

**SISTEMA PARA EL REPORTE DE FALLAS EN LA SECCIÓN ENVASE DE  
BAVARIA CERVECERÍA DE BOYACÁ.**

**GLORIA JUDITH PALACIO OSORIO  
SANDRA CRISTINA TORRES SANABRIA**

**UNIVERSIDAD INDUSTRIAL DE SANTANDER  
FACULTAD DE INGENIERÍAS FISICOMECÁNICAS  
ESCUELA DE INGENIERÍA ELÉCTRICA, ELECTRÓNICA Y TELECOMUNICACIONES  
BUCARAMANGA  
2005**

**SISTEMA PARA EL REPORTE DE FALLAS EN LA SECCIÓN ENVASE DE BAVARIA  
CERVECERÍA DE BOYACÁ.**

**SANDRA CRISTINA TORRES SANABRIA  
GLORIA JUDITH PALACIO OSORIO**

**Proyecto de grado para optar por el título de Ingeniero Electrónico**

**Director**

**MSC. JORGE H RAMÓN SUÁREZ**

**Codirector**

**ING. EDGAR FERNANDO PUERTO CORREDOR**

**BAVARIA CERVECERÍA DE BOYACÁ.**

**UNIVERSIDAD INDUSTRIAL DE SANTANDER  
FACULTAD DE INGENIERÍAS FISICOMECÁNICAS  
ESCUELA DE INGENIERÍA ELÉCTRICA, ELECTRÓNICA Y TELECOMUNICACIONES  
BUCARAMANGA  
2005**

*Este trabajo de grado esta dedicado con el amor y afectos más profundos a Dios a mis padres Gloria y Jorge, a mis hermanos, Jorge Eduardo, Ingrid Mauree y Silvia Juliana, a mis sobrinos que con su amor, sabiduría y fortaleza llegué a ser la mujer que soy.*

*A María Paula quien llena mi vida de alegría.*

*A Omar Leonardo por su amor, ternura y entrega incondicional en cada instante.*

*A Javier Ricardo y Freddy por su especial colaboración en la realización de este proyecto.*

*GLORIA JUDITH PALACIO OSORIO*

*Primero quiero agradecer a Dios por darme la paciencia, fortaleza y sabiduría durante el transcurso de mi formación profesional.*

*A mi abuelita que desde el cielo me cuida, ayuda y protege.*

*A mis padres por su amor, confianza y apoyo incondicional en cada instante de mi vida.*

*A mi hija María Valentina por su alegría y porque es mi fuente de inspiración.*

*A Juan Guillermo por su amor, ternura, fortaleza, apoyo y confianza, a pesar de las dificultades.*

*A Freddy y Javier por su orientación durante la realización de este proyecto.*

**SANDRA CRISTINA TORRES SANABRIA**

## TABLA DE CONTENIDO

<b>INTRODUCCIÓN .....</b>	<b>1</b>
<b>1. ANTECEDENTES .....</b>	<b>3</b>
1.1. MANTENIMIENTO .....	3
1.1.1. Organización del Mantenimiento .....	4
1.1.2. Objetivos del Mantenimiento .....	5
1.1.3. Tipos de Mantenimiento.....	5
➤ Mantenimiento Predictivo .....	5
➤ Mantenimiento Preventivo .....	6
➤ Mantenimiento Programado .....	7
➤ Mantenimiento Correctivo .....	7
1.2. FALLA .....	8
1.2.1. Tipos de falla de acuerdo a su gravedad .....	9
1.2.2. Tipos de falla de acuerdo a su frecuencia de ocurrencia .....	9
1.2.3. Causa de Falla.....	9
➤ Causas Básicas .....	9
➤ Causas Inmediatas .....	10
1.2.4. Modo de Falla .....	10
1.3. DESCRIPCIÓN DEL REPORTE DE FALLAS EN LA SECCION ENVASE DE BAVARIA CERVECERIA DE BOYACÁ .....	10
<b>2. DISEÑO E IMPLEMENTACIÓN DEL SISTEMA .....</b>	<b>19</b>
2.1. PROTOTIPO .....	20
2.1.1. Operación .....	20
2.1.2. Diseño de la Tarjeta .....	21
2.1.3. Componentes de la Tarjeta .....	23
➤ Microcontrolador.....	23
➤ Memoria Externa.....	27
➤ Transceiver.....	28
➤ Interfaz de usuario .....	30
2.2. COMUNICACIÓN SERIAL VÍA RS-485 .....	31
2.2.1. Forma de Conexión .....	32
2.2.2. Protocolo .....	34
➤ Formato de las tramas .....	34
➤ Funcionamiento de la comunicación .....	37
2.3. BASE DE DATOS .....	40
2.3.1 Estructura de la Base De Datos .....	40
➤ Tabla Máquinas.....	41
➤ Tabla Módulos .....	42

➤	Tabla Partes.....	43
➤	Tabla Fallas.....	44
➤	Tabla Causas de Falla.....	45
➤	Tabla Operarios.....	46
➤	Tablas Últimos Reportes y Últimos Reportes SR.....	47
➤	Reportes Pendientes .....	48
<b>3.</b>	<b>SOFTWARE DEL SISTEMA.....</b>	<b>49</b>
3.1.	PROGRAMACIÓN DE LA INTERFAZ DE USUARIO EN VISUAL BASIC .....	49
3.1.1	<i>Acceso a la Base de Datos.....</i>	51
➤	Acceso y Visualización de las Tablas de la Base de Datos.....	52
3.1.2	<i>Manejo Dinámico de los Datos .....</i>	53
➤	Función Agregar un Dato Y Guardarlo en la Base de Datos.....	55
➤	Función Editar un Registro.....	56
➤	Función Eliminar un Registro.....	57
3.1.3.	<i>Comunicación con el Prototipo.....</i>	58
3.1.4.	<i>Visualización Reportes Almacenados.....</i>	59
3.1.5	<i>Impresión de Reportes.....</i>	61
3.2.	PROGRAMACIÓN DEL MICROCONTROLADOR.....	63
3.2.1.	<i>Algoritmo general del programa.....</i>	63
3.2.2.	<i>Subrutinas Especiales del Programa Principal.....</i>	67
➤	Configuración de Registros Especiales.....	67
➤	Rutina de Lectura y Escritura de la Memoria.....	72
<b>4.</b>	<b>CONCLUSIONES .....</b>	<b>79</b>
	<b>BIBLIOGRAFIA .....</b>	<b>81</b>
	<b>ANEXO A ESTANDAR RS 485 .....</b>	<b>83</b>
	<b>ANEXO B ADAPTADOR DE INTERFAZ ESTÁNDAR RS-485 A RS-232 .....</b>	<b>85</b>
	<b>ANEXO C. MANUAL DE USUARIO PARA LA INTERFAZ DE VISUAL BASIC 6.0.....</b>	<b>88</b>
	<b>ANEXO D.MANUAL DE USUARIO DEL EQUIPO.....</b>	<b>94</b>
	<b>ANEXO E. HOJAS DE DATOS .....</b>	<b>107</b>

## LISTA DE FIGURAS

FIGURA 1. PLANILLA DE REPORTE DE MANTENIMIENTO.....	11
FIGURA 2. ORDEN DE TRABAJO.....	13
FIGURA 3. ORDEN DE TRABAJO EXPEDIDA EN SAP-PM .....	14
FIGURA 4. PLANTILLA MENSUAL .....	17
FIGURA 5. ESQUEMA GENERAL DEL SISTEMA PARA REPORTE DE FALLAS. ....	19
FIGURA 6. INTERFAZ GRÁFICA EN VISUAL BASIC.....	21
FIGURA 7. ESQUEMÁTICO DE LA TARJETA DEL PROTOTIPO.....	22
FIGURA 8. ASIGNACIÓN DE PINES DEL MICROCONTROLADOR <i>MC68HC908GP32</i> , Y SU USO EN LA TARJETA. ....	24
FIGURA 9. DISTRIBUCIÓN DE PINES DE LA MEMORIA <i>EEPROM 24FC256</i> .....	28
FIGURA 10. DIAGRAMA LÓGICO (LÓGICA POSITIVA) Y DISTRIBUCIÓN DE LOS PINES DEL CIRCUITO INTEGRADO SN75179B.....	28
FIGURA 11. FOTO DEL PROTOTIPO .....	30
FIGURA 12. DIAGRAMA DE LA RED PARA CONECTAR LOS PROTOTIPOS Y EL PC.....	31
FIGURA 13. DIBUJO DE LA BORNERA DE SALIDA EN EL PROTOTIPO .....	32
FIGURA 14. CONFIGURACIÓN TÍPICA MULTIPUNTO DE 4 HILOS PARA RS-485 .....	33
FIGURA 15. TRAMA DE REPORTE DE FALLO.....	35
FIGURA 16. TRAMA DE FECHA Y HORA .....	36
FIGURA 17. FORMATO DEL NÚMERO DE REGISTROS ALMACENADOS. ....	37
FIGURA 18. FUNCIONAMIENTO DEL PROTOCOLO .....	39
FIGURA 19. MODELO RELACIONAL DISEÑADO PARA LA BASE DE DATOS ELABORADA.....	41
FIGURA 20. ESTRUCTURA TABLA MÁQUINAS.....	42
FIGURA 21. ESTRUCTURA TABLA MÓDULOS .....	43
FIGURA 22. ESTRUCTURA TABLA PARTES .....	44
FIGURA 23. ESTRUCTURA TABLA FALLAS .....	45
FIGURA 24. ESTRUCTURA TABLA CAUSAS DE FALLAS .....	46
FIGURA 25. ESTRUCTURA TABLA OPERARIOS.....	47
FIGURA 26. ESTRUCTURA TABLAS ÚLTIMOS REPORTES Y ÚLTIMOS REPORTES SR.....	48
FIGURA 27. ESTRUCTURA TABLA REPORTES PENDIENTES .....	48
FIGURA 28 INTERFAZ DE USUARIO .....	50
FIGURA 29. CONTROL DATA .....	51
FIGURA 30. DIAGRAMA DE FLUJO CORRESPONDIENTE AL ACCESO A LA BASE DE DATOS.....	52
FIGURA 31. DIAGRAMA DE FLUJO PARA EL ACCESO Y VISUALIZACIÓN DE LAS TABLAS DE LA BASE DE DATOS..	53
FIGURA 32. BARRA DE HERRAMIENTAS PARA MANEJO DINÁMICO DE LOS DATOS.....	54
FIGURA 33. DIAGRAMA DE FLUJO PARA MANEJO DINÁMICO DE LOS DATOS .....	54
FIGURA 34. DIAGRAMA DE FLUJO AGREGAR Y GUARDAR .....	55
FIGURA 35. DIAGRAMA DE FLUJO EDITAR .....	56
FIGURA 36. DIAGRAMA DE FLUJO ELIMINAR .....	57
FIGURA 37. CONTROL MSCOMM .....	58
FIGURA 38. DIAGRAMA DE FLUJO PARA COMUNICACIÓN SERIAL .....	59
FIGURA 39. DIAGRAMA DE FLUJO VISUALIZACIÓN DE REPORTES. ....	60
FIGURA 40. DETERMINACIÓN DE LAS PROPIEDADES PARA EL OBJETO <i>PRINTER</i> .....	61
FIGURA 41. ETIQUETA QUE INFORMA LA CANTIDAD DE REPORTES PENDIENTES PARA IMPRESIÓN .....	61
FIGURA 42. DIAGRAMA DE FLUJO IMPRIMIR REPORTES.....	62
FIGURA 43. ALGORITMO GENERAL DE CONFIGURACIÓN Y FUNCIONAMIENTO DEL PROTOTIPO .....	64
FIGURA 44. REGISTROS INTKBSCR Y INTKBIER .....	68
FIGURA 45. FORMATO DE DATOS SCI PARA DATOS DE 8 BITS.....	69

FIGURA 46. REGISTRO SCBR ( <i>SCI BAUD RATE REGISTER</i> ) Y TABLA DE DIVISORES.....	70
FIGURA 47. REGISTROS SCC1, SCC2 Y SCDR .....	71
FIGURA 48. REGISTRO TBCR.....	72
FIGURA 49. SECUENCIA DE TRANSFERENCIA DE DATOS EN EL BUS SERIAL.....	73
FIGURA 50. BYTE DE CONTROL.....	74
FIGURA 51. ASIGNAMIENTO DE LA SECUENCIA DE LOS BITS DE DIRECCIÓN.....	75
FIGURA 52. ESCRITURA SECUENCIAL PARA PÁGINAS DE 6 BYTES.....	75
FIGURA 53. LECTURA.....	76
FIGURA 54. DISTRIBUCIÓN DE LA MEMORIA .....	77
FIGURA 55. PARTE FINAL DE LA DISTRIBUCIÓN DE LA MEMORIA .....	78

## **LISTA DE TABLAS**

TABLA 1. NIVELES LÓGICOS Y DE TENSIÓN PARA EL CIRCUITO INTEGRADO SN75179B.....	29
TABLA 2. FORMATO DE TRAMA DEL REPORTE DE FALLO.....	35
TABLA 3. FORMATO DE FECHA Y HORA.....	36
TABLA 4. CONVENCIONES DE LOS NOMBRES DE LOS PINES .....	69
TABLA 5. SELECCIÓN DE LA TASA DE BAUDIOS .....	70

## RESUMEN

### TITULO.

SISTEMA PARA EL REPORTE DE FALLAS EN LA SECCIÓN ENVASE DE BAVARIA CERVECERÍA DE BOYACÁ\*.

### AUTORES.

PALACIO OSORIO, Gloria Judith y TORRES SANABRIA, Sandra Cristina \*\*

### PALABRAS CLAVES

Reporte de Fallo, Mantenimiento, Mantenimiento Correctivo Falla, Estándar RS-485, Base de Datos.

### DESCRIPCION

En el presente trabajo, se describe el diseño y construcción de un sistema que permite generar, transmitir y almacenar los reportes de fallos encontrados por los operarios en el tren de envase número dos de la sección envase de Bavaria Cervecería de Boyacá.

En primera instancia, se realiza una breve introducción acerca de la importancia del mantenimiento en la industria, la cual incluye la necesidad de analizar y procesar la información obtenida en la recolección de los reportes, para generar estrategias que tiendan a mejorar la calidad, rendimiento y productividad de la empresa. En segundo lugar, se explica el proceso de diseño y construcción de los componentes del sistema para el reporte de fallo (Prototipo, Red, Base de Datos e Interfaz de Usuario), y se agregan los criterios de selección de las partes del mismo.

Como resultado se entrega un sistema que cuenta con una red industrial (RS485) de prototipos, desde los cuales se generan y almacenan hasta 999 reportes, a través de menús. Además cuenta con una Interfaz de usuario diseñada en Visual Basic, desde donde se establece la comunicación con el prototipo y con la cual se puede visualizar y manipular los reportes descargados en el PC por medio de una base de datos.

---

\* Práctica realizada en BAVARIA CERVECERÍA DE BOYACÁ.

\*\* Facultad de Ingenierías Físicomecánicas, Ingeniería Electrónica. Director: MSC. Jorge Hernando Ramón

## **ABSTRACT**

### **TITLE**

FAILURE REPORT SYSTEM FOR BAVARIA CERVECERÍA DE BOYACÁ'S BOTTLE SECTION\*.

### **AUTHORS**

PALACIO OSORIO, Gloria Judith y TORRES SANABRIA, Sandra Cristina \*\*

### **KEY WORDS**

Failure Report, Maintenance, Corrective Maintenance, Failure, Standard RS-485, Data Base.

### **DESCRIPTION**

In this document is described the design and construction of a system that allows to generate, transmit and store the failure reports that have been found by operators in the bottle train number two in the Bavaria Cervecería de Boyacá's bottle section.

On first step, it's presented a brief introduction about the importance of maintenance in industry, which includes the needing of analyze and processing information obtained in the reports collection in order to generate strategies intended to improve the enterprise's quality level, performance and productivity. On second step it's explained the design and construction process of the system's components to report the failure (Prototype, Network, Data Base and User's Interface), and selection parts criteria is added.

As a result it is given a system that includes an industrial network (RS485) of prototypes, which generates and stores until 999 reports through menus. Besides, it presents an user's Interface designed in Visual Basic, from the prototype's communication is established and lets to visualize and manipulate the reports downloaded in the PC by a data base.

---

\* Practice made in BAVARIA CERVECERÍA DE BOYACÁ.

\*\* Faculty of Engineering Physical-mechanics. Engineering Electronic, Director: MSC. Jorge Hernando Ramón

## INTRODUCCIÓN

Uno de los grandes retos que afronta la industria en este momento radica en poseer un completo sistema para la gestión de mantenimiento, pues mediante este se logra llegar a la creación de nuevas estrategias que tiendan tanto a predecir los posibles daños en las máquinas, como a disminuir la frecuencia con la cual se presentan las fallas que causan detención de la producción de las mismas; lo que conlleva a una mejora en la calidad, en el rendimiento y en la productividad de la empresa.

Otra ventaja que conlleva una buena gestión de Mantenimiento Preventivo y control de Mantenimiento Correctivo reside en que mediante los resultados obtenidos al efectuar los procesos de mantenimiento se consigue llevar y disponer de un histórico de cada máquina para tener más control sobre su rendimiento, producción y productividad.

Para apoyar los procesos de mantenimiento, críticos en la industria, se puede aprovechar la tecnología planteando soluciones innovadoras, como lo es la implementación de un prototipo electrónico, una red industrial y una aplicación de software; a fin de minimizar y optimizar tareas indispensables para el buen rendimiento de dichos procesos, logrando de esta forma responder oportuna y eficazmente a los diferentes problemas que se presenten.

Bavaria Cervecería de Boyacá decidió buscar una solución que supla las insuficiencias presentes en el procedimiento para crear los Reportes de los Fallos producidos en las máquinas instaladas en su Sección de Envase. Por lo tanto este Proyecto de Grado se realiza con el fin de crear un sistema que genere los informes de las fallas e inmediatamente los ingrese a una Base de Datos.

Este documento en su primera parte contiene los conceptos básicos sobre mantenimiento y sus diferentes clases, tipos de fallas, las formas en que se pueden presentar y las causas por las cuales se originan. También se hace una descripción del método que actualmente se emplea en la Sección Envase para reportar las averías

En la segunda parte se explican los elementos que en conjunto forman el sistema para el reporte de fallas. En primera instancia se describe como se diseño y construyó el prototipo destinado para crear los reportes de los daños presentados en las máquinas del Tren de Envase; en este subcapítulo se especifican los criterios de diseño, la forma como funciona y los componentes del equipo en mención. Luego se procede a explicar el tipo de red que se utilizó para la transmisión de los datos y el protocolo creado para tal fin. Finalmente en se describe el proceso de construcción y el contenido de cada una de las tablas que conforman la Base de Datos hecha en *Microsoft Access 2003*, donde se almacenan los reportes de fallo, con el fin de tener un histórico del funcionamiento de cada máquina instalada en el envase.

En la última parte del documento; se expone la programación de la interfaz de usuario creada en Visual Basic 6.0 que además de lograr establecer comunicación serial con el prototipo e implementar el protocolo que determina las condiciones para la transmisión de los datos; también permite al usuario visualizar y manejar los registros almacenados en las tablas de la Base de Datos e imprimir los reportes de fallo que ingresan a la misma. Luego se describe la programación realizada en el microcontrolador con el fin de establecer en él las funciones necesarias para editar los repotes, almacenarlos en la memoria del equipo y descargar dicha información al PC

## **1. ANTECEDENTES**

Para las personas que no están acostumbradas al lenguaje técnico concerniente a mantenimiento y reporte de fallos, así como al manejo que se le da a éste en la Sección Envase de Bavaria Cervecería de Boyacá, a continuación se expondrán algunos conceptos y terminologías básicas necesarias para entender el contexto en el cual se desarrolla el presente trabajo.

En ésta sección se tratarán temas que van desde lo general como lo es el concepto de Mantenimiento, hacia lo específico como la utilización de la Norma ISO14224<sup>1</sup>. También se trabajará con la información recolectada sobre las rutinas o procedimientos utilizados para el mantenimiento de las máquinas y cómo este trabajo puede ser usado para desarrollar estrategias de Mantenimiento Preventivo en la Sección Envase de Bavaria Cervecería de Boyacá.

### **1.1. MANTENIMIENTO**

Es evidente la exigencia que presenta una economía globalizada, mercados altamente competitivos y un contexto versátil donde la velocidad de cambio supera en gran medida nuestra capacidad de respuesta. En éste entorno estamos inmersos y es importante considerar algunas posibilidades que siempre han estado, pero que ahora cobran mayor relevancia.

Especialmente, la predominante necesidad de redimensionar la empresa, envuelve al Mantenimiento en metas y circunstancias que deben ser estudiadas.

---

<sup>1</sup> La Norma ISO14224 hace referencia a los procedimientos requeridos para realizar el Mantenimiento en la Industria.

Puesto que el ingreso de activos siempre procedía de la venta de un producto o servicio, esta perspectiva inicial llevó a la empresa a enfocar sus esfuerzos de mejoramiento, y con ello sus recursos, en función de la producción. Inicialmente el Mantenimiento fue un “problema” que surgió al querer producir secuencialmente, por lo tanto fue visto como un mal necesario, una función bajo el mando de la producción cuyo propósito era arreglar desperfectos en forma rápida y económica.

Actualmente, de acuerdo con la filosofía de la Calidad Total junto con sus tendencias que evidencian los requerimientos de la integración, compromiso, esfuerzo y optimización de todas las unidades que conforman a la empresa, se ha creado la necesidad de volcar la atención sobre el Mantenimiento, encontrando que éste incide en:

- ❖ Costos de Producción.
- ❖ Calidad del Producto – Servicio.
- ❖ Capacidad Operacional.
- ❖ Capacidad de Respuesta de la Empresa
- ❖ Seguridad e Higiene Industrial.
- ❖ Calidad de Vida de los Trabajadores de la Empresa
- ❖ Imagen y Seguridad Ambiental de la Empresa.

Debido a que se desprende de argumentos de tal peso, el Mantenimiento se puede definir como: La combinación de acciones administrativas y técnicas realizadas sobre los equipos instalados en una empresa con el fin de garantizar su correcto funcionamiento, para de esta forma disminuir la probabilidad de fallos en las máquinas, reparar las fallas producidas en las mismas, reducir los tiempos de parada que implican pérdidas tanto en la productividad como en la eficiencia de la empresa, evitar pérdidas humanas y satisfacer los requerimientos de calidad exigidos por el cliente de una forma rentable. Por lo tanto el Mantenimiento produce un bien real, que puede resumirse en capacidad de producir con calidad, seguridad y rentabilidad.

**1.1.1. Organización del Mantenimiento.** En este momento, se ha visto la necesidad de manejar desde el Mantenimiento una gran cantidad de información debido a:

- ❖ La exigencia de organizar adecuadamente este servicio con la introducción de programas de Mantenimiento Preventivo y el control del Mantenimiento Correctivo en base, fundamentalmente, del objetivo de optimizar la disponibilidad de los equipos productores.
- ❖ La necesidad de minimizar los costos propios de Mantenimiento, lo cual acentúa la inquietud de organización, mediante la introducción de controles adecuados de costos.
- ❖ El requerimiento de optimizar todos sus aspectos, tanto de costos, como de calidad, junto con cambio rápido de producto; lo que conduce a la necesidad de analizar de forma sistemática las mejoras que pueden ser introducidas en la gestión, tanto técnica como económica del Mantenimiento.

**1.1.2. Objetivos del Mantenimiento.** El Mantenimiento adecuado, tiende a prolongar la vida útil de los bienes, a obtener un rendimiento aceptable de los mismos y reducir el número de Fallas; por tal razón la organización e información del Mantenimiento debe estar encaminada a la permanente consecución de los siguientes objetivos:

- ❖ Evitar, reducir, y en su caso, reparar, las fallas sobre los bienes precipitados.
- ❖ Disminuir la gravedad de las Fallas que no se lleguen a evitar.
- ❖ Evitar detenciones inútiles o parada de máquinas.
- ❖ Evitar accidentes.
- ❖ Evitar incidentes y aumentar la seguridad para las personas.
- ❖ Conservar los bienes productivos en condiciones seguras y preestablecidas de operación.
- ❖ Balancear el costo de Mantenimiento con el correspondiente al lucro cesante.
- ❖ Alcanzar o prolongar la vida útil de los bienes.

### **1.1.3. Tipos de Mantenimiento.**

- **Mantenimiento Predictivo.** Este tipo de mantenimiento se basa en predecir la falla antes de que esta se produzca. La idea es conseguir adelantarse a la falla o

al momento en que el equipo o elemento deja de trabajar en sus condiciones óptimas. Para lograr esto se utilizan herramientas y técnicas de monitoreos de parámetros físicos.

