2[44]=graphdis1[24];
grapdis2[45]=';';
for (i=46;i<=50;i++)
grapdis2[i]=grapdis1[i-21];
grapdis2[51]=';';
grapdis2[52]=graphdis1[30];
grapdis2[53]=';';
for (i=54;i<=68;i++)
grapdis2[i]=grapdis1[i-23];
grapdis2[59]=';';
grapdis2[60]=graphdis1[36];
grapdis2[61]=';';
for (i=62;i<=66;i++)
grapdis2[i]=grapdis1[i-25];
grapdis2[67]=';';
grapdis2[68]=graphdis1[42];
grapdis2[69]=';';
for (i=70;i<=74;i++)
grapdis2[i]=grapdis1[i-27];

```

```
grapdis2[75]='';
grapdis2[76]=graphdis1[48];
grapdis2[77]='';
for (i=78;i<=82;i++)
grapdis2[i]=grapdis1[i-29];
grapdis2[83]='';
grapdis2[84]=graphdis1[54];
grapdis2[85]='';
for (i=86;i<=90;i++)
grapdis2[i]=grapdis1[i-31];
grapdis2[91]='';
grapdis2[92]=graphdis1[60];
grapdis2[93]='';
for (i=94;i<=98;i++)
grapdis2[i]=grapdis1[i-33];
grapdis2[99]='';
grapdis2[100]=graphdis1[66];
grapdis2[101]='';
for (i=102;i<=106;i++)
grapdis2[i]=grapdis1[i-35];
grapdis2[107]='';
for (i=108;i<=113;i++)
grapdis2[i]=grapdis1[i-36];
grapdis2[114]='';
for (i=115;i<=120;i++)
grapdis2[i]=grapdis1[i-37];
grapdis2[121]='';
for (i=122;i<=127;i++)
grapdis2[i]=grapdis1[i-38];
grapdis2[128]='';
for (i=129;i<=134;i++)
grapdis2[i]=grapdis1[i-39];
grapdis2[135]='';
for (i=136;i<=141;i++)
grapdis2[i]=grapdis1[i-40];
grapdis2[142]='';
for (i=143;i<=148;i++)
grapdis2[i]=grapdis1[i-41];
grapdis2[149]='';
for (i=150;i<=155;i++)
grapdis2[i]=grapdis1[i-42];
grapdis2[156]='';
for (i=157;i<=162;i++)
grapdis2[i]=grapdis1[i-43];
grapdis2[163]='';
for (i=164;i<=169;i++)
grapdis2[i]=grapdis1[i-44];
grapdis2[170]='';
for (i=171;i<=176;i++)
grapdis2[i]=grapdis1[i-45];
grapdis2[179]='';
for (i=180;i<=187;i++)
grapdis2[i]=grapdis1[i-46];
grapdis2[188]='';
for (i=189;i<=196;i++)
grapdis2[i]=grapdis1[i-47];
```

```

grapdis2[197]=';
for (i=198;i<=205;i++)
grapdis2[i]=grapdis1[i-48];
grapdis2[206]=';
for (i=207;i<=214;i++)
grapdis2[i]=grapdis1[i-49];
grapdis2[215]=';
for (i=216;i<=223;i++)
grapdis2[i]=grapdis1[i-50];
grapdis2[224]=';
for (i=225;i<=232;i++)
grapdis2[i]=grapdis1[i-51];
grapdis2[233]=';
for (i=234;i<=241;i++)
grapdis2[i]=grapdis1[i-52];
grapdis2[242]=';
for (i=243;i<=250;i++)
grapdis2[i]=grapdis1[i-53];
grapdis2[251]=';
for (i=252;i<=257;i++)
grapdis2[i]=grapdis1[i-54];
grapdis2[258]=';
for (i=259;i<=266;i++)
grapdis2[i]=grapdis1[i-55];
grapdis2[267]=';
for (i=268;i<=273;i++)
grapdis2[i]=grapdis1[i-56];
grapdis2[274]=';
for (i=275;i<=282;i++)
grapdis2[i]=grapdis1[i-57];
grapdis2[283]=';
grapdis2[284]=graphdis1[226];
grapdis2[285]=';
for (i=286;i<=290;i++)
grapdis2[i]=grapdis1[i-59];
grapdis2[291]=';
grapdis2[292]=graphdis1[232];
grapdis2[293]=';
for (i=294;i<=298;i++)
grapdis2[i]=grapdis1[i-61];
grapdis2[299]=';
grapdis2[300]=graphdis1[238];
grapdis2[301]=';
for (i=302;i<=306;i++)
grapdis2[i]=grapdis1[i-63];
grapdis2[307]=';
grapdis2[308]=graphdis1[244];
grapdis2[309]=';
for (i=310;i<=314;i++)
grapdis2[i]=grapdis1[i-65];
grapdis2[315]=';
grapdis2[316]=graphdis1[250];
grapdis2[317]=';
for (i=318;i<=322;i++)
grapdis2[i]=grapdis1[i-67];
grapdis2[323]=';

```



```

grapdis2[324]=graphdis1[256];
grapdis2[325]='';
for (i=326;i<=330;i++)
grapdis2[i]=grapdis1[i-69];
grapdis2[331]='';
grapdis2[332]=graphdis1[262];
grapdis2[333]='';
for (i=334;i<=338;i++)
grapdis2[i]=grapdis1[i-71];
grapdis2[339]='';
grapdis2[340]=graphdis1[268];
grapdis2[341]='';
for (i=342;i<=346;i++)
grapdis2[i]=grapdis1[i-73];
grapdis2[347]='';
for (i=348;i<=353;i++)
grapdis2[i]=grapdis1[i-74];
grapdis2[354]='';
for (i=355;i<=360;i++)
grapdis2[i]=grapdis1[i-75];
grapdis2[361]='';
grapdis2[362]=graphdis1[286];
grapdis2[363]='';
for (i=364;i<=369;i++)
grapdis2[i]=grapdis1[i-77];
grapdis2[369]='';
grapdis2[370]=graphdis1[292];
grapdis2[371]='';
for (i=372;i<=376;i++)
grapdis2[i]=grapdis1[i-79];
grapdis2[377]='';
grapdis2[378]=graphdis1[298];

```

```
ft=fopen("d:\g_db-1.txt","w+");
```

```
fwrite( grapdis2,lener2,1,ft);
```

```
fcloseall();
```

Crea si no existe un archivo para leer y escribir llamado g\_db-1.txt  
Escribe la cadena con formato CSV en el archivo g\_db-1.txt  
Cierra todos los archivos que hayan sido abiertos

### Proof Report

```

#include<windows.h>
#include<stdio.h>
#include"mx_win.h"

```

#### DECLARACIÓN DE LIBRERIAS

Librería estandar para Windows  
Librería estandar de "C"  
Librería de la tarjeta multipuertos

```
#define len1 1681
```

#### DECLARACIÓN DE CONSTANTES

Define la longitud de la cadena de respuesta del controlador  
Define la longitud de la cadena que se grabara en el archivo P\_R-B1.TXT