❖ **Ventajas**

- ☑ La intervención en el equipo o cambio de un elemento.
- ☑ Obliga al operario a dominar el proceso y a tener datos técnicos, que obligará la intervención de un método científico de trabajo riguroso y objetivo.

❖ **Desventajas**

- ☒ La implantación de un sistema de este tipo requiere una inversión inicial importante, los equipos y los analizadores de vibraciones tienen un costo elevado.
- ☒ Se debe tener un personal que sea capaz de interpretar los datos que generan los equipos y tomar conclusiones en base de ellos, trabajo que requiere un conocimiento técnico elevado de la aplicación.
- ☒ Se recomienda la implantación de éste sistema en industrias donde los paros intempestivos ocasionan grandes pérdidas.

➤ **Mantenimiento Preventivo.** Es el conjunto de actividades que se realizan periódica y programadamente, utilizando criterios tales como el comportamiento registrado en el historial de los equipos, vida útil de los mismos y recomendaciones del fabricante. Dichas actividades se efectúan con el propósito de evitar posibles fallos en los equipos, reducir la cantidad de paradas de producción imprevistas y garantizar de esta forma el correcto y eficiente desempeño de la función para la cual fueron diseñadas las máquinas.

❖ **Ventajas**

- ☑ Se exige un conocimiento de las máquinas y un tratamiento de los históricos que ayudarán en gran medida a controlar la maquinaria e instalaciones.
- ☑ El cuidado periódico conlleva un estímulo óptimo de conservación con la que es indispensable una aplicación eficaz para contribuir a un correcto sistema de calidad.

- ☑ La reducción del Mantenimiento Correctivo representa una reducción de los costos de producción y un aumento de la disponibilidad, esto posibilita una planificación de los trabajos del departamento de mantenimiento.
- ☑ Se programa con el departamento de producción el mejor momento para el paro de los procesos de producción.

❖ **Desventajas**

- ☒ Representa una inversión inicial en infraestructura y mano de obra. El desarrollo de planes de mantenimiento se debe realizar por técnicos especializados.
- ☒ Al realizar un análisis incorrecto puede aumentar el costo del mantenimiento sin mejoras sustanciales.
- ☒ Los trabajos monótonos pueden disminuir la motivación en los operarios, por lo que es necesario crear sistemas imaginativos para convertir un trabajo repetitivo en un trabajo que genere satisfacción y compromiso, la implicación de los operarios es indispensable para el éxito del plan.

➤ **Mantenimiento Programado.** Grupo de acciones realizadas en los equipos después de analizar el resultado de las rutas de inspección o recorrido que efectúan los operarios de la planta con el fin de descubrir posibles fallas en las máquinas, para de esta forma proceder a tomar las decisiones necesarias encaminadas a reparar las fallas y organizar y expedir las *órdenes de trabajo*<sup>2</sup>.

➤ **Mantenimiento Correctivo.** Conjunto de actividades realizadas en determinado equipo y en el menor tiempo posible, cuando en éste se presenta una falla que no estaba prevista y que afecta la realización de la labor para la cual fue diseñado. Dentro de este tipo de mantenimiento se pueden contemplar dos tipos de enfoques:

---

<sup>2</sup> Orden de Trabajo es una planilla que estructura todos los procedimientos concernientes al mantenimiento en un equipo de la planta.

- ❖ Mantenimiento paliativo o de campo: Encargado de la reposición del funcionamiento, aunque no quede eliminada la fuente que provocó la falla.
- ❖ Mantenimiento curativo: Encargado de la reparación propiamente pero eliminando las causas que han producido la falla.
  
- ❖ **Ventajas**
- ☑ Al estar preparado el equipo la intervención en el fallo es rápida y la reposición en la mayoría de los casos será con el mínimo tiempo.
- ☑ Se requiere de una infraestructura sencilla y un grupo de operarios competentes, por lo tanto el costo de mano de obra será el mínimo; en donde predomina la experiencia y pericia de los operarios.
- ☑ Rentable en equipos que no intervienen de manera instantánea en la producción.
  
- ❖ **Desventajas**
- ☑ Se producen paradas y daños imprevisibles en la producción que afectan a la planificación de manera incontrolada.
- ☑ Se suelen producir reparaciones de baja calidad debido a la rapidez en la intervención y a la prioridad de reponer antes que reparar definitivamente.

## 1.2. FALLA

Se dice que algo falla cuando deja de prestar el servicio que debía dar o cuando aparecen efectos indeseables, según las especificaciones de diseño con las que fue construido o instalado el bien en cuestión, por tal razón una falla se puede definir como: Avería presentada en un equipo que causa la perturbación en el desempeño de la actividad o en la prestación del servicio para el cual esta destinado; trae consecuencias como: disminución de la productividad, lesiones humanas, pérdidas de materiales, algunas veces daño ecológico y por supuesto pérdidas económicas. Las fallas se pueden clasificar de varias formas de acuerdo a su gravedad y a la frecuencia de ocurrencia.

**1.2.1. Tipos de falla de acuerdo a su gravedad.** Según la severidad del daño presentado en la máquina, las fallas se pueden clasificar en.

- ❖ **Falla Crítica:** Daño presentado en una máquina que trae como consecuencia la terminación inmediata de su labor.
- ❖ **Falla no Crítica:** Daño presentado en una máquina que no causa la terminación inmediata su de la labor

**1.2.2 Tipos de falla de acuerdo a su frecuencia de ocurrencia**

- ❖ **Esporádicas:** Son una desviación por fuera de un rango aceptable de operación normal. Generalmente, la eliminación de una falla esporádica solamente llevará las cosas de nuevo a un rango aceptable. Estos son eventos por lo general poco frecuentes y no relacionados entre sí.
- ❖ **Crónicas:** Son una desviación dentro de un rango aceptable de operación normal. Estos son eventos relativamente frecuentes. La eliminación de fallas crónicas llevará las operaciones regulares al punto máximo de una operación normal aceptable y elevará el nivel promedio esperado del desempeño.

**1.2.3. Causa de Falla.** Circunstancia que origina que un equipo no cumpla la función para la cual esta predestinado. Las causas de falla pueden ser:

➤ **Causas Básicas**

- ❖ **Diseño:** Comprendida por errores en características iniciales de fabricación como Capacidad inadecuada, Material inadecuado, Diseño inadecuado.
- ❖ **Fabricación / Instalación:** Error de Fabricación o Error de Instalación.
- ❖ **Operación / Mantenimiento:** Error de Operación, Error de Mantenimiento, Servicio fuera de diseño, Desgaste o rotura esperada.
- ❖ **Administración:** Error de documentación, Error Administrativo.

- ❖ Desconocidas: Errores por causas aleatorias.

➤ **Causas Inmediatas**

- ❖ Mecánicas: Las cuales están comprendidas por: Fuga, Vibración, Desalineamiento, Deformación, Suelto, Pegado
- ❖ Material: Cavitación<sup>3</sup>, Corrosión, Erosión, Desgaste, Fractura, Fatiga, Sobrecalentamiento, Quemado
- ❖ Instrumentos: Control, No Señal / indicador / alarma, Señal / indicador / alarma incorrecta, Software, Falla común (falla redundante).
- ❖ Eléctricas: Corto circuito, Circuito abierto, Sin /bajo energía / voltaje, Falla de energía voltaje, Falla tierra / aislamiento
- ❖ Influencia externa: Bloqueado / Taponado, Contaminación
- ❖ Misceláneos: Misceláneas, desconocidas.

**1.2.4. Modo de Falla.** Forma como se presenta la avería, o deficiencia observada en el equipo por el operario que la reportó.

### **1.3. DESCRIPCIÓN DEL REPORTE DE FALLAS EN LA SECCION ENVASE DE BAVARIA CERVECERIA DE BOYACÁ**

En la Sección Envase de Bavaria Cervecería de Boyacá el procedimiento empleado para reportar y reparar las fallas imprevistas producidas en una máquina (ejecución de mantenimiento correctivo), se fundamenta en la tramitación manual de una Planilla (ver figura 1), cuando se detecta y arregla la avería, la cual es diligenciada por el operario de turno.

---

<sup>3</sup> Formación de burbujas de vapor en corrientes fluidas

Con la información recolectada en el reporte el Ingeniero encargado genera la Orden de Trabajo (ver figura 2) mediante el software de mantenimiento instalado en la planta “SAP PM<sup>4</sup>” (ver figura 3) para de esta forma asignar la labor de reparación al técnico encargado, así como la elaboración del reporte de mantenimiento mensual.

**Figura 1.**Planilla de Reporte de Mantenimiento.

a. Parte delantera

	<b>PLANTILLA DE REPORTES DE MANTENIMIENTO</b>			<b>65606207</b>		
	ACTUALIZACIÓN 05					
TREN DE ENVASE N° _____		FECHA _____		Vo. Bo. INGENIERO _____		
<b>SUPERVISOR</b> SR _____	<b>MECÁNICOS</b> SR _____	<b>ELECTRICISTA</b> SR _____	<b>LUBRICADOR</b> SR _____	<b>TURNO</b> SR _____		
<b>LECTURA DE CONSUMO ACUMULADO DE AGUA (m<sup>3</sup>)</b>						
<b>LAVADORA</b>	<b>PASTERIZADORA</b>	<b>CONTADOR PRINCIPAL</b>	<b>HORA DE MEDICIÓN</b>			
<b>HORA</b>	<b>PRODUCCIÓN</b>	<b>ITEM</b>	<b>DESCRIPCIÓN DE TRABAJOS REALIZADOS</b>	<b>T. MEC</b>	<b>T. ELEC</b>	<b>OTRO</b>
	DECENAS/HORA					
	ESTIBAS/HORA					
	ACUMULADO ESTIBAS					
<b>HORA</b>	<b>PRODUCCIÓN</b>	<b>ITEM</b>	<b>DESCRIPCIÓN DE TRABAJOS REALIZADOS</b>	<b>T. MEC</b>	<b>T. ELEC</b>	<b>OTRO</b>
	DECENAS/HORA					
	ESTIBAS/HORA					
	ACUMULADO ESTIBAS					
<b>HORA</b>	<b>PRODUCCIÓN</b>	<b>ITEM</b>	<b>DESCRIPCIÓN DE TRABAJOS REALIZADOS</b>	<b>T. MEC</b>	<b>T. ELEC</b>	<b>OTRO</b>
	DECENAS/HORA					
	ESTIBAS/HORA					
	ACUMULADO ESTIBAS					


<sup>4</sup>SAP (Systems, Applications, Products in Data Processing) es un sistema informático compuesto por módulos integrados que abarcan todos los aspectos de la administración industrial.

b. Parte trasera.

	<b>PLANTILLA DE REPORTES DE MANTENIMIENTO</b>		<b>65606207</b>		
			ACTUALIZACIÓN 05		
SE DEBEN PROGRAMAR PARA MANTENIMIENTO LOS SIGUIENTES TRABAJOS: ..... ..... ..... ..... .....					
<b>CON EL OBJETO DE OPTIMIZAR EL FUNCIONAMIENTO DE LOS EQUIPOS, SOLICITAMOS SU PARTICIPACIÓN PARA INDICAR LOS PUNTOS A MEJORAR EN LA SUPERVISIÓN Y OPERACIÓN DEL ENVASE EN SU TURNO</b> ..... ..... .....					
<b>REPORTE DE LUBRICACIÓN EN EL TURNO</b>					
<b>EQUIPO</b>		<b>TRABAJO A REALIZAR</b>			
.....		.....			
.....		.....			
.....		.....			
		<b>GUARDAS INSTALADAS SEGÚN NORMA</b>		<b>ACCIÓN TOMADA EN CASO DE NO ESTAR SEGÚN NORMA ALGUNA GUARDA</b>	<b>RESPONSABLE</b>
		SI	NO		
DEPALETIZADORA					
TRANSCANASTAS					
DESEMPACADORA					
TRANSBOTELLAS					
LAVADORA					
PALETIZADORA					

Fuente: Sección de Mantenimiento de Bavaria Cervecería se Boyacá.

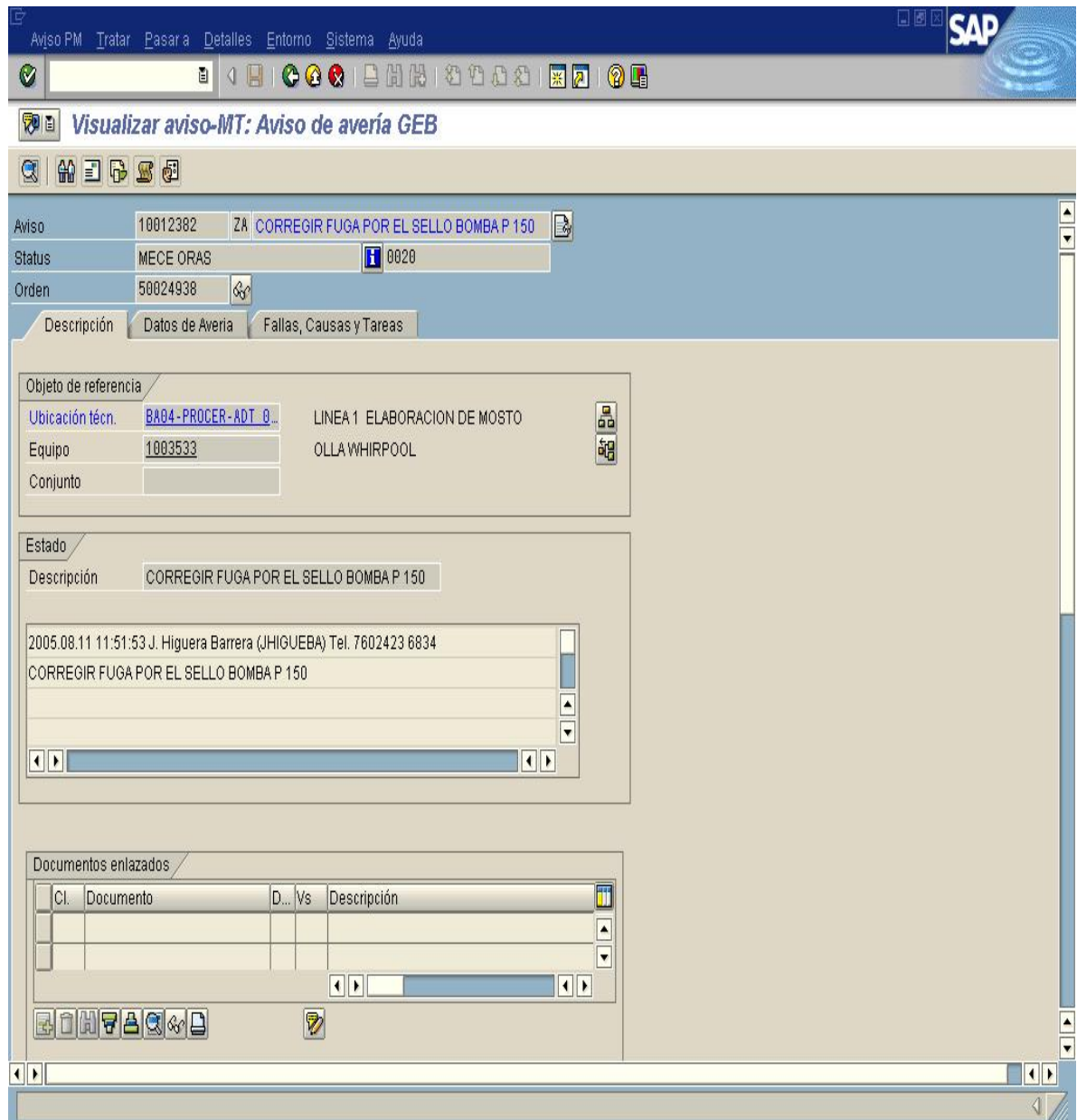
**Figura 2. Orden de Trabajo.**

									
<b>ORDEN DE TRABAJO</b>									
<b>Orden de Trabajo N°</b>	<b>5000202057</b>			<b>Reserva</b>	0000134485		Fecha: 2005.06.04		
<b>Planta</b>	<b>BA04</b>			<b>Orden interna</b>	0850000123		Prioridad 2		
<b>Responsable OT</b>	<b>FABIO TORRES</b>								
Descripción del trabajo: EFECTUAR REVISIÓN A CONEXIÓN DE TUBERÍA Texto ampliado del trabajo: SALIDA DE ACIDO OT GENERA SEGÚN RUTA DE E. Z. MAYO									
Clase de orden: ZOT3 OT DE MANTENIMIENTO PROGRAMADO GEB Código Actividad Mtto: ZO2 REPARAR Código puesto trabajo resp: INGMTTO INGENIERO MANTENIMIENTO Código Ubicación técnica: BA04-PROCER-ADT 04-SISTIMSANE SISTEMA DE LIMPIEZA Y SANEAMIENTO Código Equipo: 1003622 CIP FILTRACIÓN Código Conjunto PM:									
Fecha propuesta inicio trabajos: 2005.09.15 Fecha real inicio de trabajos: Fecha propuesta fin trabajos: 2005.09.15 Fecha real fin de trabajos:									
<b>OPERACIONES DE MANTENIMIENTO</b>									
No Op	Clave Ctrol	No SubOp	Descripción	Puesto de Trabajo	Duracion	Unidad	No Pers	No Notif	No Solp
0010	PM01		EFECTUAR REVISIÓN A CONEXIÓN TUBERÍA OT GENERADA SEGÚN RUTA DE E.Z. MAYO	MECAMANT	1.0	H	1	000203392	
<b>COMENTARIOS</b>									

Fuente: Sección de Mantenimiento de Bavaria Cervecería de Boyacá

**Figura 3.** Orden de Trabajo Expedida en SAP-PM

a. Primera Parte



b. Continuación

The screenshot displays the SAP PM 'Visualizar aviso-MT: Aviso de avería GEB' interface. The top menu bar includes 'Aviso PM', 'Tratar', 'Pasara', 'Detalles', 'Entorno', 'Sistema', and 'Ayuda'. The SAP logo is visible in the top right corner. The main title is 'Visualizar aviso-MT: Aviso de avería GEB'. Below the title, there are several data fields: 'Aviso' (10012382), 'ZA CORREGIR FUGA POR EL SELLO BOMBA P 150', 'Status' (MECE ORAS), '0020', and 'Orden' (50024938). The interface is divided into tabs: 'Descripción', 'Datos de avería', and 'Fallas, Causas y Tareas'. The 'Datos de avería' tab is active, showing a 'Posición' section with fields for 'Parte objeto' (PM135001, 1003, MOTOR), 'Sínt. avería' (PM-SHUMA, 1010, ESCAPE DE PRODUCTO), 'Texto' (FUGA POR EL SELLO BOMBA P 150), 'Causas avería' (PM-CMECA, 1004, DEFORMACION), and 'Texto causa'. Below this is the 'Datos avería' section with 'Inicio avería' (2005.07.25 08:00), 'Fin de avería' (2005.07.25 12:00), 'Duración parada' (4.00 H), and a 'Parada' checkbox. The 'Responsabilidades' section shows 'Grupo planif.' (100 / BA04, Mto Planta), 'Pto.tbjo.resp.' (INGENTTO / BA04, INGENIERO MANTENIMIENTO), 'Usuario respons.' (FBECERRT, FABIO ALCIDES BECERRA TORRRRES), and 'Autor del aviso' (H. SOLORZANO, Fecha de aviso: 2005.07.25 08:00:00). The 'Fechas extremas' section shows 'Inicio deseado' (2005.08.12 11:51:13, Prioridad: Media) and 'Fin deseado' (2005.08.14 11:51:13, Parada checkbox). The bottom of the screen shows a navigation bar with left and right arrows.

Fuente: Sección de Mantenimiento Bavaria Cervecería de Boyacá.

El procedimiento explicado anteriormente sigue los estándares del Proceso de Información para Mantenimiento especificado en la norma ISO 14224, ya que al ocurrir el evento (daño en el equipo), se realiza la captura de la Información (reportar la avería y expedir la orden de trabajo) para con esto hacer el Procesamiento y Análisis de los datos el cual se expone en un informe de Mantenimiento Mensual (ver figura 4).

Actualmente se cuenta con que este sistema es poco seguro en cuanto a conservación de la Información ya que los reportes de las fallas presentadas en cada máquina se archivan sin ser ingresados al sistema lo cual puede causar pérdida de datos.

Se puede percibir que el procedimiento descrito anteriormente es demasiado rudimentario, poco versátil y no es muy seguro ya que no proporciona un verdadero apoyo al sistema de control de calidad; puesto que no cuenta con un registro sistematizado tanto de la cantidad de daños que presenta una máquina, como del modo y las causas por las cuales se originan dichos daños; el cual asegure la disponibilidad de la información necesaria para determinar los métodos de operación y seguimiento de los procesos con el fin de que la empresa labore óptimamente.

Por lo tanto la Sección Envase de Bavaria Cervecería de Boyacá ha decidido realizar un estudio con el propósito de cambiar el proceso para crear los Reportes de Fallos en las máquinas allí instaladas, por un sistema ágil, compacto, robusto y dinámico; que se encargue del reporte y registro, en el sistema, de las fallas presentadas.

**Figura 4. Plantilla Mensual**

	<p>BAVARIA S.A          CARVECERIA DE BOYACA          GERENCIA DE MANTENIMIENTO Y SERVICIOS  <b>CAUSAS DE FALLA EN EL TREN DE ENVASE N°2; MAYO/2005</b></p>
---	---

Denominación	Datos	DEFORMACIÓN	DESALINEACIÓ N	DESCONFIGUR ACIÓN	DESGASTE	FALLA DE CTROL	RUPTURA PREMATURA DE MATERIAL	INSTRUMENTO AVERIADO	SOBRECARGA	SOLTURA MECÁNICA	PERDIDA DEL PROGRAMA EN EL PLC	TOTAL GENERAL
DEPALETIZADORA	N° Fallas				1	1	1					3
	Tiempo (min)				56	20	10					86
DESEMPACADORA	N° Fallas					1		1		1		3
	Tiempo (min)					100		116		10		226
EMPACADORA	N° Fallas			1	3				3			7
	Tiempo (min)			8	24				22			54
ETIQUETADORA N°3	N° Fallas		2				1					3
	Tiempo (min)		23				55					78
ETIQUETADORA N°4	N° Fallas	11										11
	Tiempo (min)	33										33
PALETIZADORA	N° Fallas				3	3						6
	Tiempo (min)				10	10						20
PASTERIZADOTA	N° Fallas		5	1								6
	Tiempo (min)		5	10								15
OMNIVISION N°3	N° Fallas			1	1							2
	Tiempo (min)			20	50							70
OMNIVISION N°4	N° Fallas							1			1	2
	Tiempo (min)							30			10	40
LAVA DORA DE CAJAS	N° Fallas					1				2		3
	Tiempo (min)					50				20		70
LAVADORA DE BOTELLAS	N° Fallas			1		1						2

	Tiempo (min)			60		20						80
ENVASADORA N°3	N° Fallas	2			1							3
	Tiempo (min)	30			20							50
ENVASADORA N°4	N° Fallas				6			4				10
	Tiempo (min)				30			10				40
TRANSCANASTAS	N° Fallas					1						1
	Tiempo (min)					30						30
TRANSBOTELLAS	N° Fallas					5						5
	Tiempo (min)					20						20
Total n° fallas		<b>13</b>	<b>7</b>	<b>4</b>	<b>15</b>	<b>15</b>	<b>2</b>	<b>6</b>	<b>3</b>	<b>3</b>	<b>1</b>	<b>71</b>
Total tiempo (min)		<b>63</b>	<b>28</b>	<b>98</b>	<b>190</b>	<b>250</b>	<b>65</b>	<b>156</b>	<b>22</b>	<b>30</b>	<b>10</b>	<b>912</b>

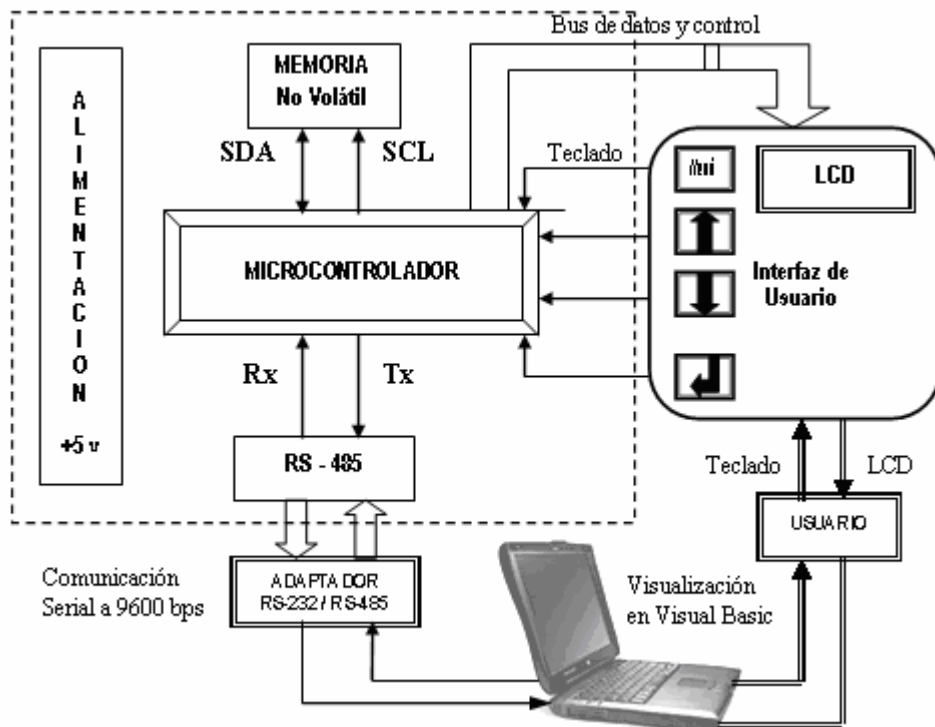
Fuente: Sección de Mantenimiento de Bavaria Cervecería de Boyacá.