```
#define len2 1717
```

#### DECLARACIÓN DE VARIABLES

```
int lener1 = 1683;
```

Contiene el valor de la longitud de la cadena de respuesta del controlador



int lener2 = 1719;	Contiene el valor de la longitud de la cadena que se grabara en el archivo P_R-B1.TXT
int port = 10;	Define el puerto por el cual se transmiten y reciben los datos
int term = 13;	Define el caracter terminador de la cadena (OD H ASCII)
int lener = 3;	Define la longitud de la cadena con la solicitud Proof Report
int i,j;	Declaración de variables enteras
char prof1[len1];	Declaración de la cadena que recibe la respuesta original.
char prof2[len2];	Declaración de la cadena que recibe el formato CSV
char bufer[4]={1,50,13};	Declaración de la cadena con la solicitud Proof Report
FILE *ft;	Declaración del puntero al archivo

**PROGRAMA**

```

void main()
{
sio_init();           Lee el archivo de configuración de la tarjeta
sio_ioctl(port,B9600, CS7|Stop2|ParityEven);  Configura el puerto para la comunicación
sio_open(port);      Habilita el puerto para recibir/transmitir
sio_flush(port,2);   Limpia el buffer de entrada y salida de datos
sio_write(port,bufer,lener);  Envía la solicitud de datos hacia el Daniel MM3000

for(i=0;i<500;i++)for(j=0;j<30000;j++);       Retardo de tiempo para esperar respuesta del Daniel MM3000

sio_read(port,prof1,len1);  Lee la cadena del buffer de entrada de datos
sio_close(port);           Deshabilita el puerto para recibir/transmitir
INCERSIÓN DE COMAS EN LA CADENA ORIGINAL PARA DARLE FORMATO CSV
prof2[0]=prof1[0];
prof2[1]='.';
prof2[2]=prof1[1];
prof2[3]='.';
for (i=2;i<49;i++)
prof2[i+2]=prof1[i];
prof2[52]='.';
for (i=49;i<99;i++)
prof2[i+4]=prof1[i];
prof2[102]='.';
for (i=99;i<146;i++)
prof2[i+4]=prof1[i];
prof2[151]='.';
for (i=146;i<195;i++)
prof2[i+5]=prof1[i];
prof2[201]='.';
for (i=195;i<243;i++)
prof2[i+6]=prof1[i];
prof2[249]='.';
for (i=243;i<291;i++)
prof2[i+7]=prof1[i];
prof2[298]='.';
for (i=291;i<339;i++)
prof2[i+8]=prof1[i];
prof2[347]='.';

```

```
for (i=339;i<387;++)
prof2[i+9]=prof1[i];
prof2[396]='.';
for (i=387;i<435;++)
prof2[i+10]=prof1[i];
prof2[445]='.';
for (i=435;i<495;++)
prof2[i+11]=prof1[i];
prof2[495]='.';
for (i=495;i<531;++)
prof2[i+12]=prof1[i];
prof2[543]='.';
for (i=531;i<579;++)
prof2[i+13]=prof1[i];
prof2[592]='.';
for (i=579;i<627;++)
prof2[i+14]=prof1[i];
prof2[641]='.';
for (i=627;i<675;++)
prof2[i+15]=prof1[i];
prof2[690]='.';
for (i=675;i<723;++)
prof2[i+16]=prof1[i];
prof2[739]='.';
for (i=723;i<771;++)
prof2[i+17]=prof1[i];
prof2[788]='.';
for (i=771;i<819;++)
prof2[i+18]=prof1[i];
prof2[837]='.';
for (i=819;i<867;++)
prof2[i+19]=prof1[i];
prof2[886]='.';
for (i=867;i<915;++)
prof2[i+20]=prof1[i];
prof2[935]='.';
for (i=915;i<963;++)
prof2[i+21]=prof1[i];
prof2[984]='.';
for (i=963;i<1011;++)
prof2[i+22]=prof1[i];
prof2[1033]='.';
for (i=1011;i<1059;++)
prof2[i+23]=prof1[i];
prof2[1082]='.';
for (i=1059;i<1107;++)
prof2[i+24]=prof1[i];
prof2[1131]='.';
for (i=1107;i<1155;++)
prof2[i+25]=prof1[i];
prof2[1180]='.';
for (i=1155;i<1203;++)
prof2[i+26]=prof1[i];
prof2[1229]='.';
for (i=1203;i<1251;++)
prof2[i+27]=prof1[i];
```

```

prof2[1278]=';';
for (i=1251;i<1299;i++)
prof2[i+28]=prof1[i];
prof2[1327]=';';
for (i=1299;i<1347;i++)
prof2[i+29]=prof1[i];
prof2[1376]=';';
for (i=1347;i<1394;i++)
prof2[i+30]=prof1[i];
prof2[1425]=';';
for (i=1394;i<1443;i++)
prof2[i+31]=prof1[i];
prof2[1474]=';';
for (i=1443;i<1491;i++)
prof2[i+32]=prof1[i];
prof2[1523]=';';
for (i=1491;i<1539;i++)
prof2[i+33]=prof1[i];
prof2[1572]=';';
for (i=1539;i<1587;i++)
prof2[i+34]=prof1[i];
prof2[1621]=';';
for (i=1587;i<1635;i++)
prof2[i+35]=prof1[i];
prof2[1670]=';';
for (i=1635;i<1682;i++)
prof2[i+36]=prof1[i];

```

```

ft=fopen("d:\p_rb-1.txt","w+");
fwrite( prof2,lener2,1,ft);
fcloseall();

```

Crea si no existe un archivo para leer y escribir llamado p\_rb-1.txt  
 Escribe la cadena con formato CSV en el archivo p\_rb-1.txt  
 Cierra todo los archivos que hayan sido abiertos

**Prover Display**

```

#include<windows.h>
#include<stdio.h>
#include"mx_win.h"

```

**DECLARACIÓN DE LIBRERIAS**

Libreria estandar para Windows  
 Libreria estandar de "C"  
 Libreria de la tarjeta multipuertos

```

#define len1 203
#define len2 239

```

**DECLARACIÓN DE CONSTANTES**

Define la longitud de la cadena de respuesta del controlador  
 Define la longitud de la cadena que se grabara en el archivo P\_D-B1.TXT

```

int lener1 = 205;
int lener2 = 239;
int port = 10;
int term = 13;

```

**DECLARACIÓN DE VARIABLES**

Contiene el valor de la longitud de la cadena de respuesta del controlador  
 Contiene el valor de la longitud de la cadena que se grabara en el archivo P\_D-B1.TXT  
 Define el puerto por el cual se transmiten y reciben los datos  
 Define el caracter terminador de la cadena (0D)

```

int lener = 3;
int i,j;
char prover1[len1];
char prover2[len2];
char bufer[4]={1,51,13};
FILE *ft;

```

H ASCII)  
Define la longitud de la cadena con la solicitud Prover Display  
Declaración de variables enteras  
Declaración de la cadena que recibe la respuesta original.  
Declaración de la cadena que recibe el formato CSV  
Declaración de la cadena con la solicitud Prover Display  
Declaración del puntero al archivo

**PROGRAMA**

```

void main()
{
sio_init();
sio_ioctl(port,B9600,CS7|Stop2|ParityEven);
sio_open(port);
sio_flush(port,2);
sio_write(port,bufer,lener);

for(i=0;i<250;i++)for(j=0;j<30000;j++);

sio_read(port,prover1,lener1);
sio_close(port);

```

Lee el archivo de configuración de la tarjeta  
Configura el puerto para la comunicación  
Habilita el puerto para recibir/transmitir  
Limpia el buffer de entrada y salida de datos  
Envía la solicitud de datos hacia el Daniel MM3000  
Retardo de tiempo para esperar respuesta del Daniel MM3000  
Lee la cadena del buffer de entrada de datos  
Deshabilita el puerto para recibir/transmitir

**INCERSON DE COMAS EN LA CADENA ORIGINAL PARA DARLE FORMATO CSV**

```

prover2[0]=prover1[0];
prover2[1]=';';
prover2[2]=prover1[1];
prover2[3]=';';
for (i=4;i<=12;i++)
prover2[i]=prover1[i-2];
prover2[13]=';';
for (i=14;i<=18;i++)
prover2[i]=prover1[i-3];
prover2[19]=';';
for (i=20;i<=33;i++)
prover2[i]=prover1[i-4];
prover2[34]=';';
for (i=35;i<=42;i++)
prover2[i]=prover1[i-5];
prover2[43]=';';
for (i=44;i<=49;i++)
prover2[i]=prover1[i-6];
prover2[50]=';';
for (i=51;i<=56;i++)
prover2[i]=prover1[i-7];
prover2[57]=';';
for (i=58;i<=63;i++)
prover2[i]=prover1[i-8];
prover2[64]=';';
for (i=65;i<=70;i++)
prover2[i]=prover1[i-9];

```

```

prover2[71]=';
for (i=72;i<=77;i++)
prover2[i]=prover1[i-10];
prover2[78]=';
prover2[79]=prover1[68];
prover2[80]=prover1[69];
prover2[81]=';
for (i=82;i<=87;i++)
prover2[i]=prover1[i-12];
prover2[88]=';
for (i=89;i<=94;i++)
prover2[i]=prover1[i-13];
prover2[95]=';
for (i=96;i<=101;i++)
prover2[i]=prover1[i-14];
prover2[102]=';
for (i=103;i<=108;i++)
prover2[i]=prover1[i-15];
prover2[109]=';
for (i=110;i<=117;i++)
prover2[i]=prover1[i-16];
prover2[118]=';
prover2[119]=prover1[102];
prover2[120]=prover1[103];
prover2[121]=';
for (i=122;i<=128;i++)
prover2[i]=prover1[i-18];
prover2[128]=';
for (i=129;i<=134;i++)
prover2[i]=prover1[i-19];
prover2[135]=';
for (i=136;i<=141;i++)
prover2[i]=prover1[i-20];
prover2[142]=';
for (i=143;i<=148;i++)
prover2[i]=prover1[i-21];
prover2[149]=';
for (i=150;i<=157;i++)
prover2[i]=prover1[i-22];
prover2[158]=';
prover2[159]=prover1[136];
prover2[160]=prover1[137];
prover2[161]=';
for (i=162;i<=187;i++)
prover2[i]=prover1[i-24];
prover2[168]=';
for (i=169;i<=174;i++)
prover2[i]=prover1[i-25];
prover2[175]=';
for (i=176;i<=181;i++)
prover2[i]=prover1[i-26];
prover2[182]=';
for (i=183;i<=188;i++)

```

```

prover2[i]=prover1[i-27];
prover2[189]=';';
for (i=190;i<=197; i++)
prover2[i]=prover1[i-28];
prover2[198]=';';
prover2[199]=prover1[170];
prover2[200]=prover1[171];
prover2[201]=';';
for (i=202;i<=207; i++)
prover2[i]=prover1[i-30];
prover2[208]=';';
for (i=209;i<=214; i++)
prover2[i]=prover1[i-31];
prover2[215]=';';
for (i=216;i<=221; i++)
prover2[i]=prover1[i-32];
prover2[222]=';';
for (i=223;i<=228; i++)
prover2[i]=prover1[i-33];
prover2[229]=';';
for (i=230;i<=237; i++)
prover2[i]=prover1[i-34];
prover2[238]=';';

```

```
ft=fopen("d:\p_db-1.txt","w+");
```

```
fwrite( prover2,lener2,1,ft);
```

```
fcloseall();}
```

Crea si no existe un archivo para leer y escribir llamado p\_db-1.txt  
Escribe la cadena con formato CSV en el archivo p\_db-1.txt  
Cierra todo los archivos que hayan sido abiertos

### Batch Display

```

#include<windows.h>
#include<stdio.h>
#include"mx_wln.h"

```

```
#define len1 375
```

```
#define len2 425
```

```
int lener1 = 377;
```

```
int lener2 = 427;
```

```
int port = 10;
```

```
int term = 13;
```

```
int lener = 4;
```

### DECLARACIÓN DE LIBRERIAS

Libreria estandar para Windows  
Libreria estandar de "C"  
Libreria de la tarjeta multipuertos

### DECLARACIÓN DE CONSTANTES

Define la longitud de la cadena de respuesta del controlador  
Define la longitud de la cadena que se grabara en el archivo DISB-1.TXT

### DECLARACIÓN DE VARIABLES

Contiene el valor de la longitud de la cadena de respuesta del controlador  
Contiene el valor de la longitud de la cadena que se grabara en el archivo DISB-1.TXT  
Define el puerto por el cual se transmiten y reciben los datos  
Define el caracter terminador de la cadena (OD H ASCII)  
Define la longitud de la cadena con la solicitud Batch Display



int i,j;	Declaración de variables enteras
char batch1[len1];	Declaración de la cadena que recibe la respuesta original.
char batch2[len2];	Declaración de la cadena que recibe el formato CSV
char bufer[4]={1,53,50,13};	Declaración de la cadena con la solicitud Prover Display
FILE *ft;	Declaración del puntero al archivo

**PROGRAMA**

void main()	
{	
sio_init();	Lee el archivo de configuración de la tarjeta
sio_ioctl(port,B9600,CS7 Stop2 ParityEven);	Configura el puerto para la comunicación
sio_open(port);	Habilita el puerto para recibir/transmitir
sio_flush(port,2);	Limpia el buffer de entrada y salida de datos
sio_write(port,bufer,4);	Envía la solicitud de datos hacia el Daniel MM3000
for (i=0;i<150;i++)for (j=0;j<30000;j++);	Retardo de tiempo para esperar respuesta del Daniel MM3000
sio_read(port,batch1,len1);	Lee la cadena del buffer de entrada de datos
sio_close(port);	Deshabilita el puerto para recibir/transmitir

**INCERSION DE COMAS EN LA CADENA ORIGINAL PARA DARLE FORMATO CSV**