De esta forma se podrán solucionar ágilmente los problemas encontrados en los equipos o en sus componentes, acortando el tiempo de parada, evitando tanto la disminución de la producción como la pérdida de información gracias a la creación de una base de datos, en la cual se actualizarán y almacenarán todos los reportes de las fallas y servirá como sustento en los procesos y estrategias de mantenimiento preventivo y programado. En esta los técnicos e ingenieros también pueden observar y estudiar el funcionamiento y desempeño de las máquinas para clasificar las fallas como esporádicas o crónicas, tratando de evitar que la producción se pare como consecuencia de la reincidencia de los daños.

## 2. DISEÑO E IMPLEMENTACIÓN DEL SISTEMA

En el presente capítulo se abordan los temas concernientes al diseño y construcción de la tarjeta electrónica del prototipo en el cual se elaboran los reportes de fallo, junto con la implementación de la red de comunicación y la aplicación de la Base de Datos creada en *MICROSOFT ACCESS 2003* con el fin de almacenar y administrar los reportes generados.

**Figura 5.** Esquema General del Sistema para Reporte de Fallas.



## 2.1. PROTOTIPO

La idea de desarrollar este prototipo, es ofrecer al usuario la posibilidad de utilizar un dispositivo que brinde confort, seguridad y confiabilidad tanto en el momento de elaborar los reportes de fallo de las máquinas, como en el momento de guardarlos en memoria y tenerlos almacenados allí, hasta que estos sean descargados e ingresados al PC.

**2.1.1. Operación.** El diseño de este prototipo permite al usuario realizar, editar y almacenar reportes de fallo junto con la fecha y hora en que fueron realizados, hasta el momento en que desde el PC se genere la orden para iniciar la descarga de datos, a través de una red estándar RS-485 diseñada para interconectar los prototipos con el PC central. Luego los datos son procesados en la Base de Datos especialmente diseñada para Bavaria Cervecería de Boyacá donde el usuario podrá disponer de ellos de acuerdo a sus necesidades.

El principio de funcionamiento del prototipo consiste en almacenar los reportes de fallo que el usuario uno a uno a creado o editado desde la interfaz de usuario (teclado y LCD) por medio de la navegación por menús diseñados para tal fin.

La ventaja principal del prototipo es su capacidad de almacenamiento puesto que permite guardar hasta 999 reportes de fallo durante el tiempo que se desee, además cuenta con la implementación del estándar RS-485 como salida de datos, por lo que se puede crear una red industrial multipunto de prototipos para realizar la descarga de los reportes desde cualquier parte de la planta. Otra ventaja que presenta este equipo radica en que los datos almacenados no se pierden a pesar de la ausencia de alimentación, ya que la memoria del sistema es no volátil, con capacidad de almacenamiento de 32 KBytes (256 Kbit). El usuario puede configurar el reloj del equipo, para garantizar la fecha y la hora exacta en que se realiza la prueba a través del menú.

Para la transmisión de la información se dispondrá de una red industrial administrada por un PC en el cual se hará la actualización y visualización de la Base de Datos, a través de la interfaz gráfica creada en Visual Basic 6.0® (ver figura 6), con el fin de obtener un registro de los reportes de fallo presentados en las máquinas. El PC actuará como maestro en el protocolo de comunicación, mientras que los prototipos operarán como esclavos y descargarán al computador los reportes de fallo previamente almacenados

**Figura 6.** Interfaz gráfica en Visual Basic



**2.1.2. Diseño de la Tarjeta.** La tarjeta del electrónica que constituye el prototipo tiene las siguientes tareas básicas: Proporcionar la alimentación necesaria para los Circuitos Integrados y demás elementos que hacen parte de ella, permitir interacción al usuario por medio del LCD y del teclado, almacenar los reportes de fallo y transmitirlos por el puerto serie a un PC

Para el diseño general de la tarjeta electrónica del prototipo, es necesario tener en cuenta las siguientes consideraciones que influyen en el resultado final.

- ❖ *Uso de una memoria EEPROM serial I2C:* Debido a la necesidad de mantener almacenados los reportes de fallo en el prototipo, aún en ausencia de alimentación y donde la velocidad de transferencia no es un punto crítico; se escogió una



**2.1.3. Componentes de la Tarjeta.** Uno de los procesos críticos al momento de realizar el diseño de cualquier tarjeta es la elección de los componentes de la misma, por lo tanto a continuación se realiza una explicación de los criterios en los cuales se basó la selección de los circuitos integrados con que se trabajaron finalmente.

- **Microcontrolador.** El microcontrolador además de ser el cerebro de la tarjeta, es el encargado del buen funcionamiento de los dispositivos externos a él como lo son: Memoria, *Transceiver*, Teclado y LCD; los cuales requieren de señales sincronizadas de control, para garantizar un óptimo desempeño.

La función primordial del microcontrolador es asegurar que la edición, almacenamiento y transmisión de los reportes de fallo de las máquinas sea correcta.

Para la elección del microcontrolador se revisaron las características técnicas y de programación que ofrecían algunos fabricantes enfatizando básicamente en el entorno de desarrollo, teniendo en cuenta que existe la necesidad de interfaces con periféricos externos, establecer comunicación serial con el PC, utilizar temporizadores y un conjunto de instrucciones fáciles de manejar.

Teniendo en cuenta los parámetros anteriores, se planteó como solución la elección de uno de los siguientes microcontroladores: de *Microchip*: *PIC16F877*, y *PIC16F874*; de *Motorola*: *MC68HC908GP32*. .Optándose finalmente por el ***MC68HC908GP32***<sup>5</sup> (ver figura 8), debido a que se adapta mejor a las necesidades del diseño; posee facilidades de trabajo como la programación, pues se dispone de un completo entorno de desarrollo para la familia de microcontroladores 68HC908G, el cual se puede descargar sin costo por Internet; y hay gran disponibilidad en el mercado a bajo costo tanto de chips, como de su tarjeta programadora.

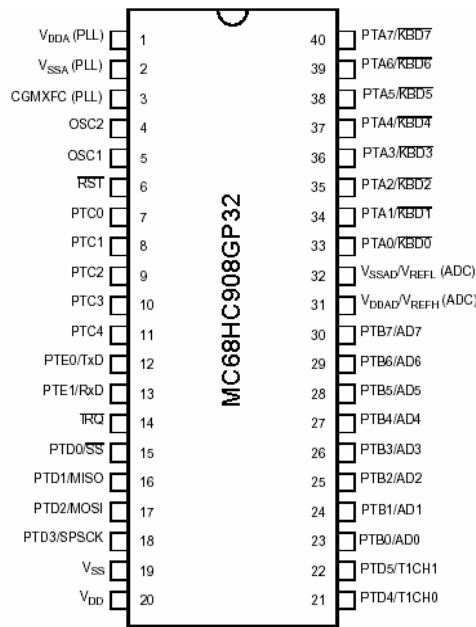
Algunas de las características de este microcontrolador son:

---

<sup>5</sup> Ver Anexo A, Hojas de datos

- ❖ El CPU 08 pertenece a la arquitectura *Von Neuman* clásica. En éste tipo de arquitectura existe un solo bus de datos, tanto para memoria de programas como para memoria de datos, lo que da origen a un mapa “lineal” de acceso a memoria y por consiguiente no existen instrucciones especiales para trabajar con datos o con código de programa; de ésta forma todas las instrucciones son aplicables a cualquier parte del mapa de memoria sin importar si se trabaja con datos o con programa.

**Figura 8.** Asignación de pines del microcontrolador *MC68HC908GP32*, y su uso en la tarjeta.



Fuente: Hoja de datos del fabricante.

- ❖ El módulo del CPU se vincula con el resto de los módulos del Microcontrolador por medio de un BUS de DATOS interno de 8 bits y un BUS de DIRECCIONES de 16 bits, que le permite direccionar código de hasta 64 KBytes. La frecuencia máxima del BUS interno es de 8 MHz reales a 5 Volts de alimentación y de 4MHz a 3 Volts.

- ❖ El mapa de memoria del **MC68HC908GP32** es de acceso continuo y sin saltos de página, en él se observan los registros de los puertos I/O y otros registros de uso general dentro de los primeros 256 KBytes y en la zona final de éste, como es costumbre en Motorola, se encuentran todos los vectores de interrupciones incluyendo el “Vector de *Reset*” y además registros de distintos periféricos (*SCI /SPI /TIMER*)
- ❖ El *CPU* está formado por dos grandes bloques, el *EXECUTION UNIT* y el “*CONTROL UNIT*”. El *Execution Unit* contiene la Unidad Lógica y Aritmética (encargada de todas las operaciones lógicas binarias y aritméticas), Registros internos del CPU (Acumulador, Puntero de Pila, Contador de Programa, Registro Índice, Registro de Código de Condiciones) y la interface con el Bus. El *control unit* contiene una máquina de estados finitos, unidades de control y tiempo para “manejar” la *execution unit*.
- ❖ EL *clock* del *CPU 08* necesita de un reloj de 4 fases para ejecutar un ciclo de máquina, por ello se necesita un cristal (oscilador externo) 4 veces superior a la frecuencia de BUS deseada.
- ❖ Gracias a una estructura interna de tipo *pipeline* en las distintas instrucciones del CPU se renuevan tantos ciclos de máquina como sean posibles, ya que mediante un mecanismo de “pre-búsqueda” se minimizan los tiempos muertos típicos de las arquitecturas *Von Neuman*.
- ❖ Tiene una Interface de Comunicación Serial (*SCI*), que permite comunicaciones asincrónicas de alta velocidad con dispositivos periféricos u otros microcontroladores; y un módulo de comunicación serial sincrónico, *SPI*, compatible con el bus *I2C (Inter-Integrate Circuit)*.
- ❖ Posee hasta 33 pines bidireccionales de Entrada/Salida en 5 diferentes puertos: A, B, D, C, E. Los puertos están mapeados como espacios de memoria, por lo general siempre en las primeras posiciones del mapa, y con dichos espacios pueden utilizarse todas las operaciones lógicas disponibles como si se tratara de

espacios de memoria estándar. Por medio del registro DDR (*Data Direction Register*) para cada uno de los puertos, se pueden establecer direcciones (*input/output*) de cada uno de los pines de los mismos.

- ❖ Tiene el módulo *MODULE BREAK*, el cual es el encargado de generar una interrupción que detenga el flujo normal del programa en una “dirección definida” para entrar en el programa *background*<sup>6</sup>, esta función es utilizada para generar rutinas de *Debbuging* (depuración) internas que permitan un rápido desarrollo o depuración ante determinadas circunstancias. Esta característica exclusiva de la familia HC908, permite la implementación de numerosos métodos de depuración de programas y herramientas de emulación poderosas sin dejar de ser por ello, económicas.
- ❖ Posee el módulo *MONITOR ROM*, el cual es exclusivo de la familia HC908 *Flash* de Motorola, que permite forzar al microcontrolador a un estado especial de “comunicación” con el mundo exterior. Este modo especial es muy útil cuando se quiere saber que está sucediendo en el interior de un microcontrolador al ejecutarse un programa dado. Si se lo utiliza correctamente, permite la grabación “En-Circuito” de la memoria *FLASH* del microcontrolador, borrado de la misma, actualización de programas, etc. Con éste módulo se tiene a disposición un mundo de herramientas interesantes de desarrollo en TIEMPO REAL con precios económicos y con prestaciones comparables a productos de mayor valor en otras marcas de microcontroladores.
- ❖ Tiene una capacidad de memoria *FLASH* para programa de 32K *bytes*, 512 *bytes* de memoria *RAM* y 307 *bytes* de *Monitor ROM*.
- ❖ Posee un *TIM* (*Timer Interfaz Module*) con dos canales (*TIM1* y *TIM2*) de 16 *bits*, que proporciona una referencia de tiempo con *IC* (*Input Capture*), *OC* (*Output Capture*) y *PWM* (*Pulse-Width-Modulation*). El componente central del *TIM* es un

---

<sup>6</sup> El programa *background* se debe entender como una rutina o programa completo que normalmente NO SE EJECUTA, salvo que se cumplan las condiciones para generar un “break-piont” (punto de quiebre en el programa del usuario durante la emulación del mismo) que permita saltar a esa rutina.

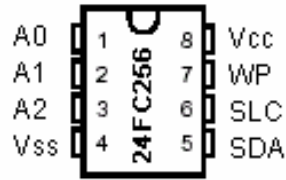
contador de 16-bit que puede operar como un contador *free-running* o como un módulo *up-counter*.

- ❖ Los pines I/O del puerto B se comparten con el conversor A/D de 8 bits de resolución. La conversión A/D se realiza por medio del Método de Aproximaciones Sucesivas, ésta puede ser simple o continua, también se cuenta con un *prescaler* con el cual se puede seleccionar el reloj del ADC.
  
- **Memoria Externa.** Como la cantidad de datos que se van a almacenar supera la capacidad que tiene el Microcontrolador (512 Bytes de RAM), fue necesario adicionar una memoria externa. Se escogió la Memoria serial *EEPROM* (*Electrically Erasable Programmable Read-Only Memory*) no volátil **24FC256** (ver figura 9) de la Empresa Microchip. Este elemento es un circuito integrado fabricado con tecnología CMOS, lo que le da alta velocidad de acceso; con capacidad de almacenar 32 KByte (256 Kbit).

Las características principales de esta memoria son las siguientes:

- ❖ Tecnología CMOS de baja potencia.
- ❖ Compatible con el Bus serial *I2C*, puesto que posee un bus para interfaz serial de 2 cables.
- ❖ Dispone de un modo Escribir-Página de 64 byte por página.
- ❖ Las líneas de direccionamiento funcional ( $A_0$ ,  $A_1$  y  $A_2$ ), correspondientes a los pines 1,2 y 3; permiten conectar en cascada hasta 8 dispositivos en el mismo bus, con lo que se obtiene un espacio de direccionamiento de hasta 2 Mbit.
- ❖ Este dispositivo cuenta con tres tipos básicos de operaciones para lectura de datos: Lectura Corriente de Direcciones, Aleatoria y Secuencial.
- ❖ Entradas *Schmitt trigger* para supresión de ruido.
- ❖ 1,000,000.0 de ciclos de borrado/escritura.
- ❖ Protección electrostática mayor de 4000 V.
- ❖ Detención de Datos, hasta 200 años.

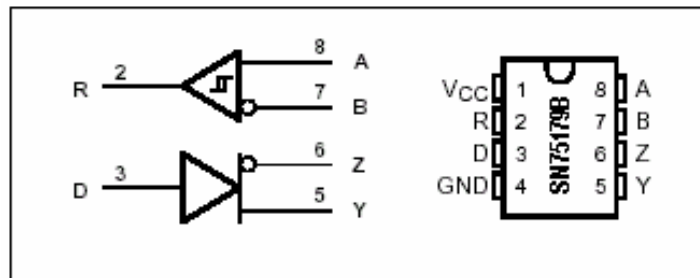
**Figura 9.** Distribución de pines de la memoria *EEPROM* 24FC256.



Fuente: Hoja de datos del fabricante.

- **Transceiver.** Para realizar la conexión de cada prototipo al Bus de la red se utilizó el integrado **SN75179b** de la empresa *Texas Instruments* (ver figura 10). Siendo éste circuito integrado económico, de fácil manejo y obtención en el mercado; además posee las mejores prestaciones que se ajustan perfectamente a los estándares internacionales.

**Figura 10.** Diagrama Lógico (lógica positiva) y distribución de los Pines del Circuito Integrado SN75179b



Fuente: Hoja de Datos del Fabricante.

Este integrado es un par diferencial *transmitter-receiver* diseñado para aplicaciones de transmisión de línea balanceada y para mejorar el desempeño de

las comunicaciones *full-duplex* de datos sobre largas líneas de Bus. El **SN75179b** posee las siguientes características:

- ❖ Compatible con los estándares TIA/EIA-422-B, TIA/EIA-485-A, y ITU Recomendación V.1
- ❖ Rangos de voltaje de Bus  $-7\text{ V}$  a  $12\text{ V}$
- ❖ Limitadores de corriente positiva y negativa
- ❖ Capacidad de manejo de corriente de salida de  $60\text{ mA}$  Máx.
- ❖ Protección térmica
- ❖ Impedancia de entrada del *Receiver* de  $12\text{ k}\Omega$  Min
- ❖ Sensibilidad en la entrada del *Receiver*:  $\pm 200\text{ mV}$
- ❖ *Histéresis* en la entrada del *Receiver*:  $50\text{ mV}$  (Típica)
- ❖ Opera desde una única fuente de alimentación de  $5\text{ V}$

**Tabla 1.** Niveles lógicos y de tensión para el Circuito integrado SN75179b

DRIVER		
INPUT D	OUTPUTS	
	Y	Z
H	H	L
L	L	H

RECEIVER	
DIFFERENTIAL INPUTS A - B	OUTPUT R
$V_{ID} \geq 0.2\text{ V}$	H
$-0.2\text{ V} < V_{ID} < 0.2\text{ V}$	?
$V_{ID} \leq -0.2\text{ V}$	L
Open	?

H = high level, L = low level, ? = Indeterminate

$V_{ID}$  = Voltaje Diferencial de Entrada

Fuente: Hoja de Datos del Fabricante.

- **Interfaz de usuario.** Con el fin de facilitar la interacción con el usuario, de tal forma que se puedan realizar los procedimientos: crear y editar los reportes de fallo, ver el estado de la memoria externa y configurar el terminal (actualizar fecha y hora, especificar el nombre del terminal y modificar la clave de acceso); fue necesario implementar una interfaz compuesta por una *LCD (Liquid Cristal Display)* y cuatro teclas individuales que conforman el teclado del prototipo (ver figura 11).

**Figura 11.** Foto del Prototipo



La *LCD* es una pantalla de cristal liquido con 16 columnas por 2 filas de caracteres alfanuméricos, en esta pantalla se visualizarán los menús junto con mensajes que guiarán al usuario en el momento de realizar la edición correspondiente al reporte de fallo o a la configuración del terminal.

Los reportes de fallos están formados de caracteres alfanuméricos, por tal motivo, se optó por programar en el microcontrolador MC68HC908GP32 las rutinas respectivas para que por medio de las teclas: **Menú** (Escapar<sup>7</sup>), ← (Aceptar), ↑ (Aumentar) y ↓ (Disminuir); el usuario pueda navegar carácter a carácter y a través

---

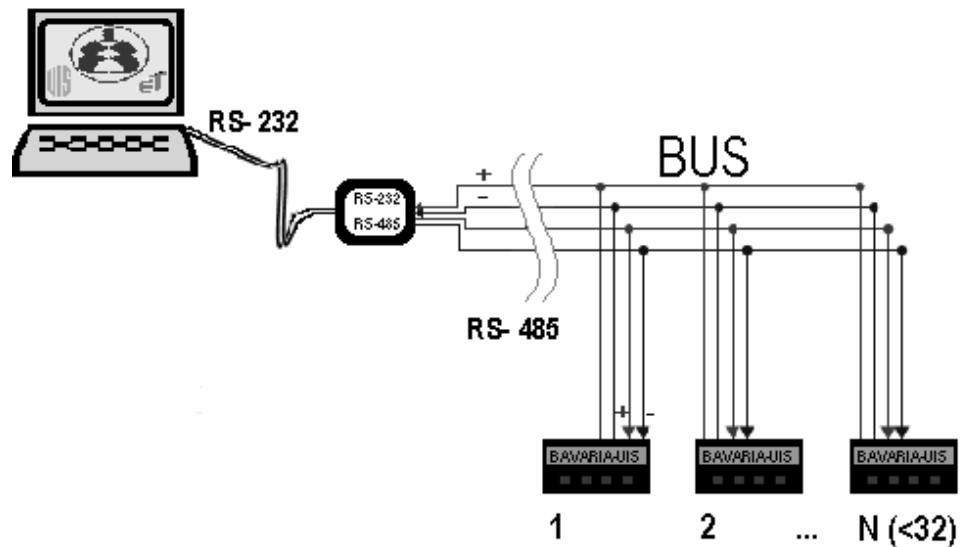
<sup>7</sup> Con esta tecla inicialmente se ingresa al menú principal, pero también sirve como elemento de salida de cualquiera de los menús.

de cada uno, durante la edición del reporte, hasta encontrar el correspondiente a los códigos requeridos para construir el reporte de fallo. Para tal fin se utilizó el puerto A del microcontrolador por ser éste el único que tiene capacidad para interrupciones de telado.

## 2.2. COMUNICACIÓN SERIAL VÍA RS-485

El propósito de éste proyecto de grado es permitir la supervisión de dos máquinas del Tren de Envase de la Cervecería desde un computador personal tipo PC, por tal razón se vió la necesidad de crear una red (ver figura 12), para realizar la comunicación entre los prototipos ubicados en cada máquina y el computador central.

**Figura 12.** Diagrama de la red para conectar los Prototipos y el PC.

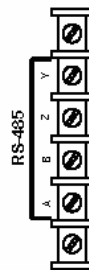


Se entiende por transmisión de datos al movimiento de los mismos desde un punto a otro, para lo cual es preciso establecer diversos parámetros tales como la distancia<sup>8</sup> y la velocidad<sup>9</sup>, con el fin de definir las condiciones de transferencia de la información. Hay que tener en cuenta que cuando la longitud del cable incrementa, la velocidad con la cual la información es transmitida debe ser reducida a fin de guardar una baja tasa de error de bits. Por lo tanto es muy importante escoger los estándares correctos que cubran la distancia de comunicación requerida y la tasa de datos necesaria.

Desde este punto de vista se analizó la conveniencia de utilizar las diferentes topologías de redes disponibles en la industria como: redes estándares TCP-IP, RS-485, Profibus, entre otras. Siendo la red estándar RS-485 una red económica que brinda soporte y seguridad para la integridad de los datos, contando con que es la más simple y fácil de utilizar debido a que no necesita hardware complejo en su implementación y no tiene protocolo asociado para su ejecución, por esto es posible desarrollarlo directamente desde el microcontrolador.

**2.2.1. Forma de Conexión.** La salida RS485 del prototipo entrega 4 líneas llamadas **A**, **B**, **Z** y **Y** (ver figura 11), en donde las líneas **A** y **B** representan la entrada del *receiver* y las líneas **Z** y **Y** la salida en el *driver* del *Transceiver* SN75179b (ver figura 13). En el protocolo RS485, estos pares de líneas funcionan como par diferencial (sin tierra absoluta como el RS232).

**Figura 13.** Dibujo de la bornera de salida en el prototipo



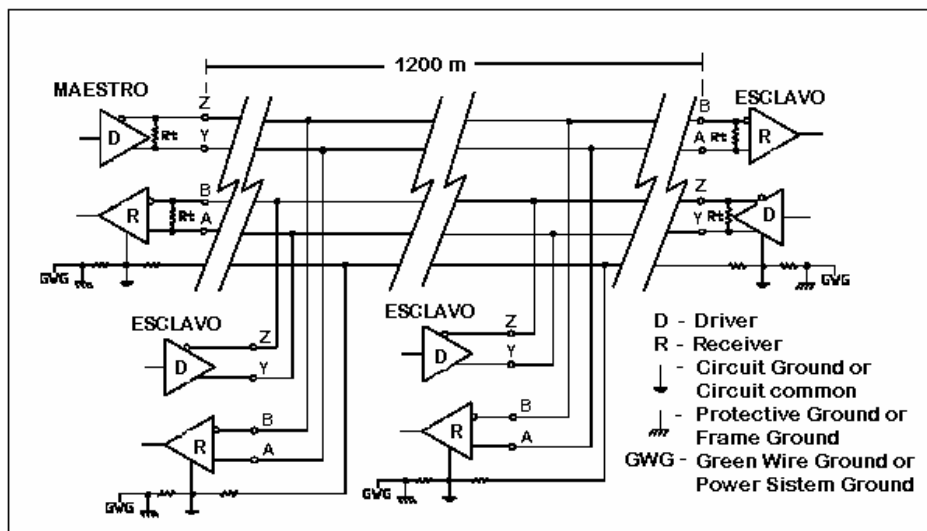
<sup>8</sup> Espacio entre los sistemas de envío y recepción.