```

batch2[0]=batch1[0];
batch2[1]=';';
batch2[2]=batch1[1];
batch2[3]=';';
batch2[4]=batch1[2];
batch2[5]=';';
batch2[6]=batch1[3];
batch2[7]=';';
for (i=8;i<=45;i++)
batch2[i]=batch1[i-4];
batch2[46]=';';
for (i=47;i<=54;i++)
batch2[i]=batch1[i-5];
batch2[55]=';';
for (i=56;i<=67;i++)
batch2[i]=batch1[i-6];
batch2[68]=';';
for (i=69;i<=74;i++)
batch2[i]=batch1[i-7];
batch2[75]=';';
for (i=76;i<=83;i++)
batch2[i]=batch1[i-8];
batch2[84]=';';
for (i=85;i<=90;i++)
batch2[i]=batch1[i-9];
batch2[91]=';';
for (i=92;i<=99;i++)
batch2[i]=batch1[i-10];
batch2[100]=';';
for (i=101;i<=108;i++)
batch2[i]=batch1[i-11];
batch2[107]=';';
    
```



```
for (i=108;j<=115;j++)
batch2[i]=batch1[i-12];
batch2[116]=';';
for (j=117;j<=124;j++)
batch2[j]=batch1[j-13];
batch2[125]=';';
for (i=126;j<=131;j++)
batch2[i]=batch1[i-14];
batch2[132]=';';
for (j=133;j<=138;j++)
batch2[j]=batch1[j-15];
batch2[139]=';';
for (i=140;j<=145;j++)
batch2[i]=batch1[i-16];
batch2[146]=';';
for (j=147;j<=154;j++)
batch2[j]=batch1[j-17];
batch2[155]=';';
for (i=156;j<=163;j++)
batch2[i]=batch1[i-18];
batch2[164]=';';
for (j=165;j<=172;j++)
batch2[j]=batch1[j-19];
batch2[173]=';';
for (i=174;j<=181;j++)
batch2[i]=batch1[i-20];
batch2[182]=';';
for (j=183;j<=188;j++)
batch2[j]=batch1[j-21];
batch2[189]=';';
for (i=190;j<=195;j++)
batch2[i]=batch1[i-22];
batch2[196]=';';
for (j=197;j<=202;j++)
batch2[j]=batch1[j-23];
batch2[203]=';';
for (i=204;j<=211;j++)
batch2[i]=batch1[i-24];
batch2[212]=';';
for (j=213;j<=220;j++)
batch2[j]=batch1[j-25];
batch2[221]=';';
for (i=222;j<=229;j++)
batch2[i]=batch1[i-26];
batch2[230]=';';
for (j=231;j<=238;j++)
batch2[j]=batch1[j-27];
batch2[239]=';';
for (i=240;j<=245;j++)
batch2[i]=batch1[i-28];
batch2[246]=';';
for (j=247;j<=252;j++)
batch2[j]=batch1[j-29];
batch2[253]=';';
for (i=254;j<=259;j++)
batch2[i]=batch1[i-30];
```

```

batch2[260]=';
for (i=261;i<=268;i++)
batch2[i]=batch1[i-31];
batch2[269]=';
for (i=270;i<=277;i++)
batch2[i]=batch1[i-32];
batch2[278]=';
for (i=279;i<=286;i++)
batch2[i]=batch1[i-33];
batch2[287]=';
for (i=288;i<=295;i++)
batch2[i]=batch1[i-34];
batch2[296]=';
for (i=297;i<=302;i++)
batch2[i]=batch1[i-35];
batch2[303]=';
for (i=304;i<=309;i++)
batch2[i]=batch1[i-36];
batch2[310]=';
for (i=311;i<=316;i++)
batch2[i]=batch1[i-37];
batch2[317]=';
for (i=318;i<=325;i++)
batch2[i]=batch1[i-38];
batch2[326]=';
for (i=327;i<=334;i++)
batch2[i]=batch1[i-39];
batch2[335]=';
for (i=336;i<=343;i++)
batch2[i]=batch1[i-40];
batch2[344]=';
for (i=345;i<=352;i++)
batch2[i]=batch1[i-41];
batch2[353]=';
for (i=354;i<=359;i++)
batch2[i]=batch1[i-42];
batch2[360]=';
for (i=361;i<=366;i++)
batch2[i]=batch1[i-43];
batch2[367]=';
for (i=368;i<=373;i++)
batch2[i]=batch1[i-44];
batch2[374]=';
for (i=375;i<=382;i++)
batch2[i]=batch1[i-45];
batch2[383]=';
for (i=384;i<=391;i++)
batch2[i]=batch1[i-46];
batch2[392]=';
for (i=393;i<=398;i++)
batch2[i]=batch1[i-47];
batch2[399]=';
for (i=400;i<=405;i++)
batch2[i]=batch1[i-48];
batch2[406]=';
for (i=407;i<=414;i++)

```

```

batch2[i]=batch1[i-49];
batch2[415]='';
for (i=416;i<=423;i++)
batch2[i]=batch1[i-50];
batch2[424]='';
batch2[425]=batch1[374];

```

```
ft=fopen("d:\disb-1.txt","w+");
```

```
fwrite( batch2,lener2,1,ft);
```

```
fcloseall();
```

Crea si no existe un archivo para leer y escribir llamado disb-1.txt  
Escribe la cadena con formato CSV en el archivo disb-1.txt  
Cierra todo los archivos que hayan sido abiertos

### Alarm Report

```

#include <windows.h>
#include <stdio.h>
#include <stdlib.h>
#include "mx_win.h"

```

#### DECLARACIÓN DE LIBRERIAS

Libreria estandar para Windows  
Libreria estandar de "C"  
Libreria estandar de "C"  
Libreria de la tarjeta multipuertos

```
#define len1 375
```

#### DECLARACIÓN DE CONSTANTES

Define la longitud de la cadena de respuesta del controlador

```
int lener1 = 377;
```

```
int port = 10;
```

```
int term = 13;
```

```
int lener = 3;
```

```
int
```

```
status,l,j,act,actua,ch,acum,place,i1,i2,i3,soni;
```

```
char batch1[len1];
```

```
char bufer[4]={1,49,13};
```

```
FILE *f1,*fent,*fsal;
```

```
char regla1[2],regla2[2];
```

```
char regscan[70],posi[230],regul,uno,cero;
```

#### DECLARACIÓN DE VARIABLES

Contiene el valor de la longitud de la cadena de respuesta del controlador

Define el puerto por el cual se transmiten y reciben los datos

Define el caracter terminador de la cadena (OD H ASCII)

Define la longitud de la cadena con la solicitud Graphic Display

Declaración de variables enteras

Declaración de la cadena que recibe la respuesta original.

Declaración de la cadena con la solicitud Graphic Display

Declaración de punteros a archivos

Declaración de cadenas auxiliares

Declaración de cadenas y variables auxiliares

#### PROGRAMA

```
void main()
```

```
{
```

```
sio_init();
```

```
sio_ioctl(port,B9600,CS7|Stop2|ParityEven);
```

```
sio_open(port);
```

```
sio_flush(port,2);
```

```
sio_write(port,bufer,lener);
```

Lee el archivo de configuración de la tarjeta  
Configura el puerto para la comunicación  
Habilita el puerto para recibir/transmitir  
Limpia el buffer de entrada y salida de datos  
Envia la solicitud de datos hacia el Daniel MM3000

```
for (i=0;i<150;i++)for (j=0;j<30000;j++);
```

Retardo de tiempo para esperar respuesta del Daniel MM3000  
Lee la cadena del buffer de entrada de datos  
Deshabilita el puerto para recibir/transmitir

```
sio_read(port,batch1,len1);
sio_close(port);
ft=fopen("d:\scan-b.txt","w+");
fwrite(batch1,len1,1,ft);
fclose(ft);
fent=fopen("d:\scan-b.txt","r");
fsal=fopen("d:\son-b.txt","w");
actua=2;
act=0;
```

**PROCESAMIENTO DE LA CADENA PARA ENTREGAR EL ARCHIVO A INTOUCH**

```
fread(regscan,sizeof(regscan),1,fent);
while(actua<58&&act<260)
{
j=0;
if(actua==2||actua==3||actua==4)
{
for(i=0;i<10;i++)
{
ch=0;
itoa(i,cade,10);
posi[act+i]='0';
if(regscan[actua]==cade[0])
{
ch=1;
posi[act+i]=NULL;
posi[act+i]='1';
fprintf(fsal,"%c",posi[act+i]);
};
if(ch==0) fprintf(fsal,"%c",posi[act+i]);
};
};
cade[0]=NULL;
cade[1]=NULL;
if(actua==5)
{
for(i=0;i<3;i++)
{
ch=0;
itoa(i,cade,10);
posi[act+i]='0';
if(regscan[actua]==cade[0])
{
ch=1;
posi[act+i]=NULL;
posi[act+i]='1';
fprintf(fsal,"%c",posi[act+i]);
};
if(ch==0) fprintf(fsal,"%c",posi[act+i]);
};
};
i=0;
```

```

j=0;
if(actua==6)
{
    for(i=0;j<3;j++)
    {
        ch=0;
        j=i+1;
        if(i==2)j=0; j=2*i;);
        ltoa(j,cade,10);
        posi[act+i]='0';
        if(regscan[actua]==cade[0])
        {
            ch=1;
            posi[act+i]=NULL;
            posi[act+i]='1';
            fprintf(fsai,"%c,",posi[act+i]);
        };
        if(ch==0) fprintf(fsai,"%c,",posi[act+i]);
    };
};
if(actua==52)
{
    uno='1';
    cero='0';
    if(regscan[53] != '.' && regscan[53] !=
';){
        regla1[0] = regscan[52];
        regla2[0] = regscan[53];
        i1=atoi(regla1);
        i2=atoi(regla2);
        i1=i1-1;
        i3=(i1*11)+i2;
        for(soni=-8;soni<99;soni++)
            if(soni%i3)
                fprintf(fsai,"%c,",cero);
            else
                fprintf(fsai,"%c,",uno);
    };
    if(regscan[53] == '.' || regscan[53] == ';'){
        regla1[0] = regscan[52];
        regla2[0] = regscan[53];
        i1=atoi(regla1);
        if(regla2[0] == '.')i2=10;
        if(regla2[0] == ';')i2=11;
        i1=i1-1;
        i3=(i1*11)+i2;
        for(soni=-8;soni<99;soni++)
            if(soni%i3)
                fprintf(fsai,"%c,",cero);
            else
                fprintf(fsai,"%c,",uno);
    };
};
}

```

```

while(actual=2&&actual=3&&actual=4&&actual
=5&&actual=6&&actual=52&&actual=53)
(
    while(regscan[actua]!='0')
    (
        while(regscan[actua]=='1' ||
regscan[actua]=='2' || regscan[actua]=='4' ||
regscan[actua]=='8')
        {
            regul=regscan[actua];
            if(regul=='1')
            {
                posi[act]=NULL;
                posi[act+1]=NULL;
                posi[act+2]=NULL;
                posi[act+3]=NULL;
                posi[act]='1';
                posi[act+1]='0';
                posi[act+2]='0';
                posi[act+3]='0';
            };
            if(regul=='2')
            {
                posi[act]=NULL;
                posi[act+1]=NULL;
                posi[act+2]=NULL;
                posi[act+3]=NULL;
                posi[act]='0';
                posi[act+1]='1';
                posi[act+2]='0';
                posi[act+3]='0';
            };
            if(regul=='4')
            {
                posi[act]=NULL;
                posi[act+1]=NULL;
                posi[act+2]=NULL;
                posi[act+3]=NULL;
                posi[act]='0';
                posi[act+1]='0';
                posi[act+2]='1';
                posi[act+3]='0';
            };
            if(regul=='8')
            {
                posi[act]=NULL;
                posi[act+1]=NULL;
                posi[act+2]=NULL;
                posi[act+3]=NULL;
                posi[act]='0';
                posi[act+1]='0';
                posi[act+2]='0';
                posi[act+3]='1';
            };
        }
    )
)

```

```
    }  
    fprintf(fsal,"%c,%c,%c,%c,",posi[act],posi[act+1],  
    posi[act+2],posi[act+3]);  
    break;  
    );  
    while(regscan[actua]=='3' ||  
    regscan[actua]=='6' || regscan[actua]=='<') // <  
    representa un 12  
    {  
        regul=regscan[actua];  
        if(regul=='3')  
        {  
            i=1;  
            posi[act]=NULL;  
            posi[act+1]=NULL;  
            posi[act+2]=NULL;  
            posi[act+3]=NULL;  
            posi[act]='1';  
            posi[act+1]='1';  
            posi[act+2]='0';  
            posi[act+3]='0';  
        };  
        if(regul=='6')  
        {  
            i=1;  
            posi[act]=NULL;  
            posi[act+1]=NULL;  
            posi[act+2]=NULL;  
            posi[act+3]=NULL;  
            posi[act]='0';  
            posi[act+1]='1';  
            posi[act+2]='1';  
            posi[act+3]='0';  
        };  
        if(regul=='<')  
        {  
            i=1;  
            posi[act]=NULL;  
            posi[act+1]=NULL;  
            posi[act+2]=NULL;  
            posi[act+3]=NULL;  
            posi[act]='0';  
            posi[act+1]='0';  
            posi[act+2]='1';  
            posi[act+3]='1';  
        };  
        if(i==1)  
        fprintf(fsal,"%c,%c,%c,%c,",posi[act],posi[act+1],  
        posi[act+2],posi[act+3]);  
        break;  
    };  
    while(regscan[actua]=='5' ||  
    regscan[actua]==':') // : representa un 10
```