<sup>9</sup> Rata con la cual los datos han sido enviados al dispositivo receptor.

El 1 lógico se representa por la condición en que el voltaje en la línea A, es mayor que el de la línea B, y al revés para el 0 lógico. Típicamente  $A - B = 5$  Volts para 1 lógico y  $A - B = -5$  Volts para 0 lógico. Los niveles lógicos en la salida del Receiver (líneas Z y Y) se pueden ver en la tabla1 (Niveles lógicos y de tensión para el Circuito integrado SN75179b, ver página 28).

La comunicación RS485 se realiza con la topología de red a cuatro hilos, en modo Maestro-Esclavo y para la transmisión se utiliza el modo *half-duplex* (ver figura 14).

**Figura 14.** Configuración típica Multipunto de 4 Hilos para RS-485



Fuente: B&B Electronics Mfg. Co. Inc. Nota de aplicación: RS-422 y RS-485. (Modificada por los Autores).

En el modo de transmisión *half-duplex* la comunicación se lleva a cabo en ambos sentidos, pero no simultáneamente. Esto es, se trata de una comunicación bidireccional, donde no hay cese de información en la línea; la información circula en un sentido o en otro pero no en ambos. Igualmente se implementó el modo Maestro-Esclavo, esta forma de operación permite tener varios prototipos conectados a un PC; en donde el PC es quien administra la red y de esta manera da la orden para que se establezca la comunicación y

transferencia de datos con cualquiera de los esclavos.

Los prototipos se conectan a las líneas como aparece en la figura 14, pasando los cables desde cada prototipo al adaptador. Es conveniente colocar una resistencia (1/4 watt) en cada uno de los extremos de la línea de transmisión para acoplar las impedancias y evitar rebotes de la señal. Estas resistencias deben ser de la misma cantidad de ohms de la impedancia característica del cable y la cual es específica por su fabricante.

El cable del bus debe ser trenzado y preferiblemente blindado, adecuado para transmisión digital de poca velocidad (menos de 10 Mbit/seg.), de los que hay muchos en el mercado.

**2.2.2. Protocolo.** Este protocolo aportó a la comunicación utilizada la capacidad de detectar errores de la transmisión y el hecho de poder ser utilizada para transferir los reportes de fallo. De este protocolo se pueden destacar las siguientes características:

- ❖ El computador o PC realiza la labor de maestro (coordina la comunicación) en tanto que los prototipos son esclavos y sólo responden a las preguntas ó comandos del maestro.
- ❖ Es un protocolo diseñado para microcontroladores, y por lo tanto utiliza caracteres de 8 bits para empaquetar sus tramas.
- ❖ Envía el nombre del prototipo, los datos del reporte de fallo, la fecha y hora de creación del mismo.
- ❖ Está diseñado para ser utilizado a velocidad de transmisión de 9600 baudios.
- ❖ Su modo de operación es *half-duplex* de parada y espera, utilizando una red de cuatro hilos y un estándar balanceado (RS-485)

➤ **Formato de las tramas**

- ❖ **Trama Principal.** Los reportes de fallos transmitidos con éste protocolo constan de tramas de 32 Bytes (ver figura 15). Se optó por esa longitud de trama debido a

que de ésta forma se facilita el almacenamiento en la memoria externa del prototipo; ya que la memoria, de 32 KBytes, está conformada por páginas de 64 Bytes, luego es más sencillo direccionar y almacenar dos reportes de fallo por página en la memoria externa.

**Figura 15.** Trama de Reporte de Fallo

<b>a</b>	<b>@</b>	<b>0</b>	<b>0</b>	<b>5</b>	<b>0</b>	<b>1</b>	<b>5</b>	<b>0</b>	<b>A</b>	<b>7</b>	<b>5</b>	<b>B</b>	<b>0</b>	<b>5</b>	<b>J</b>
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

<b>R</b>	<b>P</b>	<b>%</b>	<b>0</b>	<b>5</b>	<b>0</b>	<b>5</b>	<b>2</b>	<b>6</b>	<b>1</b>	<b>5</b>	<b>2</b>	<b>5</b>	<b>1</b>	<b>3</b>	<b>&amp;</b>
16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31

Cada trama esta constituida por: código del Terminal (1 byte), carácter de inicio del reporte de fallo (1 byte), caracteres de reporte de fallo (16 bytes), carácter de mitad de trama (1 byte), caracteres de fecha y hora (12 bytes), carácter de fin de trama (1 byte). Ver tabla 2.

**Tabla 2.** Formato de trama del Reporte de Fallo.

# de Bytes	Ejemplo	Contenido
1	a	Código de terminal.
1	@,	Carácter de INICIO. Fue especificado para indicar que aquí comienza el reporte de fallo.
3	005	Código de Máquina
2	01	Código de Módulo
2	50	Código de Parte
3	A75	Código de Falla
3	B05	Código de Causa de Falla
3	JRP	Código de Operario
1	a	Código de terminal, se usa aquí para

		direccionar los datos
2	05	Año
2	05	Mes
2	26	Día
2	15	Hora
2	25	Minutos
2	13	Segundos
1	&	Carácter de FIN, especificado para indicar el final del reporte

- ❖ **Trama Fecha y Hora.** Se envía desde el PC al terminal, con el fin de que el prototipo actualice la fecha y hora, para que tanto el PC como el equipo queden sincronizados. Ver figura 16. Los Bytes se especifican en la tabla 3

**Figura 16.** Trama de Fecha y hora

a	0	5	/	0	5	/	2	6	_	1	5	:	2	5	:	1	3	&
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18

**Tabla 3.** Formato de fecha y hora.

# de Bytes	Ejemplo	Contenido
1	a	Carácter de comienzo de trama, se usa para direccionar los datos.
2	05	Año
1	/	Carácter de separación
2	05	Mes
1	/	Carácter de separación
2	26	Día
1	–	Espacio en blanco (separa la fecha de la hora)
2	15	Hora
1	:	Carácter de separación
2	25	Minutos
1	:	Carácter de separación

2	13	Segundos
1	&	Carácter de FIN, especificado para indicar el final del reporte

- ❖ **Trama Número de Registros.** Contiene la información concerniente a la cantidad de reportes almacenados en la memoria externa del prototipo, ó el número de registros descargados en cada transmisión en el PC (3 bytes) y un carácter de fin de la trama (1 byte). Ver figura 17.

**Figura 17.** Formato del Número de registros almacenados.

a	0	0	3	&
0	1	2	3	4

### ➤ **Funcionamiento de la comunicación**

- ❖ Edición del Reporte de Fallo. El repote de fallo se edita desde cada terminal teniendo en cuenta las siguientes consideraciones:
  - ☑ El prototipo debe estar encendido y el PC puede o no estar encendido. Cada vez que se edite un reporte de fallo, éste se almacena en la memoria externa, controlada por el microcontrolador MC68HC908GP32; de tal forma que cuando el PC presente un fallo, no se encuentre disponible o esté apagado los reportes no se perderán.
  - ☑ El reporte se edita desde cada prototipo siguiendo la secuencia de menús y comentarios visualizados en cada pantallazo y utilizando el teclado configurado para tal fin. Durante el ingreso de cada carácter, en cada código se va indicando al usuario que clase de carácter debe ingresar, es decir si éste se trata de un número

o por el contrario de una letra, puesto que hay códigos numéricos, alfanuméricos o alfabéticos.

- ☑ Al terminar la edición del reporte se pide al usuario almacenarlo en la memoria y allí es guardado, hasta que el PC le da la orden al prototipo para realizar la descarga de todos los datos.
  
- ❖ Comunicación entre el terminal y el PC. Una vez conectados los prototipos al Bus de transmisión el PC envía el carácter “a” (Nombre del primer prototipo); el prototipo que responde a éste nombre envía de vuelta al PC el mismo carácter (“a”) como confirmación. De esta forma se establece comunicación entre el prototipo “a” y el PC. El otro prototipo conectado al bus, que corresponde al nombre “b”, tan solo escucha, no responde nada y sigue escuchando el canal hasta que se le llame, es decir, cuando el PC envíe el carácter “b”.

Cuando el PC recibe el carácter que indica el nombre del prototipo con el que ha establecido comunicación, envía inmediatamente la fecha y la hora actual para que el prototipo actualice la hora de su reloj y se sincronice con el PC. Como confirmación de haber recibido la hora el terminal le envía al PC el número de reportes almacenados en la memoria externa hasta ese instante. Este número de reportes es almacenado en el PC. De esta forma tanto el prototipo como el PC están listos para iniciar la transferencia de datos.

Si el número correspondiente a la cantidad de reportes de fallo es diferente de cero, el prototipo inicia la descarga de todos los reportes almacenados en la memoria externa. El PC recibe los datos y éstos se almacenan ahora en la Base de Datos diseñada para administrar la información recolectada.

Al finalizar la transferencia de datos, el PC compara el número de reportes recibido con el número de reportes enviado por el terminal antes de iniciarse la descarga. Al realizar la comparación se pueden presentar dos casos:

- ☑ Los números son iguales, entonces el PC le envía al terminal dicho número (es decir el número de reportes recibidos que es igual al número de reportes almacenados en la memoria externa del prototipo.)

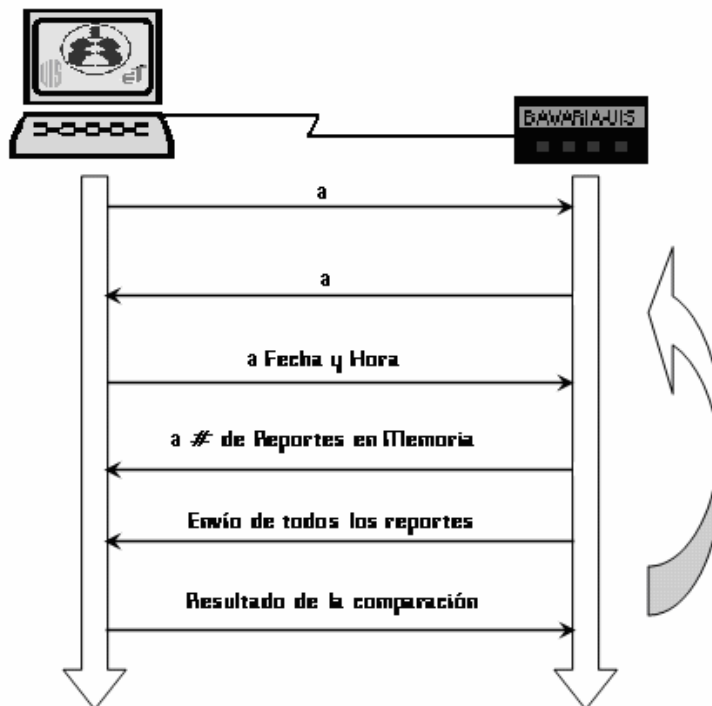
- ☑ Los número son diferentes, entonces el PC le envía al terminal el número: **0**.

Cuando el terminal recibe el número de reportes que llegaron al PC, compara esta cantidad con el número de registros que tiene almacenados en la memoria externa. Al realizar ésta comparación se pueden presentar dos casos:

- ☑ Los número son iguales, es decir el número de reportes de fallo que llegaron al PC es igual al número de reportes almacenados en la memoria; entonces la comunicación se realizó satisfactoriamente
- ☑ Los números son diferentes, entonces el Terminal continúa almacenando los datos y no dispone de éstas posiciones de la memoria para almacenar más reportes de fallo.

Para cualquiera de los dos casos el PC envía de nuevo un nombre de terminal, de forma tal, que aquel prototipo que responda a éste nombre, realice de nuevo el ciclo de comunicación.

**Figura 18.** Funcionamiento del Protocolo



## **2.3. Base de Datos**

Para implementar el sistema de reporte de fallas de las máquinas instaladas en el tren de envase de Bavaria Cervecería de Boyacá, se requiere almacenar la información concerniente a las máquinas, a sus componentes, a las fallas que pueden presentar dichas máquinas y a los reportes de las fallas que se produzcan para cada máquina. Por tal razón se construyó una base de datos en MICROSOFT OFFICE ACCESS 2003® con el fin de albergar la información mencionada anteriormente y disponer en cualquier momento de un historial del comportamiento de cada uno de los equipos instalados en el tren de envase, de esta forma se optimiza y facilita el control del Mantenimiento Preventivo de la Sección de Envase de la Empresa.

**2.3.1 Estructura de la Base De Datos.** La base de datos elaborada consta de 9 tablas en las cuales los datos se agrupan en campos (columnas) según el tipo de información y en registros (filas) o colección de campos, los cuales pueden ser ampliados en su totalidad. Toda la información contenida en las tablas es tipo texto, debido a que es más sencillo la manipulación de esta mediante la interfaz desarrollada en visual Basic 6.0®.

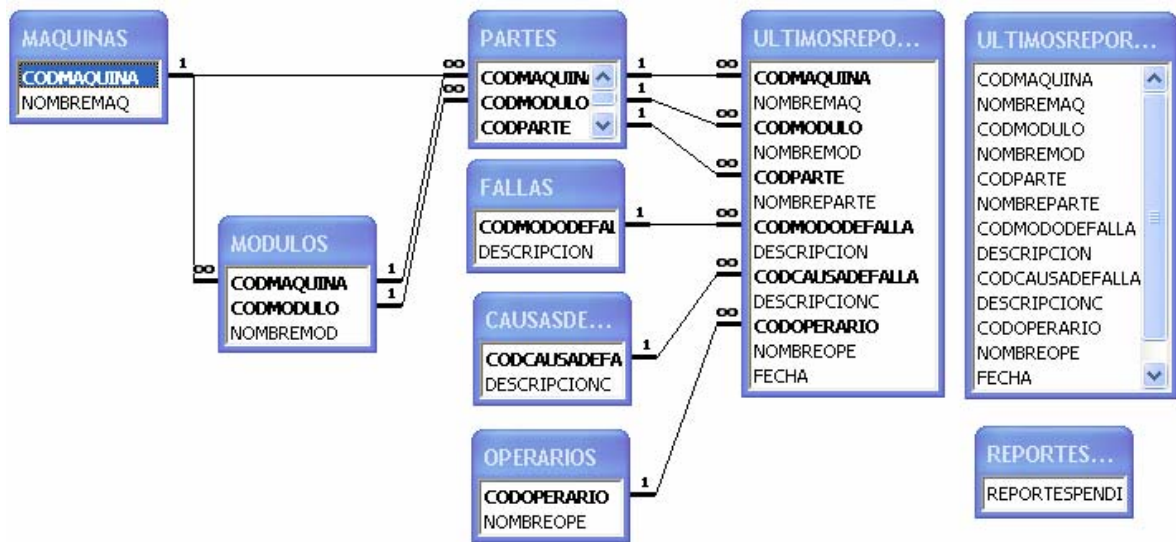
Para el diseño de la base de datos se optó por un modelo de tipo relacional, debido a que permite realizar cambios y ampliaciones en la base de datos sin que esto implique modificaciones, en su estructura o en sus programas de aplicación. Otra ventaja del modelo relacional, radica en que al enlazar varias tablas, se comparte información entre ellas, por lo tanto se evita la duplicidad de datos y se ocupa menos espacio en disco; además de lograr encontrar y reunir información almacenada en diferentes tablas, de forma fácil y rápida.

Las relaciones utilizadas para las siete tablas que constituyen el modelo relacional de esta base de datos son del tipo Uno a Muchos, en la que cada registro de una tabla principal o madre puede tener más de un registro enlazado a otras tablas secundarias o hijas; pero cada registro contenido en las tablas hijas solo puede estar enlazado con un registro en la

tabla madre. Dichas relaciones se producen mediante un campo con las mismas características que tienen en común las tablas implicadas y se denomina campo clave, como se ilustra en la figura 19. Una característica que vale destacar del modelo relacional reside en que permite desplegar y visualizar desde la tabla madre las tablas con las cuales esta relacionada como se muestra en la figura 20.

En la implementación de la base de datos, las relaciones comienzan a desencadenarse de la tabla Máquinas y terminan en la tabla Últimos Reportes debido a que esta necesita la información contenida en las otras 6 tablas relacionadas.

**Figura 19.** Modelo Relacional Diseñado para la Base de Datos Elaborada

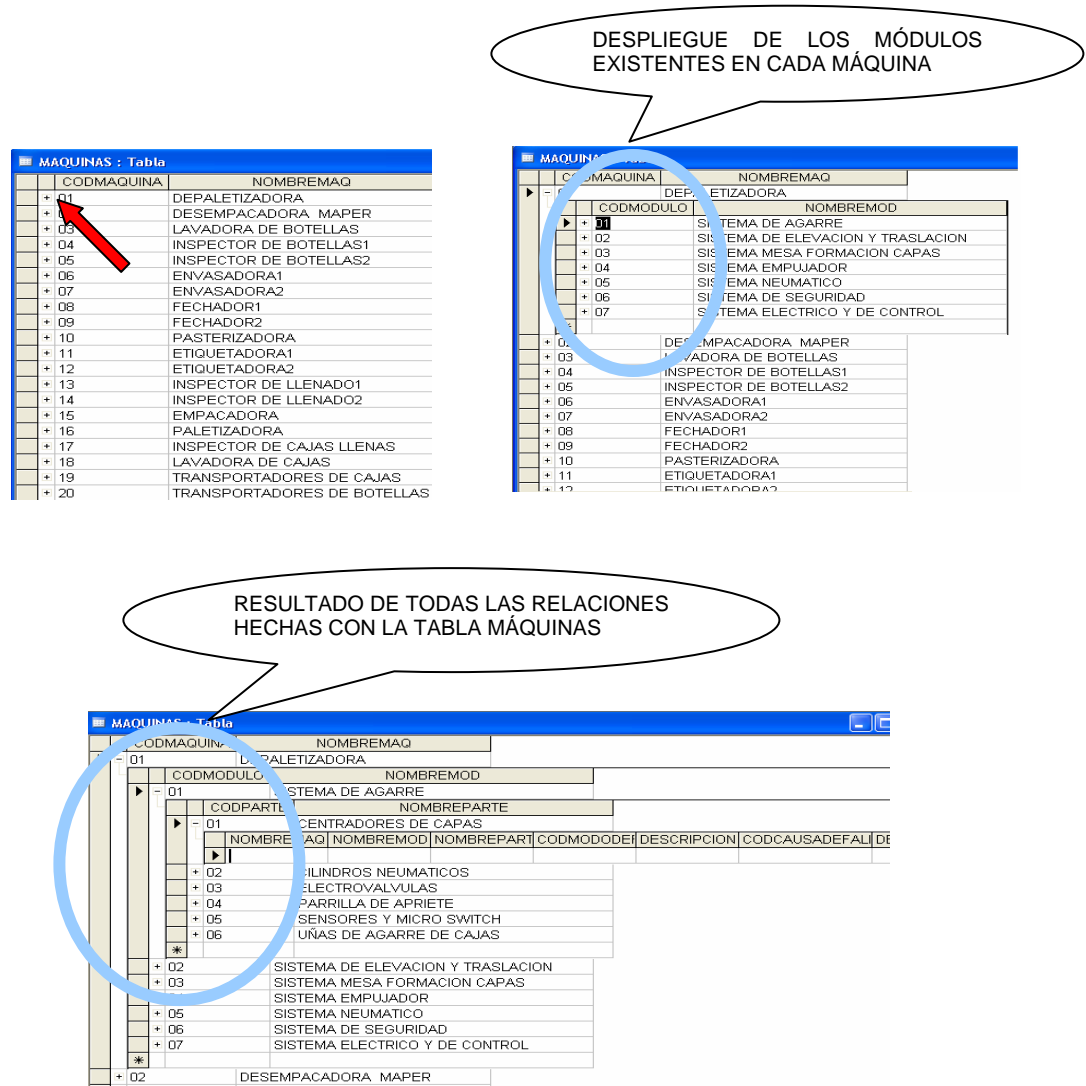


- **Tabla Máquinas.** Almacena el listado de las máquinas instaladas en el tren de envase y los códigos con los cuales se identifican.

Consta de dos campos (Código de Máquina y Nombre de Máquina) y 20 registros (filas).

En esta tabla el campo clave o llave principal es el Código de Máquina, ya que por medio de él se ligan las tablas Módulos, Partes y Últimos Reportes, ver figura 20.

**Figura 20. Estructura Tabla Máquinas**



- **Tabla Módulos.** Alberga el total de los módulos o agrupación de partes que en conjunto desempeñan una labor específica y conforman cada máquina instalada en el tren de envase.

La información contenida en esta tabla está organizada en tres campos (Código de Máquina, Código de Módulo y Nombre de Módulo) y cuenta con 152 registros.

Debido a que la tabla Módulos desempeña el papel de hija en la relación con la tabla Máquinas, hereda el campo clave Código de Máquina mediante el cual se relacionan entre si. Adicionalmente tiene un segundo campo clave (Código de Módulo) con el fin de establecer relación madre e hija con la tabla Partes, ver figura 21

**Figura 21.** Estructura Tabla Módulos

CODMAQUINA	CODMODULO	NOMBREMOD
+ 01	01	SISTEMA DE AGARRE
+ 01	02	SISTEMA DE ELEVACION Y TRASLACION
+ 01	03	SISTEMA MESA FORMACION CAPAS
+ 01	04	SISTEMA EMPUJADOR
+ 01	05	SISTEMA NEUMATICO
+ 01	06	SISTEMA DE SEGURIDAD
+ 01	07	SISTEMA ELECTRICO Y DE CONTROL
+ 02	01	SISTEMA CABEZAL DE AGARRE
+ 02	02	SISTEMA DE MANEJO CAJAS Y BOTELLAS
+ 02	03	SISTEMA ACCIONAMIENTO PRINCIPAL
+ 02	04	SISTEMA NEUMATICO
+ 02	05	SISTEMA DE LUBRICACION
+ 02	06	SISTEMA DE SEGURIDAD
+ 02	07	SISTEMA ELECTRICO Y DE CONTROL
+ 03	01	SISTEMA DE CARGE Y DESCARGE
+ 03	02	SISTEMA EXTRACCION DE ETIQUETAS
+ 03	03	SISTEMA MANEJO DE AGUA
+ 03	04	SISTEMA MANEJO DE SODA
+ 03	05	SISTEMA MANEJO DE VAPOR
+ 03	06	SISTEMA ACCIONAMIENTO PRINCIPAL
+ 03	07	SISTEMA NEUMATICO

CODMAQUINA	CODMODULO	NOMBREMOD
- 01	01	SISTEMA DE AGARRE
		NOMBREPARTE
+ 01	01	CONTRADORES DE CAPAS
+ 02	01	CILINDROS NEUMATICOS
+ 03	01	ELECTROVALVULAS
+ 04	01	ARRILLA DE APRIETE
+ 05	01	SENSORES Y MICRO SWITCH
+ 06	01	UNAS DE AGARRE DE CAJAS

➤ **Tabla Partes.** La tabla partes esta compuesta por la totalidad de las partes que en conjunto forman los módulos de las máquinas.

Esta constituida por cuatro campos (Código de Máquina, Código de Módulo, Código de Parte y Nombre de Parte) y de los cuales solo los dos últimos son propios, con la aclaración que Código de Parte es un campo clave y sirve como canal de conexión con la tabla Últimos Reportes. Mientras que los dos primeros campos son heredados de la tabla Módulos, y por medio de estos campos claves se logran instituir tanto las relaciones con las tablas Máquinas y Módulos como la relación con la tabla que contiene los Reportes recibidos del prototipo.