```

        posi[act]=0;
        posi[act+1]='1';
        posi[act+2]='1';
        posi[act+3]='1';
    };
    if(i==1)
fprintf(fsai,"%c,%c,%c,%c,%c,",posi[act],posi[act+1],
posi[act+2],posi[act+3]);
        break;
    };
    while(regscan[actua]!='9')
    {
        regul=regscan[actua];
        i=1;
        posi[act]=NULL;
        posi[act+3]=NULL;
        posi[act+1]='0';
        posi[act+2]='0';
        posi[act]='1';
        posi[act+3]='1';
        if(i==1)
fprintf(fsai,"%c,%c,%c,%c,%c,",posi[act],posi[act+1],
posi[act+2],posi[act+3]);
        break;
    };
    while(regscan[actua]!=':') //
representa un 11
    {
        regul=regscan[actua];
        i=1;
        posi[act]=NULL;
        posi[act+1]=NULL;
        posi[act+3]=NULL;
        posi[act+2]='0';
        posi[act]='1';
        posi[act+1]='1';
        posi[act+3]='1';
        if(i==1)
fprintf(fsai,"%c,%c,%c,%c,%c,",posi[act],posi[act+1],
posi[act+2],posi[act+3]);
        break;
    };
    while(regscan[actua]!='=') //
representa un 13
    {
        regul=regscan[actua];
        i=1;
        posi[act]=NULL;
        posi[act+2]=NULL;
        posi[act+3]=NULL;
        posi[act]='1';
        posi[act+1]='0';
        posi[act+2]='1';
        posi[act+3]='1';

```

```

        if(i==1)
fprintf(fsal,"%c,%c,%c,%c,",posi[act],posi[act+1],
posi[act+2],posi[act+3]);
        break;
    );
    while(regscan[actua]!='?') //
representa un 15
    {
        regul=regscan[actua];
        i=1;
        posi[act]=NULL;
        posi[act+1]=NULL;
        posi[act+2]=NULL;
        posi[act+3]=NULL;
        posi[act]='1';
        posi[act+1]='1';
        posi[act+2]='1';
        posi[act+3]='1';
        if(i==1)
fprintf(fsal,"%c,%c,%c,%c,",posi[act],posi[act+1],
posi[act+2],posi[act+3]);
        break;
    );
    break;
);
if(regscan[actua]=='0')
{
    posi[act]=NULL;
    posi[act+1]=NULL;
    posi[act+2]=NULL;
    posi[act+3]=NULL;
    posi[act]='0';
    posi[act+1]='0';
    posi[act+2]='0';
    posi[act+3]='0';
};
break;
);
    act=act+4;
    if(actua==2||actua==3||actua==4)
act=act+8;
    actua=actua+1;
};
if(act<220)
{
    act=act-4;
    acum=act;
    for(act=acum;act<220;act++)
    {
        posi[act]=NULL;
        posi[act]='0';
        fprintf(fsal,"%c,",posi[act]);
    };
};
};

```

```
fcloseall();
}
```

### Delivery Ticket

```
#include<windows.h>
#include<stdio.h>
#include<string.h>
#include"mx_win.h"
```

**DECLARACIÓN DE LIBRERIAS**  
 Librería estandar para Windows  
 Librería estandar de "C"  
 Librería estandar de "C"  
 Librería de la tarjeta multipuertos

```
#define len1 1120
#define len2 1152
```

**DECLARACIÓN DE CONSTANTES**  
 Define la longitud de la cadena de respuesta del controlador  
 Define la longitud de la cadena que se grabara en el archivo D\_TB-1.TXT

```
int lener1 = 1122;
int lener2 = 1154;
int port = 10;
int term = 13;
int lener = 4;
int i, j, largo;
char deltic1[len1];
char deltic2[len2];
char bufer[4]={1,52,49,13};
FILE *ft;
```

**DECLARACIÓN DE VARIABLES**  
 Contiene el valor de la longitud de la cadena de respuesta del controlador  
 Contiene el valor de la longitud de la cadena que se grabara en el archivo D\_TB-1.TXT  
 Define el puerto por el cual se transmiten y reciben los datos  
 Define el caracter terminador de la cadena (OD H ASCII)  
 Define la longitud de la cadena con la solicitud Delivery Ticket  
 Declaración de variables enteras  
 Declaración de la cadena que recibe la respuesta original.  
 Declaración de la cadena que recibe el formato CSV  
 Declaración de la cadena con la solicitud Delivery Ticket  
 Declaración del puntero al archivo

**PROGRAMA**

```
void main()
{
  sio_init();
  sio_ioctl(port,B9600, CS7|Stop2|ParityEven);
  sio_open(port);
  sio_flush(port,2);
  sio_write(port,bufer,lener);
  for (i=0;i<400;i++)for (j=0;j<30000;j++);
  sio_read(port,deltic1,len1);
  sio_close(port);
  INCERSION DE COMAS EN LA CADENA ORIGINAL PARA DARLE FORMATO CSV
  largo = strlen(deltic1);
```

Lee el archivo de configuración de la tarjeta  
 Configura el puerto para la comunicación  
 Habilita el puerto para recibir/transmitir  
 Limpia el buffer de entrada y salida de datos  
 Envía la solicitud de datos hacia el Daniel MM3000  
 Retardo de tiempo para esperar respuesta del Daniel MM3000  
 Lee la cadena del buffer de entrada de datos  
 Deshabilita el puerto para recibir/transmitir



```

if (largo==1060 || largo == 1012 || largo == 292){
dellic2[0]=dellic1[0];
dellic2[1]=';';
dellic2[2]=dellic1[1];
dellic2[3]=';';
dellic2[4]=dellic1[2];
dellic2[5]=';';
dellic2[6]=dellic1[3];
dellic2[7]=';';
for (i=4;i<52;i++)
dellic2[i+4]=dellic1[i];
dellic2[56]=';';
for (i=52;i<100;i++)
dellic2[i+5]=dellic1[i];
dellic2[105]=';';
for (i=100;i<148;i++)
dellic2[i+6]=dellic1[i];
dellic2[154]=';';
for (i=148;i<196;i++)
dellic2[i+7]=dellic1[i];
dellic2[203]=';';
for (i=196;i<244;i++)
dellic2[i+8]=dellic1[i];
dellic2[252]=';';
for (i=244;i<292;i++)
dellic2[i+9]=dellic1[i];
dellic2[301]=';';
for (i=292;i<340;i++)
dellic2[i+10]=dellic1[i];
dellic2[350]=';';
for (i=340;i<388;i++)
dellic2[i+11]=dellic1[i];
dellic2[399]=';';
for (i=388;i<436;i++)
dellic2[i+12]=dellic1[i];
dellic2[448]=';';
for (i=436;i<484;i++)
dellic2[i+13]=dellic1[i];
dellic2[497]=';';
for (i=484;i<532;i++)
dellic2[i+14]=dellic1[i];
dellic2[546]=';';
for (i=532;i<580;i++)
dellic2[i+15]=dellic1[i];
dellic2[595]=';';
for (i=580;i<628;i++)
dellic2[i+16]=dellic1[i];
dellic2[644]=';';
for (i=628;i<676;i++)
dellic2[i+17]=dellic1[i];
dellic2[693]=';';
for (i=676;i<724;i++)
dellic2[i+18]=dellic1[i];
dellic2[742]=';';
for (i=724;i<772;i++)
dellic2[i+19]=dellic1[i];

```