La estructura de esta tabla alberga 1117 registros o filas en las que se almacena y organiza la información de los cuatro campos de la misma, como se ilustra en la figura 22

**Figura 22.** Estructura Tabla Partes

VISUALIZACIÓN DE LOS FALLOS REPORTADOS PARA CADA PARTE

CODMAQUINA	CODMODULO	CODPARTE	NOMBREPARTE
01	01	01	CENTRADORES DE CAPAS
01	01	02	CILINDROS NEUMATICOS
01	01	03	ELECTROVALVULAS
01	01	04	PARRILLA DE APRIETE
01	01	05	SENSORES Y MICRO SWITCH
01	01	06	UNAS DE AGARRE DE CAJAS
01	02	01	ACOPLES
01	02	02	CONTRAPESAS
01	02	03	CORREAS/CADENAS
01	02	04	DESLUZAMIENTO TRASLACION
01	02	05	FRENOS MOTORES
01	02	06	MOTORES Y FRENOS
01	02	07	POLEAS DE RODADURA
01	02	08	REDUCTORES
01	02	09	SENSORES Y MICRO SWITCH
01	03	01	BLOQUEADORES
01	03	02	DESVIADOR DE CAJAS SALIDA
01	03	03	MESA
01	03	04	MOTORES
01	03	05	REDUCTORES
01	03	06	SENSORES Y MICRO SWITCH
01	03	07	TRANSPORTADOR

CODMAQUINA	CODMODULO	CODPARTE	NOMBREPARTE
01	01	01	CENTRADORES DE CAPAS
01	01	02	CILINDROS NEUMATICOS
01	01	03	ELECTROVALVULAS
01	01	04	PARRILLA DE APRIETE
01	01	05	SENSORES Y MICRO SWITCH
01	01	06	UNAS DE AGARRE DE CAJAS
01	02	01	ACOPLES
01	02	02	CONTRAPESAS
01	02	03	CORREAS/CADENAS
01	02	04	DESLUZAMIENTO TRASLACION
01	02	05	FRENOS MOTORES
01	02	06	MOTORES Y FRENOS
01	02	07	POLEAS DE RODADURA
01	02	08	REDUCTORES
01	02	09	SENSORES Y MICRO SWITCH
01	03	01	BLOQUEADORES
01	03	02	DESVIADOR DE CAJAS SALIDA
01	03	03	MESA
01	03	04	MOTORES

- **Tabla Fallas.** En esta tabla se guarda el listado de la forma como el operario percibe las averías que se pueden producir en todas las máquinas y que originan la pérdida de capacidad de las mismas para desempeñar la labor para la cual fueron diseñadas. Dichas fallas son clasificadas y codificadas según norma ISO 14224, por Bavaria en dos categorías: modos de fallas por detección humana (Códigos H) y modos de fallas por detección de instrumentos (Códigos I).

Los datos de esta tabla se organizan en dos campos (Código Modo de Falla y Descripción) y 33 registros como se puede apreciar en la figura 23.

El campo clave para esta tabla es Código Modo de Falla mediante el cual se relaciona con la tabla Últimos Reportes.

**Figura 23.** Estructura Tabla Fallas

FALLAS : Tabla		
	CODMODODEFALLA	DESCRIPCION
▶	H01	APARIENCIA INADECUADA DEL PRODUCTO
+	H02	ALTO REBOSE
+	H03	CAIDA DE BOTELLAS
+	H04	CONSUMO EXCESIVO DE INSOMOS
+	H05	CONSUMO EXCESIVO DE SERVICIOS
+	H06	DIFICULTAD EN SINCRONIZACION
+	H07	EMISION ANORMAL DE GASES
+	H08	ESCAPE DE FLUIDOS DE SERVICIOS
+	H09	ESCAPE DE INSUMOS
+	H10	ESCAPE DE PRODUCTO
+	H11	INESTABILIDAD ESTRUCTURAL
+	H12	NO FUNCIONA
+	H13	NO SE DETIENE
+	H14	OBSTRUCCION
+	H15	OLORES EXTRAÑOS
+	H16	PARADA INESPERADA
+	H17	PERDIDA DE CAPACIDAD
+	H18	PERDIDA DE CICLO
+	H19	RETARDO EN OPERACIÓN
+	H20	RUIDO ANORMAL
+	H21	VARIACION DE NIVEL NO CONTROLADO
+	I01	ACTIVACION DE ALARMAS
+	I02	ACTIVACION DE PROTECCIONES ELECTRICAS
+	I03	ACTIVACION DE PROTECCIONES MECANICAS
+	I04	BAJA LECTURA DE AISLAMIENTO
+	I05	COMUNICACIÓN DEFICIENTE
+	I06	CONCENTRACION ANORMAL

- **Tabla Causas de Falla.** La tabla Causas de Falla almacena el listado de las circunstancias por las cuales se puede presentar una falla en un componente de una determinada máquina. Bavaria Cervecería de Boyacá tiene clasificadas las causas de falla las de la siguiente forma: Causas Externas (Códigos E), Causas por Falla de Instrumentación (Códigos F), Causas Mecánicas (Códigos M), y Causas Operacionales (Códigos O).

Los datos que contiene esta tabla se organizan en dos campos (Código Causa de Falla y DescripciónC) y 43 registros ver figura 24.

La tabla Causas de Falla se relaciona con la tabla Últimos Reportes por medio del campo clave Código Causa de Falla.

**Figura 24. Estructura Tabla Causas de Fallas**

CAUSASDEFALLA : Tabla	
CODCAUSADEFALLA	DESCRIPCIONC
+ E01	CONTAMINACION
+ E02	DESCARGA ATMOSFERICA
+ E03	FALLA EN EL SUMINISTRO DE INSUMOS
+ E04	FALLA EN EL SUMINISTRO DE PRODUCTOS
+ E05	FALLA EN EL SUMINISTRO DE SERVICIOS
+ E06	HUMEDAD
+ E07	PLAGAS
+ E08	TAPONAMIENTO O BLOQUEO
+ E09	TEMPERATURA AMBIENTE
+ E10	VANDALISMO
+ F01	ARRANQUE FORZADO
+ F02	DAÑO DE TARJETA ELECTRONICA
+ F03	DESBALANCEO DE VOLTAJE
+ F04	DESCALIBRACION
+ F05	DESCONFIGURACION
+ F06	FALLA DE CONTROL
+ F07	FALLA EN EL SUMINISTRO DE ENERGIA
+ F08	FALLA EN LA TIERRA O EL AISLAMIENTO
+ F09	INSTRUMENTO AVERIADO
+ F10	PERDIDA DEL PROGRAMA EN EL PLC
+ F11	PICO DE VOLTAJE
+ F12	PROBLEMA DE REGULACION
+ F13	SOBRECARGA
+ M01	AGARROTAMIENTO
+ M02	CAVITACION
+ M03	CORROSION

- **Tabla Operarios.** Esta tabla alberga los nombres y códigos de los operarios que laboran en el tren de envase de Bavaria Cervecería de Boyacá, los cuales son los encargados de reportar las fallas producidas en las máquinas.

La información contenida en la tabla en mención esta organizada en dos campos (Código Operario y Nombre Operario) y contiene 54 registros ver figura 25.

La tabla operarios establece conexión con la tabla Últimos Reportes mediante el campo clave Código Operario.

**Figura 25. Estructura Tabla Operarios**

LISTADO DE LOS FALLOS REPORTADOS POR CADA OPERARIO

CODOPERARIO	NOMBREOPE
AIR	ALVARO OCTAVIO INFANTE RIAÑO
AJC	ALEJANDRO ORTEGON CORREDOR
AZR	ABRAHAM ZULLUAGA RENGIFO
CAP	CARLOS ALBERTO PATIÑO
CFG	CARLOS ARIOSTO FUENTES GIL
CGC	CARLOS JAVIER GARCIA CAMACHO
CMP	CARLOS ALBERTO MORA PINZON
CSP	CARLOS ALBERTO SIABATTO PALACIOS
DHV	DILVAR ALONSO HIGUERA VALDERRAMA
EPC	EDGAR FERNANDO PUERTO CORREDOR
EZR	ECCEHOMO ZAMBRANO RIVERA
FBT	FABIO ALCIDES BECERRA TORRES
FCD	FRANCISCO JAVIER CASTRO DAVILA
FHH	FERNANDO DE JESUS HERNANDEZ HIGUERA
FPN	FABIO ERNESTO PEREZ NIÑO
GBC	GERMAN BAEZ CABRA
GMB	GONZALO MORALES BARRERA
HNS	HERNAN ENRIQUE NAVARRETE SANCHEZ
HSG	HAROLD WILSON SOLORZANO GALVIS
IRT	IGNACIO RUEDA TORRES
JAM	JUAN BERNARDO ALZATE MEJIA
JCI	JOSE ALVARO CARMONA LOPEZ

CODMAQUINA	NOMBREMAQ	CODMODULO	NOMBREMOD	CODPARTE	NOMBREPART	CODMODO	CODCAUSA	DESCRIPCION	CODCAUSAC	CODOPERARIO	NOMBREOPERARIO	FECHA	HORA
AIR	ALVARO OCTAVIO INFANTE RIAÑO												
AJC	ALEJANDRO ORTEGON CORREDOR												
AZR	ABRAHAM ZULLUAGA RENGIFO												
CAP	CARLOS ALBERTO PATIÑO												
CFG	CARLOS ARIOSTO FUENTES GIL												
CGC	CARLOS JAVIER GARCIA CAMACHO												
CMP	CARLOS ALBERTO MORA PINZON												
CSP	CARLOS ALBERTO SIABATTO PALACIOS												
DHV	DILVAR ALONSO HIGUERA VALDERRAMA												
EPC	EDGAR FERNANDO PUERTO CORREDOR												
EZR	ECCEHOMO ZAMBRANO RIVERA												
FBT	FABIO ALCIDES BECERRA TORRES												
FCD	FRANCISCO JAVIER CASTRO DAVILA												
FHH	FERNANDO DE JESUS HERNANDEZ HIGUERA												
FPN	FABIO ERNESTO PEREZ NIÑO												
GBC	GERMAN BAEZ CABRA												
GMB	GONZALO MORALES BARRERA												
HNS	HERNAN ENRIQUE NAVARRETE SANCHEZ												
HSG	HAROLD WILSON SOLORZANO GALVIS												

- **Tablas Últimos Reportes y Últimos Reportes SR.** Éstas tablas se crearon con el fin de almacenar y ordenar los reportes, enviados por los prototipos, en 14 campos ver figura 26 (Código de Máquina, Nombre de Máquina, Código de Módulo, Nombre de Módulo, Código de Parte, Nombre de Parte, Código Modo de Falla, Descripción, Código Causa de Falla, DescripciónC, Código Operario, Nombre Operario, Fecha y Hora) donde sus datos son tipo texto y su número de registros es indeterminado, ya que este aumenta a medida que se reciben reportes.

En la tabla Últimos reportes existen seis campos clave que corresponden los códigos de Máquina, Módulo, Parte, Modo de Falla, Causa de Falla y operario; ya que mediante estos se enlazan con las 6 tablas anteriores, y los registros se van organizando de acuerdo a la clave principal o al Código De la Máquina en la cual ocurrió la falla. Mientras que en la tabla Últimos Reportes SR los registros se organizan y guardan según el orden de ocurrencia de la falla, esto se debe a que

esta tabla no esta relacionada con ninguna otra y por lo tanto en ella no hay campo clave.

**Figura 26.** Estructura Tablas Últimos Reportes y Últimos Reportes SR

CODMAQUINA	NOMBREMAQ	CODMODULO	NOMBREMOD	CODPARTE	NOMBREPARTE	CODMODODEFA	DESCRIPCION	CODCAUSAFALLA	DESCRIPCIONC	CODOPERARI	NOMBREOPE	FECHA	HORA
05	INSPECTOR DE 03	SISTEMA MESA 04	PARRILLA DE AP	H03	CAIDA DE BOTEL	F07	FALLA EN EL SUN JAM	JUAN BERNARDO	05/05/09	00:03:55			
07	ENVASADORA 2 02	SISTEMA DE EL 03	ELECTROVALVUL	H01	APARIENCIA INAJ	F05	DESCONFIGURAC	JULIO BENJAMIN	05/05/09	00:02:47			

- **Reportes Pendientes.** La tabla Reportes Pendientes es la encargada de guardar la información concerniente a la cantidad de reportes que se han recibido de los prototipos y los cuales están pendientes para ser impresos.

Su estructura esta conformada por un campo (Reportes pendientes) y un registro que varía a medida que se reciben o imprimen reportes. Debido a que cuando ingresa cierta cantidad de reportes a la base de datos el número almacenado aumenta, pero cuando se imprimen los informes esta cifra disminuye a cero.

Esta tabla no hace parte de la estructura relacional de la base de datos, ya que los datos almacenados en ella son completamente independientes de los registros de las demás tablas, por lo tanto en esta tabla no hay campo clave

**Figura 27.** Estructura Tabla Reportes Pendientes

REPORTESPENDIENTES : T	
REPORTESPENDIENTES	
▶	
*	

### **3. SOFTWARE DEL SISTEMA**

En este capítulo se describe el software de programación del sistema, tanto del acceso, manejo y visualización de los registros de la Base de datos desarrollada en *Microsoft ACCESS 2003*, proceso que se realiza en el PC utilizando *Visual Basic 6.0*. Como el de la programación del microcontrolador cerebro de la tarjeta en lenguaje de alto nivel (**Lenguaje C**).

#### **3.1. PROGRAMACIÓN DE LA INTERFAZ DE USUARIO EN VISUAL BASIC**

Con el fin de visualizar, procesar y actualizar los registros almacenados en la base de datos, se hizo necesario desarrollar una aplicación en Visual Basic 6.0 que permitiera establecer comunicación de forma serial con el prototipo y que a su vez fuera de fácil utilización para el usuario final.

Visual Basic permite diseñar una gran cantidad de aplicaciones con el fin de suplir necesidades existentes en la industria de una forma más sencilla que con otros lenguajes de programación, debido a que este software permite ejecutar ciertas labores mediante instrucciones gráficas sin necesidad de escribir código.

Visual Basic es un lenguaje basado en objetos, ya que para elaborar una aplicación se utilizan objetos con propiedades y métodos. Las propiedades comúnmente poseen valores lógicos o constantes apropiadas para cada una de ellas y definen tanto el aspecto gráfico (color, dimensión, posición, tipo y tamaño de letra), como la manera en que dichos objetos responden a las órdenes del usuario (si se activa o por el contrario se desactiva).

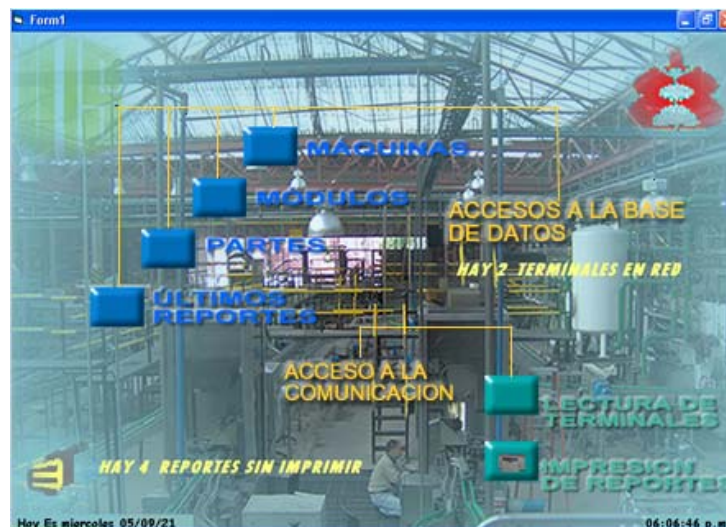
Los métodos son funciones que vienen preprogramadas; realizan tareas específicas y se pueden utilizar para cualquier aplicación con solo invocarlos. Este software también se caracteriza por ser orientado a eventos, pues para realizar algunas tareas es necesario esperar una acción del usuario tal como dar clic a un botón.

La aplicación elaborada consta de 16 formularios distribuidos de la siguiente forma:

- ❖ Para presentación de la aplicación, acceso a todas las ventanas de la interfaz e impresión de los reportes recibidos se ha destinado un formulario.
- ❖ Para manejo de la información almacenada en la base de datos, se utilizaron 14 formularios.
- ❖ Para establecer comunicación serial entre el prototipo y el PC, además de transmitir los reportes editados hasta la aplicación y por consiguiente a la base de datos, se reservó un formulario.

Adicionalmente en el proyecto de Visual se creo un módulo en el cual se programaron funciones públicas caracterizadas porque se puede acceder a ellas desde todos los formularios que conforman la aplicación.

**Figura 28** Interfaz De Usuario



**3.1.1 Acceso a la Base de Datos.** Es indispensable establecer conexión entre la Base de Datos y la aplicación en Visual Basic con el fin de tener acceso a la información almacenada en la Base de Datos y de esta forma lograr una interacción entre las dos partes.

Para tal fin en primera instancia se utilizó el Control *DATA* debido a que permite acceder a los datos definiendo las propiedades *CONNECT*, *DATA BASENAME* y *RECORDSOURCE* desde la ventana de propiedades (ver figura 29). La visualización de las tablas se realiza enlazando controles tales como *DBGRID* con la propiedad *DATASOURCE* al control *DATA*.

**Figura 29.** Control DATA



Entorno VISUAL BASIC 6.0

Al trabajar con éste sistema de acceso a datos, se encontró el inconveniente que al poner a funcionar la aplicación en WINDOWS XP se genera un error concerniente al motor para bases de datos Microsoft Jet y se comprobó que empleando este control la aplicación solo funciona correctamente para ACCESS 97 y WINDOWS 98, por lo tanto se vio la necesidad de cambiar el medio de acceder a los registros contenidos en la base de datos.

El ingreso a la base de datos desde la interfaz se realiza mediante el objeto de acceso a datos DAO (*Data Access Object*)<sup>10</sup>, debido a que proporciona compatibilidad con todas las

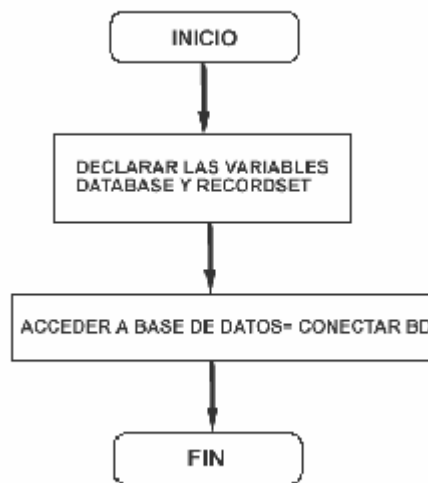
---

<sup>10</sup> [serie de objetos que conforman la estructura de un sistema de base de datos relacional en el cual se pueden realizar todas las operaciones necesarias para crear bases de datos, definir tablas campos e índices, establecer relaciones entre tablas, examinar y consultar la base de datos]

versiones de Microsoft, lo cual da gran confiabilidad y funcionalidad a la hora de instalar la aplicación.

Es preciso anotar que para emplear el método mencionado anteriormente, se requiere agregar la librería *DAO 3.6 object library* al proyecto que se está construyendo en Visual Basic 6.0 con el fin de acceder a cualquier tabla elaborada en ACCESS 2003 desde la aplicación. También es indispensable crear los objetos *Database*<sup>11</sup> y *Recordset*<sup>12</sup> para abrir la base de datos dando la ubicación exacta o “*Path*” de la misma y lograr manipular los datos contenidos en ella.

**Figura 30.** Diagrama de flujo correspondiente al acceso a la base de datos



- **Acceso y Visualización de las Tablas de la Base de Datos.** Mediante la aplicación se permite acceder y visualizar cada una de las tablas de la Base de

---

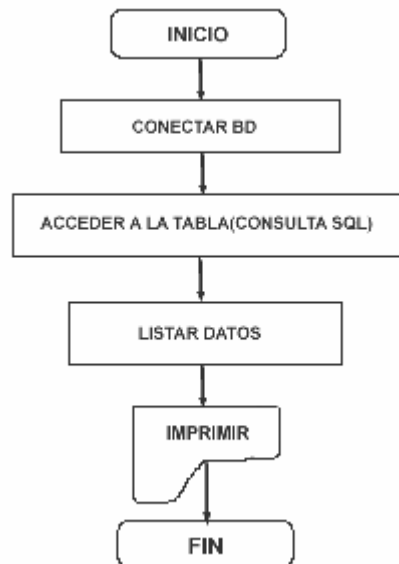
<sup>11</sup> [Representa una base de datos abierta, se utilizan sus métodos y propiedades para manipular la base de datos]

<sup>12</sup> [representa los registros de una base de datos relacional, o los registros que se generan al ejecutar una consulta y almacena los datos en registros y campos]

Datos de la siguiente forma. La conexión y acceso a cada una de las tablas se obtiene mediante la ejecución de una consulta SQL del tipo Select \* From a la tabla con la cual se desea trabajar.

La visualización de los datos se consigue por medio de la programación de una función que se encarga de llenar el control LISTVIEW con el resultado de los datos obtenidos al realizar la consulta Select \* From a la base de datos.

**Figura 31.** Diagrama de flujo para el acceso y Visualización de las tablas de la base de datos



**3.1.2 Manejo Dinámico de los Datos** Al realizar un análisis de las operaciones que se podían efectuar con las tablas que conforman la Base de Datos, se comprobó que estas no siempre permanecerían estáticas sino que por el contrario en algún momento se tendría la necesidad de hacer un movimiento dinámico de las mismas.

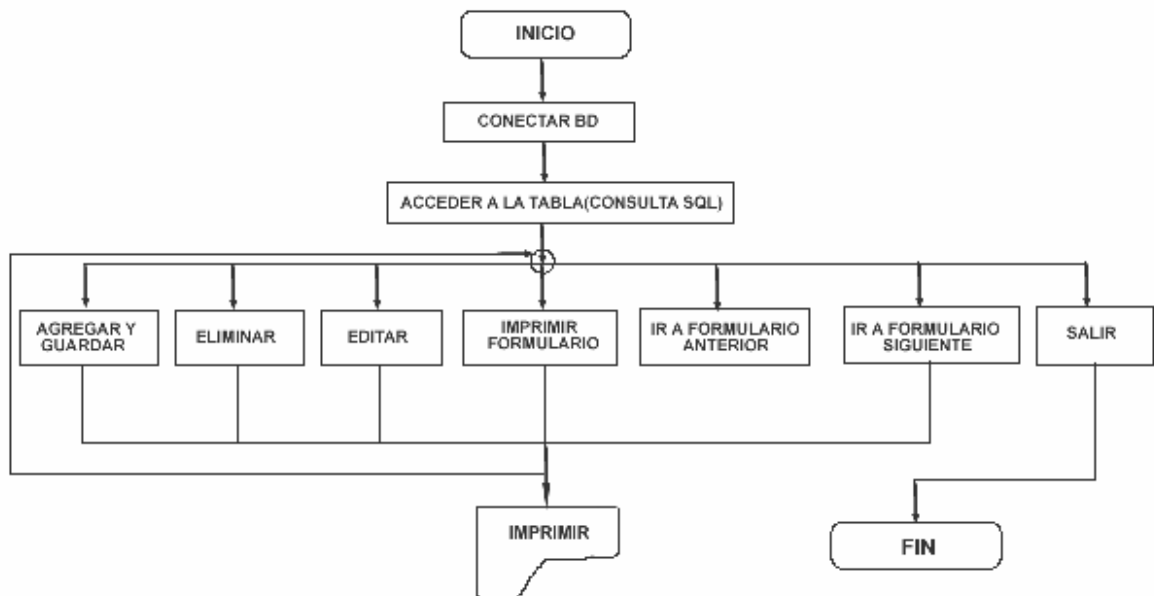
Por lo tanto fue preciso habilitar las funciones agregar, editar o eliminar registros en los formularios concernientes a cada una de las tablas Máquinas, Módulos y partes, debido a

que en estas se presenta la posibilidad, que en un momento determinado se requiera modificar la información almacenada en ellas. Con el fin de efectuar los procedimientos mencionados anteriormente y otros que faciliten la utilización de la aplicación para el usuario final, se dispuso de una barra de herramientas que contiene los elementos necesarios para ejecutar directamente las funciones: imprimir formulario, desplazarse a los formularios anterior o siguiente y salir de la aplicación. Así como también permite acceder a los formularios en los que se pueden efectuar las rutinas agregar y guardar, eliminar y editar (ver figura 32)

**Figura 32.** Barra de Herramientas Para Manejo Dinámico De Los Datos

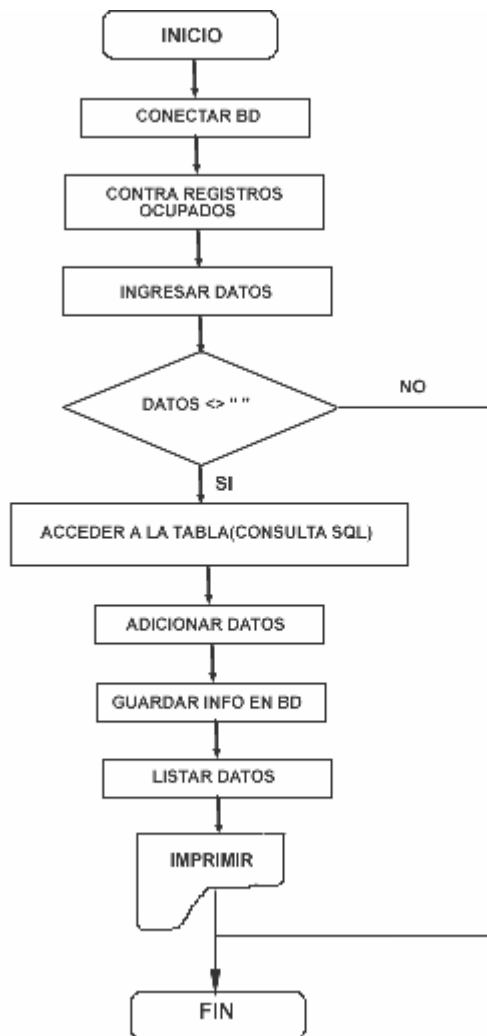


**Figura 33.** Diagrama de Flujo Para Manejo Dinámico de los Datos



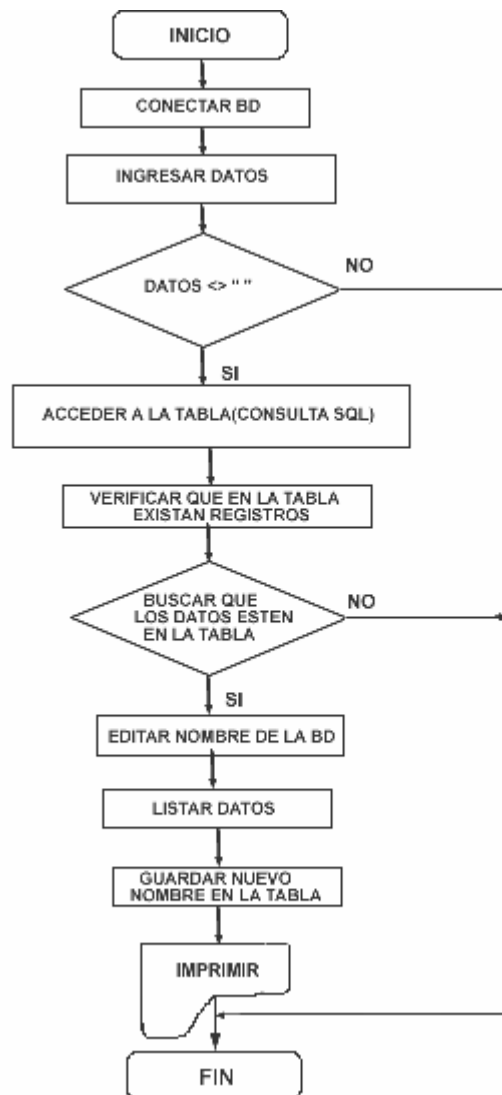
- **Función Agregar un Dato Y Guardarlo en la Base de Datos** Para sistemas como el que se está tratando en este trabajo de grado es preciso tener en cuenta que se puede instalar un nuevo componente (Máquina, Módulo o Parte). Por lo tanto se hace indispensable habilitar el proceso de insertar registros en la base de datos creando una rutina que lo ejecuta. Para tal fin se utilizaron los métodos AddNew y Update del objeto recordset ya que permiten agregar un nuevo registro a la tabla deseada, actualizar la información almacenada y grabarla en la base de datos

**Figura 34.** Diagrama de Flujo Agregar y Guardar



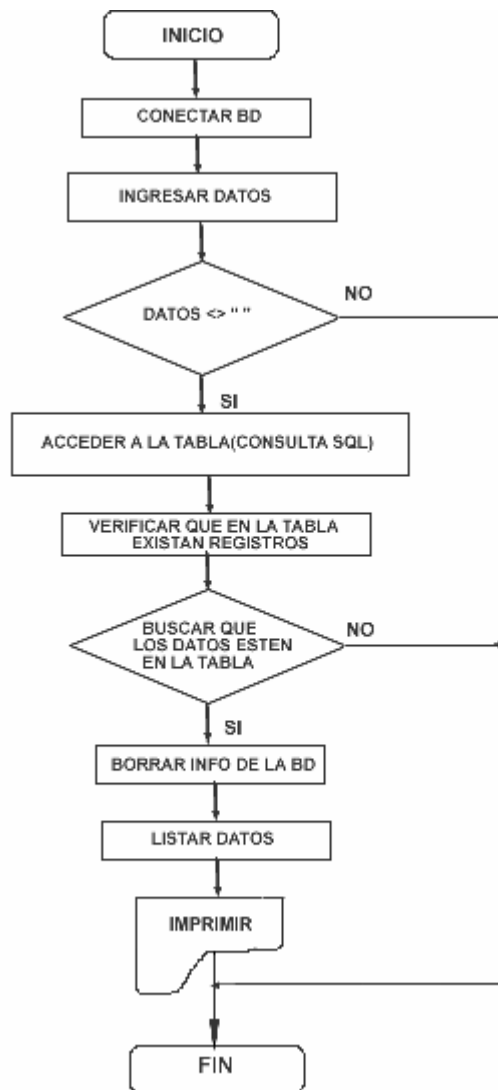
- **Función Editar un Registro.** En ciertas ocasiones se requiere editar el nombre de algún componente del tren de envase debido a que es inevitable reemplazarlo. Por razón a lo anterior se vio en la necesidad de programar este procedimiento utilizando los métodos *Edit* y *Update* del objeto Recordset que logran concretar la modificación de algún dato almacenado en la tabla sobre la cual se esta trabajando.

**Figura 35.** Diagrama de Flujo Editar




- **Función Eliminar un Registro.** Muchas veces se hace indispensable excluir información del sistema porque simplemente no se utiliza o porque para el caso específico de esta base de datos se elimina ya sea una máquina, un módulo o una parte del tren de envase. Por lo tanto en la rutina encargada de realizar esta labor se utilizó el método *Delete* del Recordset que permite suprimir un dato de forma definitiva de la base de datos.

**Figura 36.** Diagrama de Flujo Eliminar



**3.1.3. Comunicación con el Prototipo.** La comunicación serial con el prototipo se estableció mediante el control de comunicaciones con el que cuenta Visual Basic 6.0 (*MSComm*) para el cual es indispensable establecer mediante código los valores para las propiedades (*CommPort*), (*Settings*), (*InputLen*), (*PortOpen*) (ver figura 37) en las cuales se determinan condiciones para la transmisión de los datos tales como: el puerto serie que se ocupará, la velocidad a la cual se transmitirán los datos, el tipo de paridad que se manejará, el número de bits de datos que se utilizarán para formar un bloque, la cantidad de bits empleados para indicar que se acaba de recibir un bloque de datos y el número de caracteres que leerá el búfer de entrada.

**Figura 37.** Control Mscomm



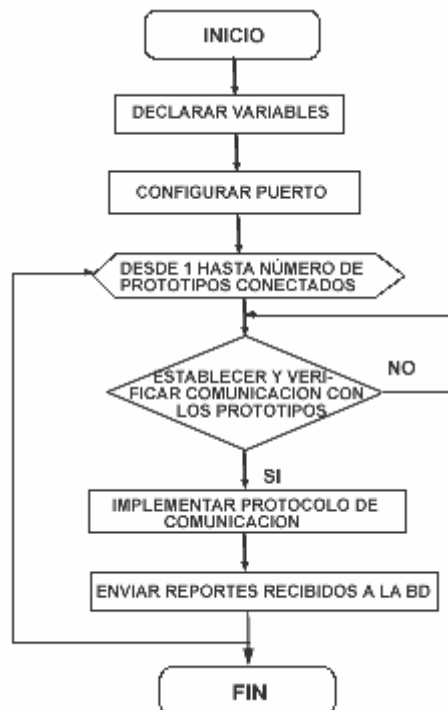
```
MSComm1.CommPort = 1
MSComm1.Settings = "9600,n,8,1"
MSComm1.InputLen = 1
MSComm1.PortOpen = True
```

Fuente: Entorno VISUAL BASIC 6.0

A medida que los caracteres de la trama diseñada en el protocolo se van recibiendo se realiza un proceso que permite la asociación de los códigos provenientes de los reportes editados en el dispositivo con los nombres que los identifican para de esta forma ir ingresando la información recibida en el campo que corresponda (máquina, módulo, parte, modo de falla, causa de falla y operario) de las tablas Últimos Reportes y Últimos ReportesSR existentes en la base de datos, la primer tabla organiza los registros según la máquina en la cual ocurrieron las fallas, debido a que esta relacionada y por lo tanto distribuye los registros de acuerdo a la primer clave que en este caso corresponde al código de la máquina; mientras que en la segunda tabla se organizan los registros según el orden de fecha y hora en el que se editaron los reportes, pues esta tabla no esta relacionada con ninguna de las otras y por lo tanto introduce los datos en el mismo orden en el que se reciben del equipo de tal forma que se coloca el último registro recepcionado al final de la tabla.

Otro aspecto que cabe destacar en esta parte, consiste en la creación de una rutina encargada de enviar y actualizar en la tabla Datos de Visual, la cantidad de reportes que ingresaron a la base de datos después de concluir la transmisión

**Figura 38.** Diagrama de Flujo para Comunicación Serial

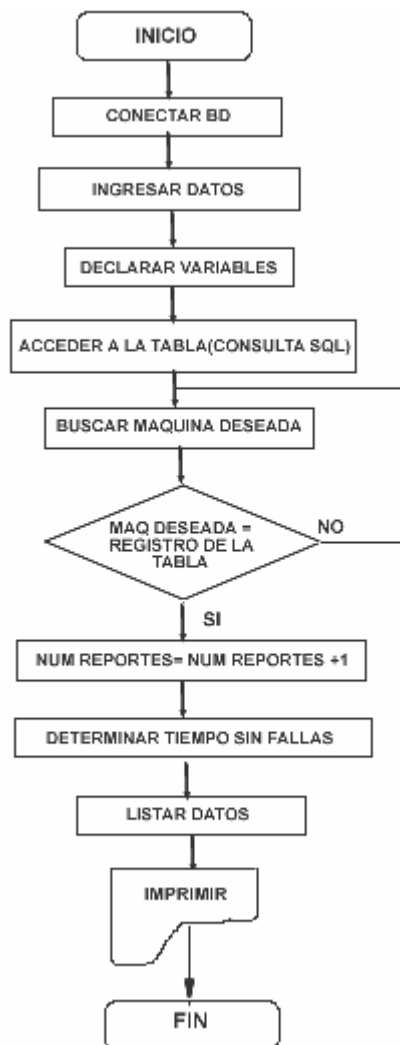


**3.1.4. Visualización Reportes Almacenados.** Una vez que el proceso de comunicación haya terminado y los nuevos reportes se encuentren dentro de la base de Datos, se accede a la tabla donde se almacenaron, mediante la ejecución de una consulta del tipo `Select * From`, para luego buscar mediante comparaciones, los reportes de la máquina que se desee. Adicionalmente se programo una pequeña rutina para que en el caso que se quiera observar una mayor cantidad de reportes que los que han ingresado a la Base de Datos, aparezca una etiqueta informando al usuario el número total de informes existentes para esa máquina.

Simultáneamente esta rutina se elaboro de tal forma que se determine el tiempo que la máquina ha trabajado sin fallas, restando la fecha actual con la fecha del último reporte encontrado.

La visualización de los reportes se obtiene por medio de la programación de una función que se encarga de llenar el control LISTVIEW con el resultado de la búsqueda de reportes. Mientras que el tiempo de trabajo sin fallas se visualiza en una etiqueta.

**Figura 39.** Diagrama de Flujo visualización de Reportes.



**3.1.5 Impresión de Reportes.** Para efectuar la impresión de los reportes que se descargan del prototipo se utilizó el objeto *PRINTER* de Visual Basic 6.0 ya que este permite que la aplicación establezca comunicación con la impresora configurada en el sistema y le envíe la información que se desea imprimir. Para realizar el proceso mencionado anteriormente se tuvieron en cuenta aspectos tales como: calidad de la impresión, número de veces que se imprimirá el documento y tamaño de letra que se utilizará; todos estos parámetros se configuraron mediante las propiedades *Copies*, *SCALEMODE*, *FONTSIZE*, *PRINTQUALITY*, *COLORMODE* del objeto *PRINTER* (ver figura 40).

**Figura 40.** Determinación de las Propiedades para el Objeto *PRINTER*

```
Printer.Copies = 1  
Printer.ScaleMode = 4  
Printer.FontSize = 17  
Printer.PrintQuality = -1  
Printer.ColorMode = 1
```

Conjuntamente con lo anterior se tuvo en cuenta el hecho de que pueden existir varios reportes acumulados en la base de datos que todavía están sin imprimir, por lo tanto para establecer cuantos y cuales reportes están pendientes para impresión, se dispuso que la aplicación realizara una consulta a la tabla Datos de Visual e informara al usuario final mediante una etiqueta esta cifra (ver figura 41)

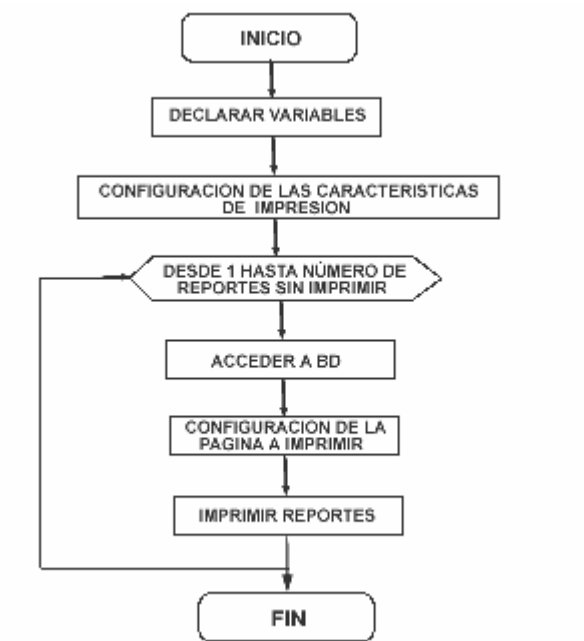
**Figura 41.** Etiqueta que Informa la Cantidad de Reportes Pendientes para Impresión

**HAY 4 REPORTES SIN IMPRIMIR**

Otro aspecto que se analizó fue la organización de cierta cantidad de reportes en una hoja ya que sería un desperdicio imprimir cada reporte por separado. Para lo cual se hizo una

función que distribuyera mediante coordenadas hasta 8 reportes como máximo en una hoja, además se estableció que cuando el número de reportes pendientes excediera del máximo se continuara imprimiendo en otra hoja y al no detectar más reportes pendientes se culminara el proceso haciendo que la cifra almacenada en la tabla Datos de Visual volviera a cero

**Figura 42.** Diagrama de Flujo Imprimir Reportes



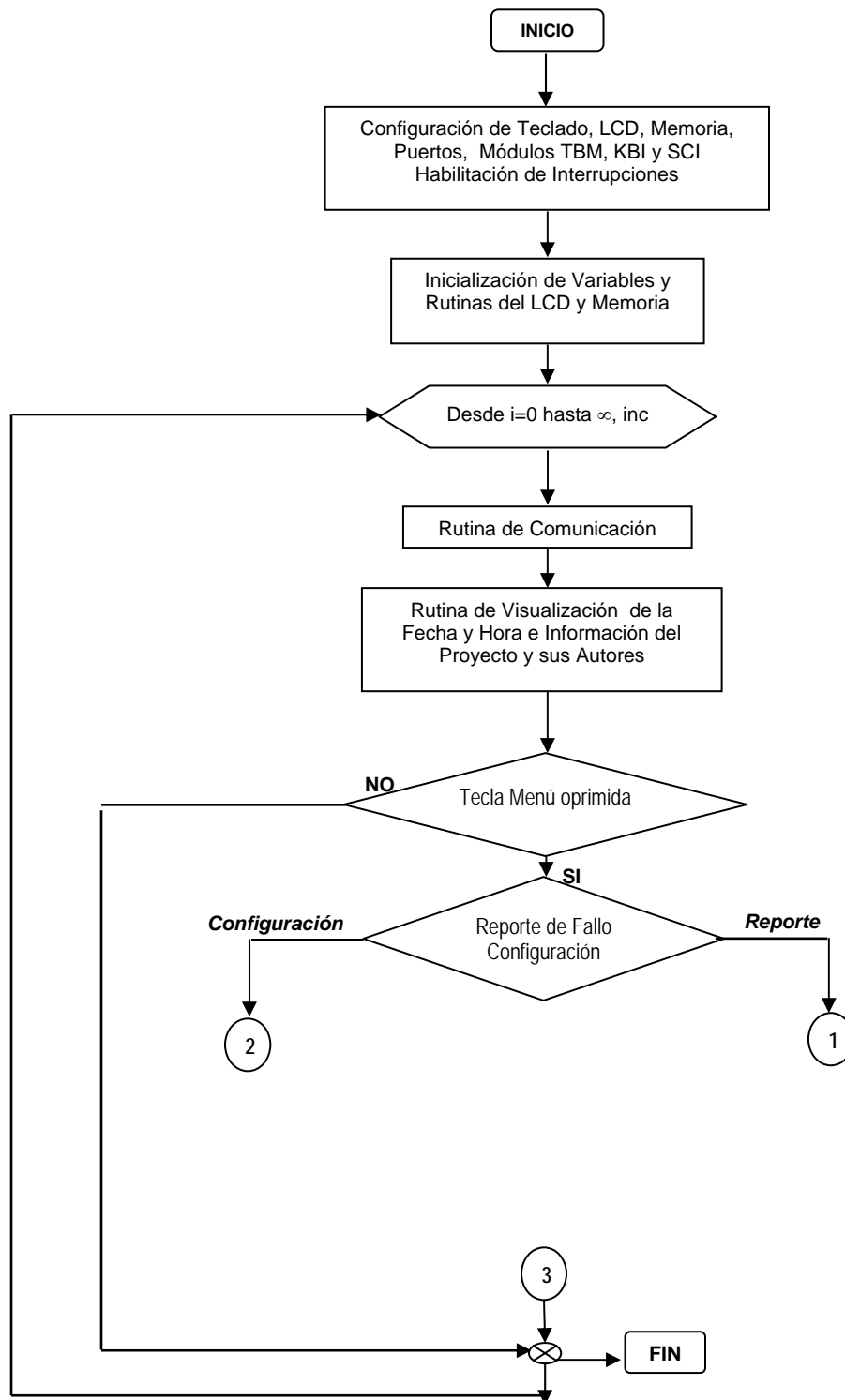
## 3.2. PROGRAMACIÓN DEL MICROCONTROLADOR

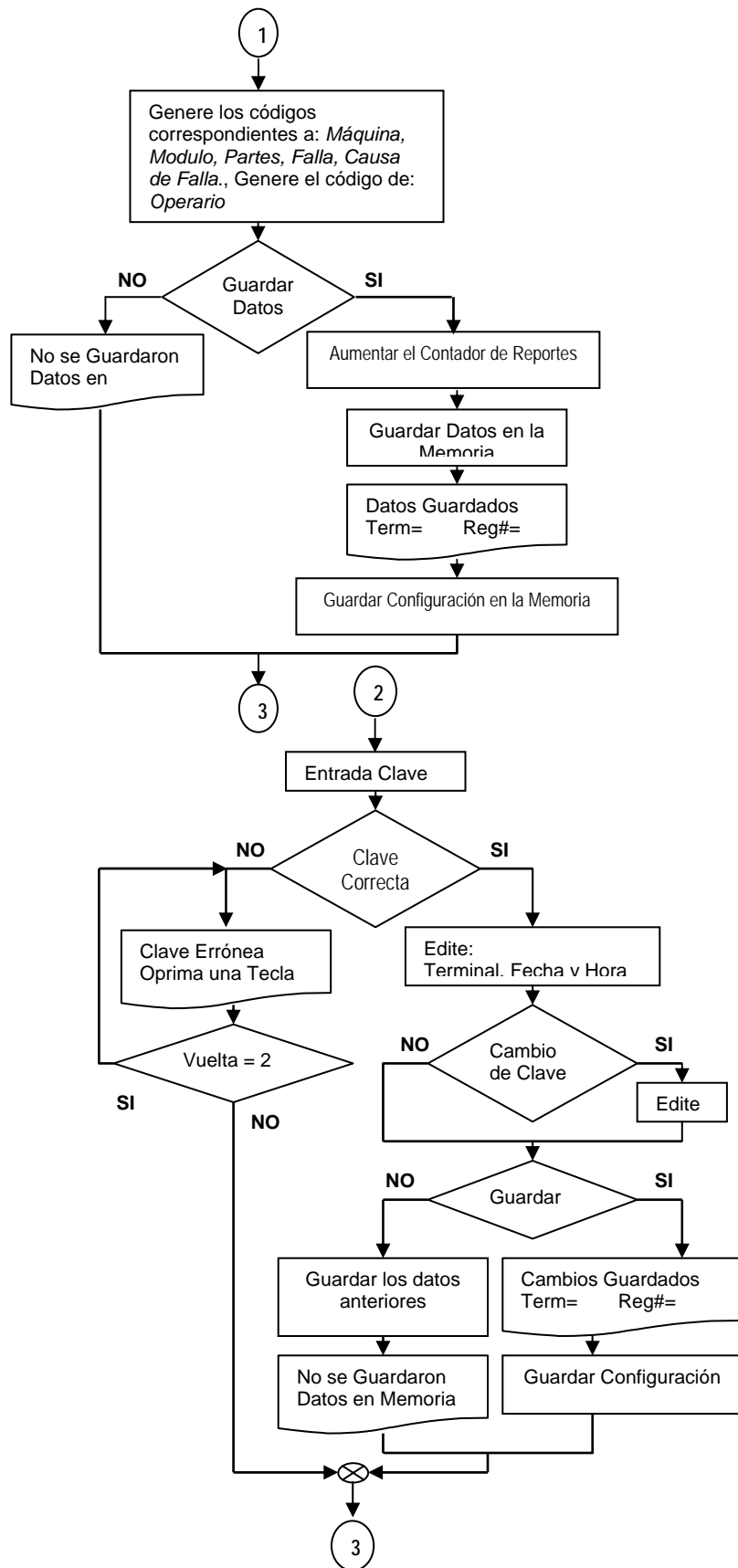
Para entender mejor los programas del microcontrolador y no complicar al lector con códigos difíciles de asimilar; a continuación se plantearán los algoritmos utilizados, las subrutinas del programa principal y las encargadas del manejo de los dispositivos periféricos junto con los diagramas de estados y tiempos; así como la explicación de las Interrupciones empleadas.

**3.2.1. Algoritmo general del programa.** Para el manejo de los periféricos del microcontrolador fue necesario seguir los siguientes pasos para configurar el equipo, traducidos a instrucciones ejecutadas por el microcontrolador:

- ❖ Definición de variables.
- ❖ Configuración del Teclado, de la LCD y de la Memoria Serial EEPROM.
- ❖ Configuración de los módulos: SCI y TBM.
- ❖ Inicialización del Módulo LCD.
- ❖ Leer configuración inicial del programa.
- ❖ Acuse de cualquier interrupción pendiente.
- ❖ Inicializar el contador y Activar interrupciones del módulo TBM.
- ❖ Habilitar Pull-up para los pines PTA7 y PTC2.
- ❖ Rutina para visualizar en la LCD la fecha y la hora, e información correspondiente a los autores del proyecto.
- ❖ Escuchar el canal para el llamado del PC y en dado caso iniciar Rutina de Comunicación con éste. Para programar esta rutina fue necesario utilizar las siguientes subrutinas:
  - ☑ Enviardatos\_pc: Coloca en el registro SCDR (*SCI Data Register*) los datos que serán enviados al PC y hace un sondeo para determinar si el transmisor está lleno
  - ☑ Wakeup: Recibe la fecha y la hora que envía el PC y la actualiza en el programa principal para que pueda ser visualizada en la LCD.

**Figura 43.** Algoritmo general de configuración y funcionamiento del Prototipo





- ☑ Transm\_nregistros: Permite enviar el número total de registros, almacenados en la memoria externa, que se Transmitirán al PC.
- ☑ Leer \_ enviar: Esta rutina lee secuencialmente los datos del Reporte almacenado en la memoria externa y los envía al PC a través del módulo serial
- ☑ Confirmación: Permite recibir la confirmación desde el PC de que todos los reportes enviados llegaron. Si la confirmación es satisfactoria, entonces se da la orden de sobre escribir en la memoria y guardar la nueva configuración.
  
- ❖ Esperar la interrupción del teclado para desplazarse por el Menú y seleccionar la opción deseada (Edición de Reporte de Fallo ó Configuración del Prototipo). Para ejecutar la rutina correspondiente al Menú, fue necesario utilizar las siguientes subrutinas:
  - ☑ Detectar\_tecla: Rutina que detecta cuando una tecla es oprimida.
  - ☑ Esperar\_tecla: Rutina que evalúa y determina cual de las cuatro teclas (MENU, OK, UP o DOWN) fue oprimida.
  - ☑ Edición: Rutina que le permite al usuario Ingresar los códigos correspondientes a: **Máquina, Módulo, Parte, Modo de Falla, Causa de Falla y Operario**; utilizando las teclas UP y DOWN para navegar a través del carácter y encontrar el carácter alfabético o numérico deseado, y OK para desplazar el cursor a lo largo de la segunda línea de la LCD. En la primera línea de la LCD se visualiza el nombre correspondiente a cada código a medida que el reporte se va creando y en la segunda línea se visualiza el código generado.
  - ☑ Edición2: permite al usuario modificar **Hora** y **Fecha**, las cuales se visualizan siempre al inicio de la segunda línea de la LCD mientras no se haya ingresado al Menú. Esta modificación se realiza de la misma forma como se editó el Reporte de Fallo.
  - ☑ Edición3: Esta rutina le permite al usuario editar el nombre del **Terminal**, de la misma manera como se editaron los códigos del Reporte de Fallo.
  - ☑ Edición4: Con esta rutina se permite al usuario decidir si guarda o no los cambios hechos a la configuración que inicialmente tenía el prototipo.
  - ☑ Entrada\_clave: Con la creación de esta rutina se permite al usuario modificar la **Clave de Acceso** al menú de Configuración del Prototipo. Esta modificación se realiza de la misma forma como se editó el Reporte de Fallo.