```

deltic2[791]=';
for (i=772;i<820;i++)
deltic2[i+20]=deltic1[i];
deltic2[840]=';
for (i=820;i<868;i++)
deltic2[i+21]=deltic1[i];
deltic2[889]=';
for (i=868;i<916;i++)
deltic2[i+22]=deltic1[i];
deltic2[938]=';
for (i=916;i<964;i++)
deltic2[i+23]=deltic1[i];
deltic2[987]=';
for (i=964;i<1011;i++)
deltic2[i+24]=deltic1[i+1];
deltic2[1035]=';
for (i=1011;i<1059;i++)
deltic2[i+25]=deltic1[i+2];
deltic2[1082]=';
for (i=1059;i<1107;i++)
deltic2[i+26]=deltic1[i];
deltic2[1132]=';
}

```

#### GENERACION DEL MENSAJE PARA EL USUARIO

```

if (largo==109){
for (i = 1; i <= 18; i +2)
deltic2[i]=' ';
for (i = 0; i <= 18; i +2)
deltic2[i]=' ';
deltic2[19]='T';
deltic2[20]='I';
deltic2[21]='C';
deltic2[22]='K';
deltic2[23]='E';
deltic2[24]='T';
deltic2[25]=' ';
deltic2[26]='E';
deltic2[27]='N';
deltic2[28]=' ';
deltic2[29]='P';
deltic2[30]='R';
deltic2[31]='O';
deltic2[32]='C';
deltic2[33]='E';
deltic2[34]='S';
deltic2[35]='O';
for (i = 36; i <= 98; i +2)
deltic2[i]=' ';
for (i = 37; i <= 98; i +2)
deltic2[i]=' ';
lener2=98;
};

```

```
ft=fopen("d:\d_ib-1.txt","w+");
```

```

Crea si no existe un archivo para leer y escribir
llamado d_ib-1.txt

```

```
fwrite( deltic2,lener2, 1,ft);
```

Escribe la cadena con formato CSV en el archivo d\_lb-1.txt

```
fcloseall();
```

Cierra todo los archivos que hayan sido abiertos

**Command**

**DECLARACIÓN DE LIBRERIAS**

```
#include<windows.h>
#include<stdio.h>
#include"mx_win.h"
```

Librería estándar para Windows  
 Librería estándar de "C"  
 Librería de la tarjeta multipuertos

**DECLARACIÓN DE VARIABLES**

```
int port = 10;
int term = 13;
int lener2=39;
int abajo,i,j;
char buf[21];
char buf1[38];
int buf3[20];
FILE *fp;
```

Define el puerto por el cual se transmiten y reciben los datos  
 Define el caracter terminador de la cadena (0D H ASCII)  
 Contiene el valor de la longitud de la cadena que se grabara en el archivo COMANDO2.TXT  
 Declaración de variables enteras  
 Declaración de la cadena que recibe el comando original.  
 Declaración de la cadena que recibe la respuesta del controlador.  
 Declaración de la cadena que contiene el comando que se transmite al controlador.  
 Declaración del puntero al archivo

**PROGRAMA**

```
void main()
{
```

```
fp=fopen("d:\comando1.txt","r");
```

Abre el archivo que contiene el comando original de Intouch

```
for(i=0;i<21;i++)
fread(&buf[i],sizeof(char),1,fp);
```

Lee el comando original del archivo comando1.txt

```
fclose(fp);
sio_init();
sio_ioctl(port,B9600, CS7|Stop2|ParityEven);
sio_open(port);
sio_flush(port,2);
```

Cierra el archivo comando1.txt  
 Lee el archivo de configuración de la tarjeta  
 Configura el puerto para la comunicación  
 Habilita el puerto para recibir/transmitir  
 Limpia el buffer de entrada y salida de datos

```
for(i=0;i<20;i++)
{
if (buf[i]!=10)
buf3[i]=buf[i];
else{
buf3[i]=13;
abajo=i;
i=20;
};
};
```

Transmite el caracter 01 H ASCII

```
sio_putch(port, 1);
for (i=1 ; i<=abajo; i++)
sio_putch(port,buf3[i]);
```

Transmite la cadena con el comando

```
for (i=0;i<60;i++)for (j=0;j<30000;j++);
sio_read(port,buf1,lener2);
sio_flush(port,2);
sio_close(port);
fp=fopen("d:\comando2.txt","w+");

fwrite(buf1,lener2,1,fp);