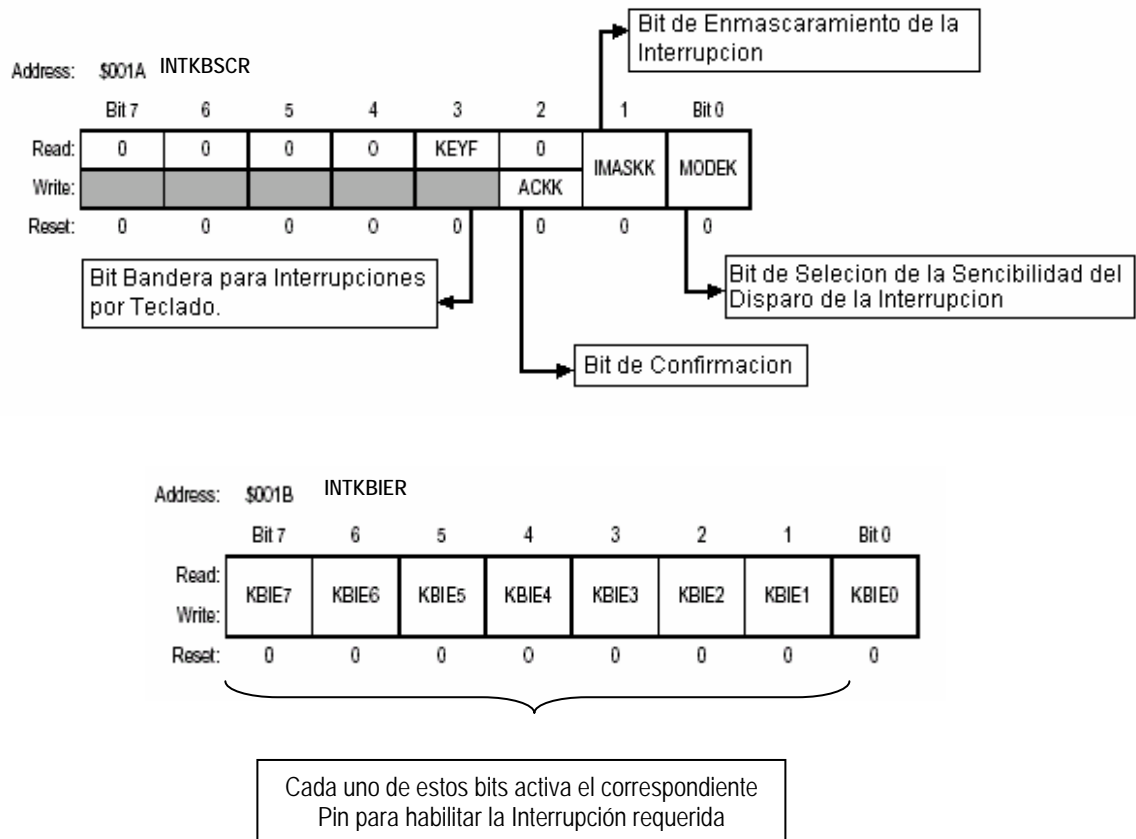
- ☑ **Password:** Esta rutina determina si la clave ingresada por el usuario para acceder al menú de Configuración del Prototipo es o no correcta. También le permite al usuario equivocarse 3 veces antes de salirse de dicho menú.
- ☑ **Leer\_config:** Con esta rutina se puede leer los últimos datos almacenados en la memoria externa, de tal forma que cuando se edite una nueva Configuración del Prototipo y ésta no sea guardada, al llamar a esta rutina se pueden volver a almacenar estos últimos reportes.
- ☑ **Guardar\_config:** Esta rutina guarda la nueva configuración que el usuario ha decidido crear.

**3.2.2. Subrutinas Especiales del Programa Principal.** En esta sección se explicaran detalladamente las subrutinas que incluyen los dispositivos periféricos del Microcontrolador así como las interrupciones utilizadas.

- **Configuración de Registros Especiales.** Como se mencionó en el capítulo de Diseño e Implementación del Sistema, la función principal del Microcontrolador es asegurar que la edición, almacenamiento y transmisión de los reportes de fallo de las máquinas sea correcta, por tal razón se utilizaron las siguientes interrupciones con el fin de optimizar el programa del Microcontrolador.
- ❖ **Interrupción de Teclado.** Se habilitó una interrupción externa por puerto, útil para el teclado, de tal forma que al oprimir la tecla MENÚ el programa salta e ingresa a la rutina Menú, en la cual el usuario puede seleccionar entre la Edición del Reporte de Fallo ó la Configuración del Prototipo.

Para tal fin se empleó el puerto A del Microcontrolador, puesto que éste dispone de 8 pines con los cuales se puede acceder al Módulo KBI (*Keyboard Interrupt Module*). La configuración del Módulo KBI se hace posible por medio de los siguientes registros INTKBSCR (*Keyboard Status and Control Register*) y INTKBIER (*Keyboard Interrupt Enable Register*) (Ver figura 44)

**Figura 44.** Registros INTKBSCR y INTKBIER



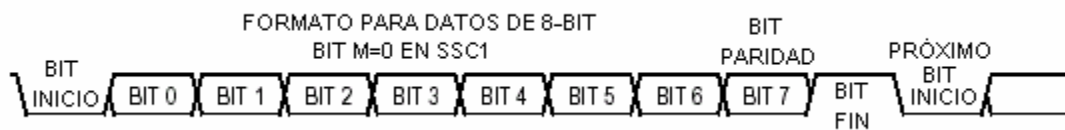
Fuente: Hoja de Datos del Fabricante.

- ❖ **Comunicación Serial.** Los Reportes de Fallo editados y almacenados en el prototipo deben ser descargados continuamente para su posterior estudio y análisis en una Base de Datos creada en un PC remoto; razón por la cual fue necesario realizar la Transmisión de datos de forma Serial. Así de ésta forma se puede construir una pequeña red industrial con la menor cantidad de cables posibles.

Hacia tal propósito se implementó el Módulo SCI (*Serial Communications Interface Module*) del microcontrolador. Éste es una interfaz serie útil para comunicaciones asíncronas de alta velocidad con otros dispositivos periféricos. El Transmisor y

Receptor de éste módulo operan independientemente, aunque ellos usan el mismo generador de la tasa de baudios. Durante una operación normal la CPU monitorea el estado del SCI, escribe el dato a ser leído y procesa el dato recibido, para tal fin utiliza el registro SCDR (*SCI Data Register*). El SCI usa el estándar NRZ (*Non-Return-to-Zero mark/space*) (ver figura 45).

**Figura 45.** Formato de Datos SCI para datos de 8 bits



Fuente: Hoja de datos del fabricante.

Los pines de I/O del Módulo SCI son compartidos con el puerto E del microcontrolador (Ver tabla 4).

**Tabla 4.** Convenciones de los nombres de los pines

<b>Nombre Genérico del Pin</b>	TxD	RxD
<b>Nombre completo del Pin</b>	PTE0/TxD	PTE1/RxD
<b>Asignación de pin en el Encapsulado P-DIP de 40-Pin</b>	12	13

Fuente: Hoja de datos del fabricante.

La tasa de baudios para el transmisor y el receptor se puede seleccionar a través de la configuración de los bits SCP1, SCP0, y SCR2, SCR1, SCR0 del Registro de Tasa de Baudios (SCBR). La tasa se puede calcular utilizando la siguiente fórmula:

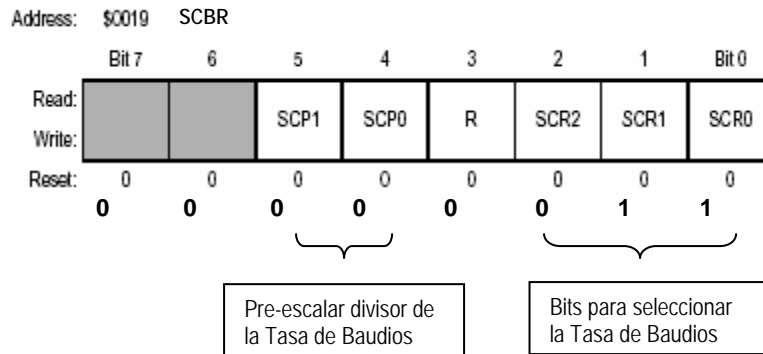
$$\text{Baudios} = \frac{\text{SCI Clock Source}}{64 \times PD \times BD} \quad \text{SCI Clock Source} = f_{BUS} \text{ o } CGMXCLK$$

*PD = Divisor Pr escalar.*

*BD = Divisor de la Rata de Baudios.*

Para este caso se seleccionó una tasa de 9600 baudios, por lo tanto el registro será igual a: SCBR = 03 h = 00000011 b. (Ver figura 46).

**Figura 46.** Registro SCBR (*SCI Baud Rate Register*) y Tabla de divisores.



Fuente: Hoja de Datos del Fabricante.

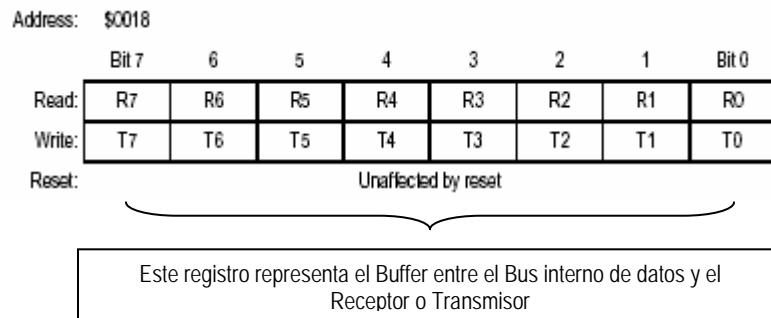
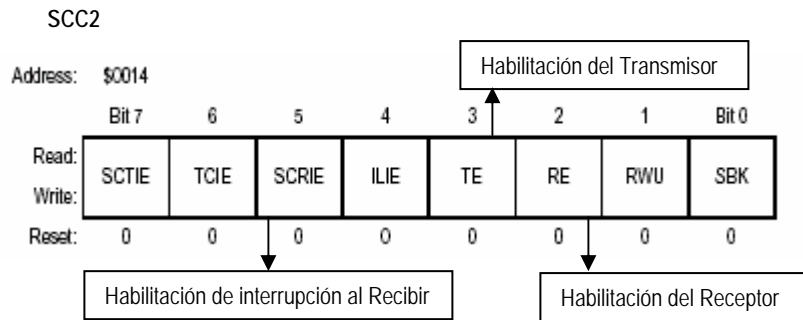
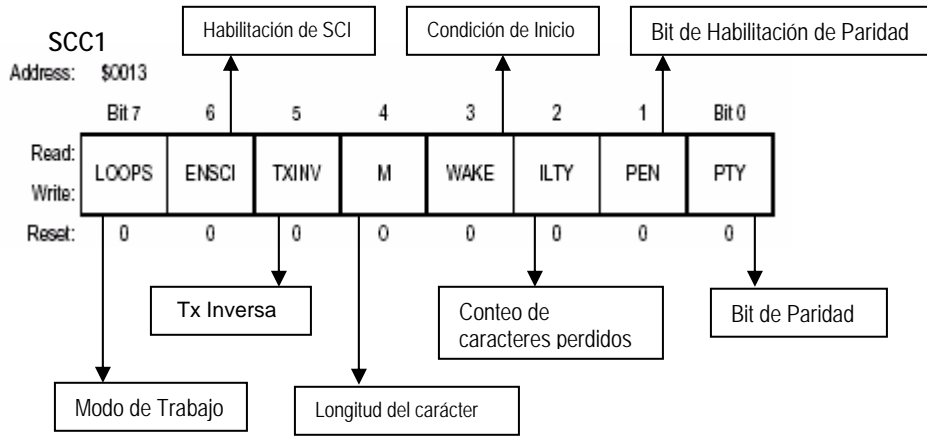
**Tabla 5.** Selección de la Tasa de Baudios

SCP1 & SCP0	Divisor Pre-escalar	SCR2,SCR1 y SCR0	División de la Tasa de Baudios
00	1	000	1
01	3	001	2
10	4	010	4
11	13	011	8
		100	16
		101	32
		110	64
		111	128

Fuente: Hoja de Datos del Fabricante.

La configuración del Módulo se realiza por medio de los registros SCBR (*SCI Data Register*), SCC1 (*SCI Control Register 1*), SCC2 (*SCI Control Register 2*) (Ver figura 47).

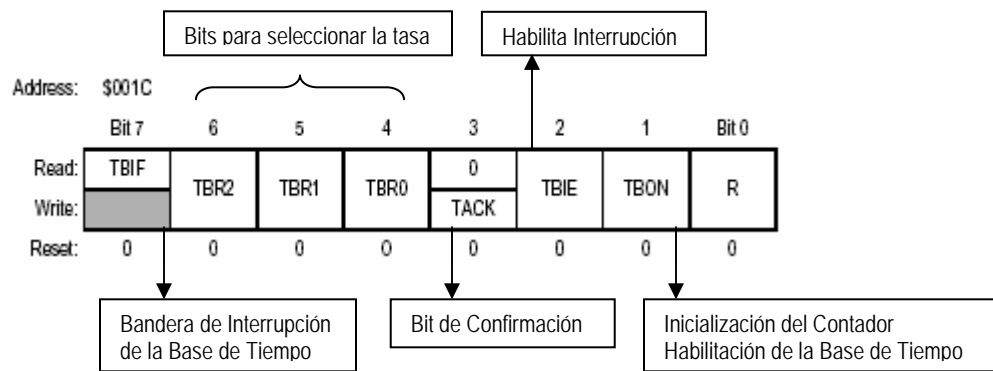
**Figura 47. Registros SCC1, SCC2 y SCDR**



Fuente: Hoja de Datos del Fabricante.

- ❖ **Base de Tiempo.** Para programar el Reloj de Tiempo Real, el cual se visualiza al inicio de la segunda línea de la LCD, fue necesario utilizar el Módulo TBM. Al configurar éste módulo se pueden generar interrupciones periódicas con las que se logra controlar un contador que genera el reloj del sistema. Este módulo cuenta con un único registro, el TBCR (*Timebase Control Register*), que es usado para habilitar las interrupciones de la base de tiempo y fijar la tasa de dicho contador. (Ver figura 48).

**Figura 48.** Registro TBCR.



Fuente: Hoja de datos del Fabricante

- **Rutina de Lectura y Escritura de la Memoria.** La memoria 24FC256, soporta un bus bidireccional de dos cables y un protocolo de Transmisión, compatible con el protocolo I2C. El bus debe ser controlado por el dispositivo Maestro (Microcontrolador) el cual genera el Reloj Serial (SCL), el control de acceso al bus y genera las condiciones de STOP y START, mientras la memoria trabaja como un esclavo. Ambos, maestro y esclavo pueden operar como Transmisor<sup>13</sup> o Receptor<sup>14</sup>, pero el dispositivo maestro determina cual modo es activado.

Los pines PTC2 y PTC3 del microcontrolador son conectados al pin SDA y SCL de la memoria respectivamente, y por medio de estos se programa el protocolo I2C.

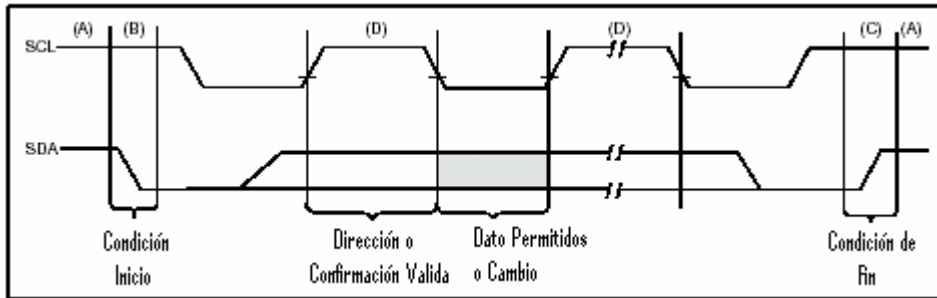
<sup>13</sup> Un dispositivo transmisor está definido como aquel que envía datos sobre el Bus.

<sup>14</sup> Un dispositivo Receptor está definido como aquel que recibe los datos de el Bus.

❖ **Características del Bus.** Se han definido las siguientes características para el protocolo del Bus:

- ☑ La transferencia de Datos puede ser iniciada sólo cuando el Bus no esté ocupado.
- ☑ Durante la transferencia de Datos, la línea de Datos (SDA) debe mantenerse estable siempre que la línea de reloj sea ALTO. Los cambios en la línea de Datos siempre que la línea de reloj sea ALTO serán interpretados como condiciones de START o STOP. (figura 49)

**Figura 49.** Secuencia de Transferencia de Datos en el Bus Serial



Fuente: Hoja de datos del Fabricante

Se han definido las siguientes condiciones del Bus:

- ☑ Bus no ocupado: Las líneas de reloj y datos se mantiene en ALTO<sup>15</sup>
- ☑ Inicio de la transferencia de DATOS (Start): Una transición de ALTO a BAJO en la línea SDA mientras SCL es ALTO. Todas la operaciones deben ser precedidas por una condición de Start
- ☑ Fin de la transferencia de Datos (Stop): Una transición de BAJO a ALTO de la línea SDA mientras SCL es ALTO. Todas las operaciones deben terminar en una condición de STOP.
- ☑ Datos Válidos: El estado de la línea de datos representa un DATO válido cuando después de una condición de Start la línea de DATOS es estable para la duración de un periodo ALTO de la señal de reloj. Los DATOS en la línea deben ser

<sup>15</sup> Al inicio del programa se habilitan los dispositivos internos de *pullup* para los pines PTC y PTC3 del Microcontrolados

cambiados durante un periodo BAJO de la señal de reloj. Hay un bit de DATOS por pulso de reloj.

- ☑ Confirmación: Cada dispositivo cuando es direccionado está obligado a generar una señal de confirmación después de la recepción de cada byte. El dispositivo Maestro debe generar un pulso de reloj extra el cual es asociado con los bits de confirmación. Un dispositivo que confirma debe tener un BAJO estable en la línea SDA durante el pulso de confirmación del reloj.
- ☑ Tiempo de Confirmación: El transmisor debe liberar la línea SDA en éste punto, permitiendo al receptor llevar a bajo la línea SDA para confirmar los 8 bits de DATOS; el receptor liberará la línea SDA en éste punto, entonces el transmisor puede continuar enviando DATOS.
- ☑ Byte de Control: Es el primer Byte que se recibe desde el dispositivo Maestro, después de la condición de START (Figura 50).

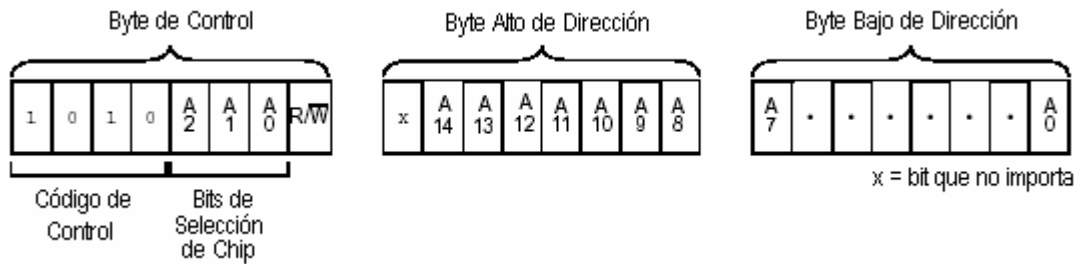
**Figura 50.** Byte de Control.



Fuente: Hoja de Datos del Fabricante.

- ☑ Bytes de dirección: Son los siguientes dos Bytes recibidos por la memoria y corresponden al Byte más significativo y al menos significativo de la dirección del primer Byte de Datos. (Figura 51)

**Figura 51.** Asignamiento de la secuencia de los bits de dirección

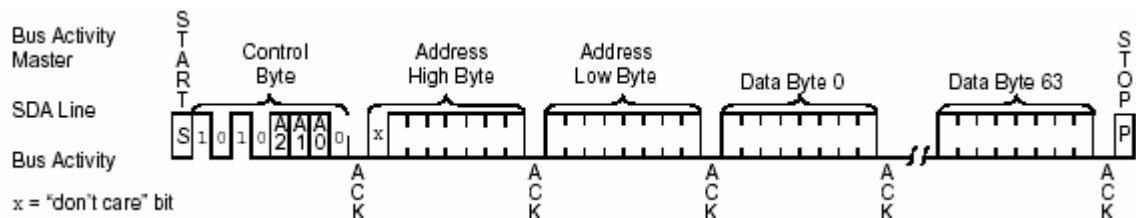


Fuente: Hoja de Datos del Fabricante

Con el fin de acceder a la Memoria y escribir o leer en ella fue necesario crear las siguientes subrutinas del programa principal, para esto se estudiaron tanto las características de estados y tiempos de la memoria, como el concepto del protocolo I2C.

- ❖ **Subrutina de Escritura.** Después de que la Memoria recibe del Maestro el byte de control con el bit R/W en 0 y los dos bytes de Dirección, que le indican desde que posición de la memoria debe comenzar a escribir, envía un bit de confirmación. Cuando el Maestro recibe la confirmación transmite la secuencia de palabras que serán escritas desde la posición direccionada de la Memoria. Al finalizar la transferencia de DATOS, el dispositivo Maestro recibe una confirmación y envía una condición de STOP (figura 52).

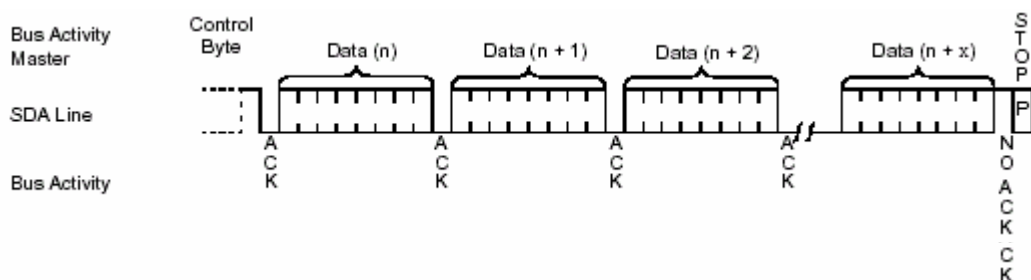
**Figura 52.** Escritura secuencial para páginas de 6 bytes.



Fuente: Hoja de Datos del Fabricante.

- ❖ **Subrutina de Lectura.** La operación de lectura es indicada como la operación de escritura, con la excepción que el bit R/W del byte de control es fijado a 1 (figura 53).