fcloseall();}
```

Lee la cadena del buffer de entrada de datos  
Limpia el buffer de entrada y salida de datos  
Deshabilita el puerto para recibir/transmitir  
Crea si no existe un archivo para leer y escribir  
llamado comando2.txt  
Escribe la cadena con formato CSV en el  
archivo comando2.txt  
Cierra todo los archivos que hayan sido abiertos

Los anteriores programas son para la estación de carga B1; sin embargo estos son iguales para todas las estaciones de carga cambiando solamente el puerto por el cual se transmite y recibe la información.

---

## GLOSARIO

- ▣ **Actuador:** Es un elemento físico que convierte una señal eléctrica de la salida del sistema en una acción física, por ejemplo: válvulas de control de flujo, motores de velocidad variable, etc.
- ▣ **API:** American Petroleum Institute: Instituto Americano del Petróleo.
- ▣ **ASCII:** (American Standard Code for Information Interchange) Código americano estándar para el intercambio de información.
- ▣ **BS&W:** Porcentaje de Barro, Sal y Agua.
- ▣ **Bucle de Control:** Se dice que el proceso tiene un bucle de control cuando existe retroalimentación en el sistema. También se le conoce como lazo de control, este puede ser local o remoto.
- ▣ **Buffer:** Memoria intermedia. Es una porción reservada de la memoria que se utiliza para almacenar datos mientras son procesados.
- ▣ **Bus de Comunicación:** Es una vía o canal sobre la cual se transfieren datos entre dispositivos remotos.
- ▣ **Bus local:** Es un canal o ruta común entre dispositivos del Hardware.
- ▣ **Clepsidra:** Reloj de agua, muy usado en la antigüedad clásica y en la Edad Media.
- ▣ **Conmutador:** Dispositivo mecánico o electrónico que comanda el flujo de señales eléctricas de una línea a otra.
- ▣ **Consola de Operación:** Se refiere al equipo de cómputo que integradamente con los paquetes de software desempeñan el papel de interfaz entre el usuario y el proceso.
- ▣ **Control Numérico:** Es un método de control de movimiento de los componentes de máquinas utilizando números. El control numérico puede controlar el movimiento de una cabeza cortante por medio de información binaria contenida en un disco. El



control numérico es usado en tareas industriales, tales como fresado, torneado, matizado y perforación.

- ☐ **Convertidor A/D** : Dispositivo que convierte señales analógicas a señales digitales.
- ☐ **Convertidor D/A**: Dispositivo que convierte señales digitales a señales analógicas.
- ☐ **CPU**: (Control Processing Unit) unidad central de proceso. Es la parte de una computadora que realiza la computación.
- ☐ **CSV**: (Comma Separated Value) valor separador por comas.
- ☐ **CRT**: (Cathode Ray Tube) tubo de rayos catódicos. El tubo de vacío utilizado como pantalla de presentación en una terminal de video o en TV. El término se utiliza con frecuencia para referirse a la terminal en sí misma.
- ☐ **Data Highway**: Protocolo de comunicación entre equipos de control.
- ☐ **DCS**: (Distributed Control System) sistema de control distribuido. Sistema de control en el proceso que utiliza computadoras desdobladas en la línea de fabricación para su control.
- ☐ **Driver**: Es una rutina de programa que contiene las instrucciones necesarias para controlar la operación de un dispositivo periférico.
- ☐ **GUI**: (Graphic User Interface) Interfaz gráfica de usuario basada en gráficos.
- ☐ **Máquina Virtual**: Es una computadora que utiliza memoria virtual.
- ☐ **Megabits/seg**: 1,000,000 ó 1,048,576 bytes o caracteres sobre segundo.
- ☐ **Memoria RAM**: (Random Access Memory) memoria de acceso aleatorio.
- ☐ **Memoria Virtual**: Es una técnica que simula más memoria que la que realmente existe y permite a la computadora ejecutar varios programas simultáneamente, sin importar su tamaño.

- ☐ **Modem:** Modulador-Demodulador. Es un dispositivo que adapta una terminal o computadora a una línea telefónica. Convierte los pulsos digitales de la computadora a frecuencias dentro del rango del audio del teléfono y los vuelve a convertir en pulsos en el lado posterior.
- ☐ **Multibus:** Una arquitectura de bus de Intel que se usa en una variedad de aplicaciones industriales, militares y aeroespaciales. Se considera un bus avanzado debido a sus características, tales como pase de mensajes, autoconfiguración e interrupciones de software. MULTIBUS I es un bus de 16 bits y MULTIBUS II es una arquitectura de 32 bits.
- ☐ **Multimaster:** Es un diseño de bus que permite que las tarjetas adicionales procesen independientemente de la CPU, y sean capaces de acceder a la memoria de la computadora y sus periféricos por su cuenta.
- ☐ **Multiprogramación:** Ejecución de dos o más programas de computadora al mismo tiempo. Actualmente significa lo mismo que multitarea.
- ☐ **Multitarea:** Tradicionalmente significaba ejecutar dos o más tareas dentro del mismo programa al mismo tiempo. Hoy, es la ejecución de dos o más programas en una computadora al mismo tiempo. La multitarea se realiza debido a las diferencias de velocidad de entrada/salida y procesamiento.
- ☐ **PLC:** (Programmable Logic Controller) controlador lógico programable. Computadora utilizada en aplicaciones de control de procesos.
- ☐ **PROM:** (Programmable Read Only Memory) memoria programable de solo lectura.
- ☐ **Protocolo:** Conjunto de normas y regulaciones que gobiernan la transmisión y recepción de datos.
- ☐ **Redundancia:** Repetición de la misma información.
- ☐ **Relé de Potencia:** Receptor que por la acción de la corriente de línea cierra el generador local para enviar su corriente reforzada a una nueva línea, pero con más corriente.

- 📖 **Scripts:** Es un programa de computadora que condiciona la ejecución del software de la interfaz de operación Intouch.
- 📖 **Servomecanismo:** Sistema de control retroalimentado en el que la salida es algún elemento mecánico, sea posición, velocidad o aceleración.
- 📖 **Sistema:** Conjunto de objetos físicos o abstractos que ordenadamente relacionadas entre sí contribuyen a lograr un resultado o propósito común.
- 📖 **Sistema Determinístico:** Es un sistema de control en el que la respuesta a la entrada es predecible y repetible.
- 📖 **Sistema Estocástico:** Es un sistema de control en el que la respuesta a la entrada no es predecible y no repetible.
- 📖 **Sistema no Lineal:** Es un sistema en el que la respuesta a la entrada no es constante, por lo tanto no produce una señal lineal.
- 📖 **Tags:** Nombre asignado a una variable perteneciente a la base de datos de la interfaz de operación Intouch.
- 📖 **Transductor:** Dispositivo que convierte una forma de energía en otra. En aplicaciones de control de procesos, se emplea un transductor para convertir la presión en una lectura eléctrica.
- 📖 **Umbral:** Valor de una magnitud por debajo del cual no es perceptible o no ocurre un fenómeno.
- 📖 **Viscosidad:** Propiedad de los fluidos que se gradúa por la velocidad de salida de aquéllos al traves de tubos.

---

## BIBLIOGRAFIA

- ☐ Andrew S. Tanenbaum  
"Sistemas Operativos: Diseño e Implementación"  
Ed. Prentice Hall  
México, 1988  
471 pags.
- ☐ Astrom Karl J.  
"Sistemas Controlados por Computador"  
Ed. Paraninfo  
Madrid, 1988  
498 pags.
- ☐ Bollinger G. Jhon  
"Computer Control of Machine and Process"  
Ed. Adison. Wesley publishing Company Inc.  
Estados Unidos, 1989  
613 pags.
- ☐ Bryan G.T.  
"Control Systems for Technicians"  
Ed. Stoughton  
Inglaterra, 1979  
320 pags.
- ☐ Canales Ruiz Roberto  
"Análisis de Sistemas Dinámicos y Control Automático"  
Ed. Limusa  
México, 1980  
517 pags.
- ☐ Chesmond C.J.  
"Control System Technology"  
Ed. Eduard Arnold  
Londres, 1984  
465 pags.
- ☐ De Laurent T. Ami  
"Automation Systems for Control and Data Adquisición"  
Ed. ISA  
Estados Unidos, 1992  
229 pags.

- ☐ Faison Ted  
"Borland C++ 3.1 Programación orientada a Objetos"  
Ed. Prentice Hall  
México, 1993  
1335 pags.
- ☐ Freedman Alan  
"Diccionario de Computación"  
Ed. Mc Graw Hill  
México, 1994  
934 pags.
- ☐ Hostetter G.  
"Sistemas de Control"  
Ed. Mc Graw Hill  
México, 1990  
584 pags.
- ☐ Howard L. Harrison  
"Controles Automaticos"  
Ed. Trillas  
México, 1985  
582 pags
- ☐ Jovic Fran  
"Process Control Systems (Principles of Design and Operation)"  
Ed. Gulf Publisher Company  
Londres, 1986  
464 pags.
- ☐ Nó José y Angulo José María  
"Control de Procesos Industriales por Computadora"  
Ed. Paraninfo  
España, 1988  
175 pags.
- ☐ Pfaffenberger, Bryan, Ph. D  
"Diccionario para usuarios de computadoras"  
Ed. Prentice Hall  
México, 1992  
378 pags.
- ☐ Schildt Herbert  
"Programación en Turbo C"  
Ed. Mc Graw Hill  
México, 1990  
382 pags.

BIBLIOGRAFIA

---

☐ Schimilt Neil M. y Farwell Robert F.  
"A Fondo: Robotica y Sistemas Automáticos"  
Ed. Anaya  
México, 1989  
138 pags.

☐ Schmitt Neil M  
"Robotica y Sistemas Automáticos"  
Ed. Anaya Multimedia S.A.  
Madrid, 1988  
327 pags.

☐ Vandorent Arnold  
"Data Acquisition Systems"  
Ed. Prentice Hall  
Estados Unidos, 1982  
291 pags.