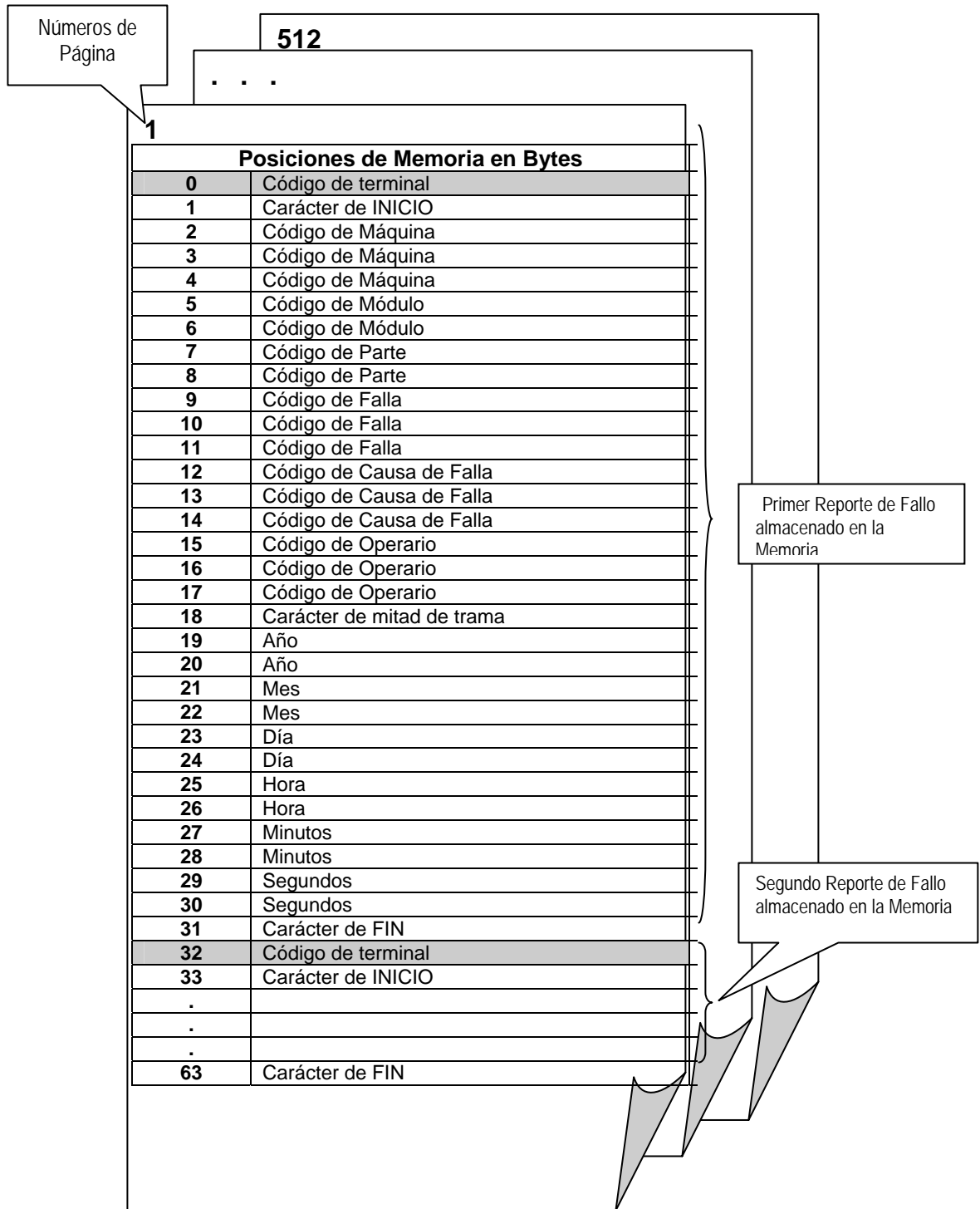
**Figura 53.** Lectura



Fuente: Hoja de Datos del Fabricante.

La memoria EEPROM 24FC256 esta distribuida en 512 páginas las cuales pueden almacenar hasta 64 bytes de datos para un total de 32K bytes de capacidad de almacenamiento. Para esta aplicación la memoria se organizó de la siguiente manera (figura54).

**Figura 54.** Distribución de la Memoria





#### 4. CONCLUSIONES

- Se profundizó en líneas de investigación de la Industria, en especial en el campo de Mantenimiento Industrial, junto con las metodologías para la pronta solución a problemas de administración de Reportes de Fallo de las máquinas; tales como recolección de informes, manipulación de la información y almacenamiento de la misma.
- Se logró construir un Prototipo funcional diseñado especialmente para cumplir con las necesidades de la Sección de Envase de Bavaria Cervecería de Boyacá, en materia de adquisición de Reportes de Fallo, a muy bajo costo, de fácil manipulación e instalación para los usuarios finales.
- Se diseñó una Interfaz práctica de menús en el prototipo con la cual se pueden generar, almacenar los Reportes de Fallos fácil y rápidamente.
- Se desarrolló una Red Industrial utilizando el estándar RS-485 con protocolo propio, diseñado exclusivamente para los Prototipos que funcionan como esclavos de un PC el cual controla el acceso al medio y almacena los reportes en una base de datos.
- Se elaboró una Base de Datos relacional en la cual se almacena información concerniente a las máquinas, a sus componentes y a las fallas que se produzcan en ellas. Con esto se logró obtener un archivo sistematizado, organizado tanto por la fecha en la que ocurrió la falla, como

por la máquina en la que se presentó la avería. Lo cual sirve como apoyo, a la hora de analizar el comportamiento de las máquinas, a fin de elaborar los reportes de mantenimiento mensual; así como las estrategias de mantenimiento preventivo y programado.

- Se logró diseñar y construir una aplicación en Visual Basic 6.0, de fácil utilización para el usuario final, que permite el manejo de la información contenida en la Base de datos y que es capaz de establecer comunicación serial entre el PC y los prototipos conectados en red, a fin de descargar los reportes que se encuentran allí almacenados e ingresarlos a la base de datos.
  
- Para obtener comunicación de tipo serial entre los prototipos y el PC, se utilizó el control de comunicaciones de Visual Basic 6.0, para el cual, es necesario configurar los parámetros: velocidad de transmisión, bit de paridad, bit de datos y bit de parada, a fin de que la trama proveniente del prototipo sea transmitida sin inconvenientes y llegue correctamente a la aplicación.
  
- Se comprobó que la utilización del objeto DAO (Objeto de Acceso a Datos) y sus librerías, como medio de acceso a la información contenida en la Base de Datos , proporciona versatilidad a la hora de instalar la aplicación en cualquier PC, sin importar el sistema operativo instalado en el PC de trabajo, debido a que DAO presenta compatibilidad con todas las versiones de WINDOWS®

## BIBLIOGRAFIA

- DACKAR HENRY, MS-Access avanzado, Producciones Henry Dackar, Colombia 1996
  
- CORREA URIBE GUILLERMO, Desarrollo de algoritmos y sus aplicaciones en Basic, Pascal, Cobol y C, tercera edición, Mc. Graw Hill, Santafé de Bogotá 1997.
  
- GOTTFRIED BYRON S. Programación en C segunda edición, Mc. Graw Hill. Madrid España 1997
  
- MOTOROLA digital dna, MC68HC908GP32 Technical Data.
  
- HALVORSON MICHAEL “Aprenda Microsoft Visual Basic Ya”.
  
- Normas:  
ISO 9000, Año 2000. Norma para el control de calidad.  
ANSI/TIA/EIA568A. Norma cableado estructurado.  
ISO/ DIS 14224.
  
- <http://www.grupoempresarialbavaria.com/> (GRUPO EMPRESARIAL BAVARIA.):

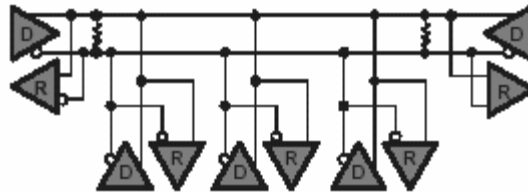
- [http://www.rs485.com/rs422-485\\_application\\_note\\_appendix\\_b\\_eia\\_standard\\_rs-423\\_da.htm](http://www.rs485.com/rs422-485_application_note_appendix_b_eia_standard_rs-423_da.htm) (Protocolo RS48-422):
  
- <http://www.analog.com/> (Analog Devices):
  
- <http://www.ti.com/> (Texas Instruments):
  
- <http://www.motorola.com/semiconductors> (Motorola Inc.):
  
- <http://www.elGuille.com>
  
- <http://www.Monografia.com>

## ANEXO A ESTANDAR RS 485

### Estándar TIA/EIA – 485

Debido a la falta de la capacidad para permitir conexiones bidireccionales multipunto en un Bus (ver figura 1A), se creó éste estándar para así satisfacer esta característica. El nuevo estándar (TIA/EIA-485 o ISO/IEC 8284) define características eléctricas de interconexión incluyendo *driver*, línea, y *receiver*. Éste permite tasas de datos en el rango de los 35 Mbps y superiores, y permite longitudes de línea hasta de 1200 m; por supuesto ambos límites no pueden ser alcanzados al mismo tiempo. Las recomendaciones se dan en cuanto a cableado y terminaciones de línea. El estándar no especifica el conector o algún protocolo necesario

**Figura 1A.** Conexión Diferencial Multipunto utilizando múltiples Transceivers



Fuente: Nota de aplicación slla067a de Texas Instruments

- **Características Eléctricas.** El estándar TIA/EIA-485 describe el método de transmisión diferencial *half-duplex* o *full-duplex* diseñado para cables de par trenzado y otros medios balanceados. Éste requiere *drivers* para entregar un voltaje mínimo de salida de 1.5 V hasta con 32 unidades de carga de 12 k $\Omega$  cada una, más la resistencia en cada terminación del Bus. El nivel de voltaje de modo común en el bus puede variar entre -7 V y 12 V y los *receivers* deben ser bastante

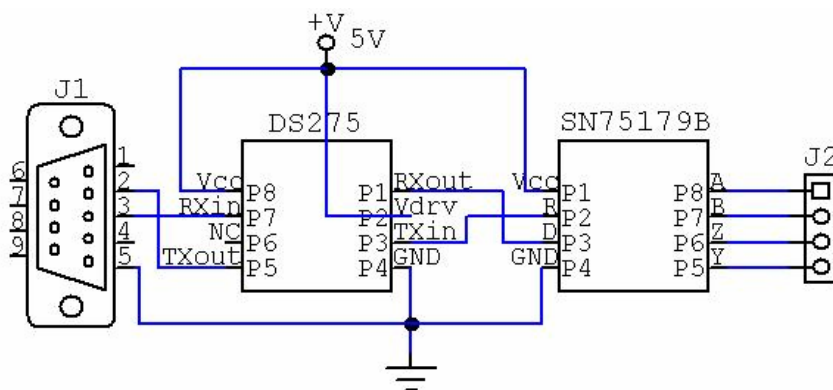
sensibles para determinar el estado del bus basado en un nivel diferencial de señal de 200 mV.

- **Protocolo.** Ningún protocolo en particular es especificado en el estándar, sin embargo muchos protocolos populares toman el estándar RS-485 como una solución. Éstos incluyen Profibus (EN 50170), Interbus-S y MODbus.
  
- **Aplicaciones.** Debido a su naturaleza de transmisión diferencial, 485 es robusto contra el ruido eléctrico. Igualmente, gracias a su amplio rango de voltaje de modo común, es tolerante a cambios en los potenciales de tierra entre los nodos. Por estas dos razones, este estándar es preferido para aplicaciones que requieran baja emisión de ruido y susceptibilidad. Este tiene un gran valor en largas líneas de transmisión.
  
- **Características.**
  - ❖ Permite transmisión de línea balanceada en modo multipunto.
  - ❖ Admite diferentes modos de transmisión (*half-duplex* o *full-duplex*), al igual que topologías de red (dos o cuatro hilos).
  - ❖ Hasta 32 *Transceivers* pueden compartir la red multipunto.
  - ❖ Interface muy robusta (rango de voltaje de modo común: -7 V a 12 V)
  - ❖ Alta tolerancia al ruido.
  - ❖ Ideal en aplicaciones de largas distancias (hasta 1200 m).
  - ❖ Económico, al utilizarse sólo un o dos pares de cables.

## ANEXO B ADAPTADOR DE INTERFAZ ESTÁNDAR RS-485 A RS-232

Este módulo adaptador cumple la función principal de convertir los datos provenientes del prototipo en formato RS-485 a formato RS-232 para que puedan ser recibidos y analizados en el PC por el puerto serial (COM1) (figura B1)

**Figura B1.** Diagrama esquemático del Adaptador.



El adaptador recibe los niveles de tensión en el estándar RS-485 y los pasa a lógica TTL/CMOS mediante el transceiver SN75179b, luego el transceiver DS275 toma éstos niveles de tensión y los pasa al estándar RS-232.

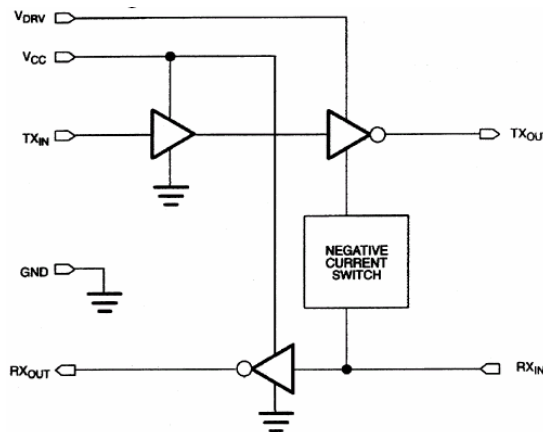
**B.1. Circuitos Integrados.** Para implementar éste módulo adaptador se utilizaron circuitos los integrados SN75179b y DS275, los cuales se especifican a continuación.

**B.1.1. Transceiver SN75179b.** Este circuito integrado también hace parte del prototipo por lo que su funcionamiento, características y diagrama (lógico y de pines) se encuentran especificados en la página \_\_\_ de este libro.

**B.1.2. Transceiver DS275.** Este *transceiver* es un circuito integrado CMOS de bajo costo y muy bajo consumo de potencia para puertos seriales con interfaz RS-232. Las entradas de *receiver* convierten los niveles de la señal de RS-232 a niveles de tensión en la lógica TTL/CMOS.

El transmisor emplea un único circuito el cual toma corriente de la señal RS-232 recibida cuando esta señal está en un estado negativo. Puesto que la mayoría de las comunicaciones seriales se mantienen en un estado estáticamente negativo, el uso de la señal recibida para la alimentación negativa reduce grandemente el consumo de potencia del DS275. Esta característica es especialmente importante para sistemas con alimentación por medio de baterías (Figura B2).

**Figura B2.** Diagrama de Bloques



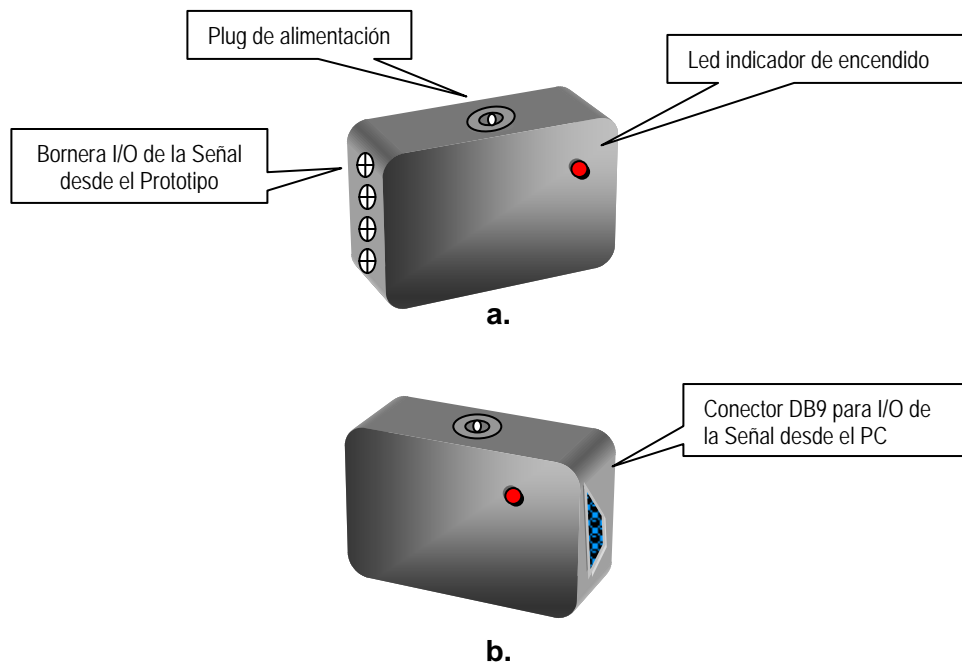
Fuente: Hoja de Datos del Fabricante.

- **Sección del Receiver.** El pin RX<sub>IN</sub> es la entrada del *Receiver* para una señal RS232 cuyos niveles pueden variar entre  $\pm 3$  a  $\pm 15$  volts. Las señales recibidas son invertidas y cambiadas a los niveles normales de +5 volts de la lógica TTL/CMOS. La salida asociada con RX<sub>IN</sub> es RX<sub>OUT</sub> la cual varía entre V<sub>CC</sub> y Gnd. Por lo tanto un nivel negativo (*mark*) en RX<sub>IN</sub> produce un 1 lógico en RX<sub>OUT</sub> y un nivel positivo (*space*) en RX<sub>IN</sub> produce un 0 lógico en RX<sub>OUT</sub>.

- **Sección del Transmisor.**  $TX_{IN}$  es la entrada compatible en la lógica TTL/CMOS para datos digitales desde el sistema del usuario. Un 1 lógico en  $TX_{IN}$  produce un nivel negativo (*mark*) en  $TX_{OUT}$  y un 0 lógico en  $TX_{IN}$  produce un nivel positivo (*space*) en  $TX_{OUT}$

**B.2. Conexiones.** El Adaptador contiene las siguientes conexiones (Figura B3 y B4).

**Figura B4.** Vistas: a. Frontal y Lateral izquierda del Adaptador, b: Frontal y Lateral derecha del Adaptador



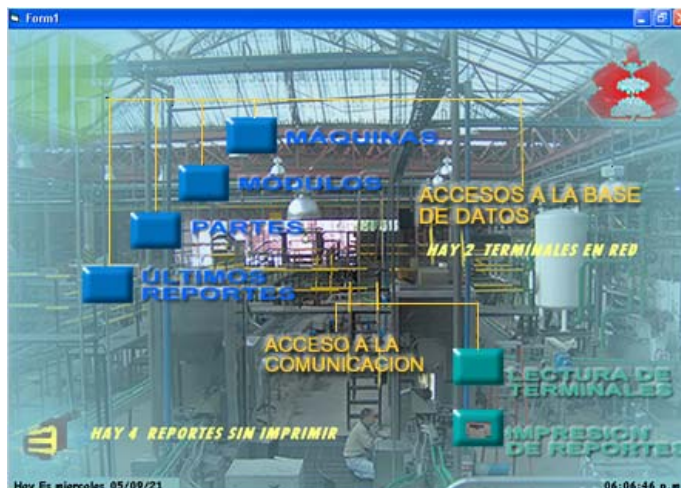
## ANEXO C. MANUAL DE USUARIO PARA LA INTERFAZ DE VISUAL BASIC 6.0

Este anexo se elaboró con el fin de presentar al usuario una descripción de los elementos que conforman la aplicación y los pasos que debe seguir para su correcta utilización

En el momento en que el usuario ponga a funcionar la aplicación en pantalla aparecerán 6 botones, de los cuales se encuentran 4 destinados para acceso a la Base de Datos en color azul y 2 en verde (uno encargado de establecer comunicación con el prototipo y el otro de imprimir los reportes recibidos).

También se proporciona al usuario el dato de fecha, hora y cantidad de reportes que están pendientes para impresión lo cual se ilustra en la figura C1.

**Figura C1.** Presentación de la Aplicación.



**Acceso a la Base de datos.** Al hacer clic sobre los botones de acceso a la base de datos se ingresa a los formularios que además contener las tablas Máquinas, Módulos, Partes y Últimos reportes, también poseen los elementos necesarios para el procesamiento de los registros que se encuentran almacenados en ellas.



En el momento de hacer clic sobre cualquiera de estos botones se accede a la base de datos permitiendo visualizar los listados de las máquinas, módulos o partes de las máquinas instaladas en el tren de envase N° 2 de Bavaria Cervecería de Boyacá. Adicionalmente cada ventana consta de una barra de herramientas, localizada en la parte inferior, con la cual se puede acceder a los siguientes procedimientos: Introducir y Guardar, Eliminar; Editar, Imprimir, desplazarse al formulario anterior, pasar al formulario siguiente y Salir de la aplicación ver figura C2.

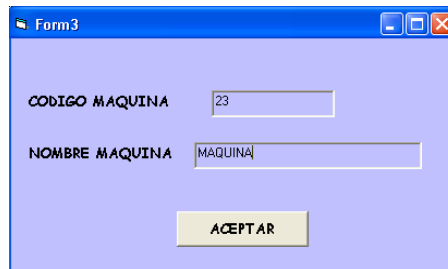
**Figura C2.** Formulario Maquinas.



- ❖ Al oprimir Agregar y Guardar, aparece un cuadro de dialogo (ver figura C3) en el que se debe insertar tanto el nombre, como el código de la máquina que se quiere ingresar al sistema. La operación se sella en el momento de dar la orden aceptar con lo cual el nuevo registro aparece en la base de datos.

Hay que aclarar que para la tabla máquinas se permite comenzar a incluir datos a partir del registro número 21 debido a que las 20 posiciones anteriores ya están ocupadas. Algo similar pasa con la tabla módulos para la cual se pide ingresar además del código de la máquina, el código del módulo y el nombre del módulo; por lo tanto el primer código aceptado es 20, 05. Mientras que para la tabla partes hay que introducir el código de la maquina, el código del modulo, el código de la parte y el nombre de la parte, por lo tanto el primer código aceptado será el 20, 04, 04. Si se intenta agregar un dato en una posición que ya esta ocupada el sistema generará un error.

**Figura C3.**Cuadro de Dialogo Insertar y Guardar.



The image shows a screenshot of a Windows-style dialog box titled "Form3". It has a light blue background and a blue border. At the top right, there are standard window control buttons (minimize, maximize, close). The dialog contains two text input fields. The first field is labeled "CODIGO MAQUINA" and contains the number "23". The second field is labeled "NOMBRE MAQUINA" and contains the word "MAQUINA". Below these fields is a yellow button with the text "ACEPTAR" in black capital letters.

- ❖ Al chiquear eliminar, aparece un cuadro de dialogo similar al anterior (ver figura C4), en el cual se debe introducir la información que se desea retirar del sistema. Para finalizar la operación es necesario dar la orden de eliminar oprimiendo el botón con el mismo nombre.

**Figura C4.**Cuadro de Dialogo Eliminar.

The screenshot shows a dialog box titled "Form4" with a light blue background. It contains two text input fields: "CODIGO MAQUINA" with the value "23" and "NOMBRE MAQUINA" with the value "MAQUINA". Below the fields is a yellow button labeled "ELIMINAR".

- ❖ Para editar un nombre existente en la base de datos; se debe ingresar tanto el código como el nombre, ya sea de la máquina, el módulo o la parte que se desea reemplazar. Luego al hacer clic en editar aparece otro cuadro de dialogo en el cual se pide escribir la información que se quiere almacenar en la base de datos, para terminar el proceso se debe dar aceptar (ver figura C5).

Cabe anotar que en esta parte se varia una posición que ya esta ocupada.

**Figura C5.**Cuadro de Dialogo Editar.

The screenshot shows a dialog box titled "Form5" with a light blue background. It contains two text input fields: "CODIGO MAQUINA" with the value "23" and "NOMBRE MAQUINA" with the value "MAQUINA". Below the fields is a yellow button labeled "EDITAR".

Overlaid on the bottom of "Form5" is a smaller dialog box titled "INTERFAZ" with a light yellow background. It contains a text input field with the text "introduzca el nuevo nombre" and two buttons: "Aceptar" and "Cancelar". Below the input field is a text box containing "MAQUINA Nº 23".



Sirve para acceder a la ventana Últimos Reportes con el fin de realizar consultas acerca de las fallas que se han presentado en determinada máquina. Ya que en esta se guarda el historial del funcionamiento de las máquinas instaladas en el tren de envase de Bavaria Cervecería de Boyacá

Al dar clic sobre el botón correspondiente a la tabla Últimos Reportes aparece un cuadro de dialogo (ver figura C6) en el que se debe seleccionar tanto el nombre de la máquina sobre la cual se desean buscar los fallos mas recientes, como el número de informes que se quieren observar, pues si solo se elige uno de los ítem el sistema inmediatamente genera un error y pide que se seleccionen las dos condiciones para ejecutar la consulta. Por último se tiene que dar la orden de búsqueda haciendo clic en el botón buscar de la barra de herramientas, para que la aplicación indague en la base de datos sobre la información requerida y muestre el resultado de la consulta en una tabla (ver figura C7).

**Figura C6.** Cuadro de Dialogo Buscar Últimos Reportes.



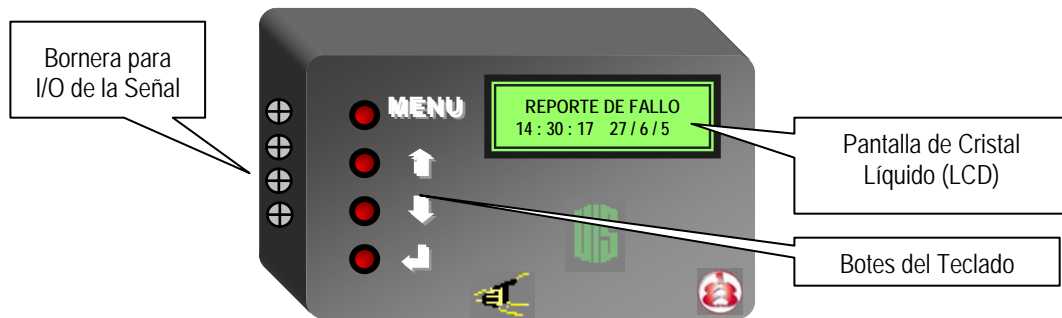


## ANEXO D. MANUAL DE USUARIO DEL EQUIPO

El presente anexo se hizo con el fin de familiarizar al usuario con los elementos que conforman el equipo, la interfaz (teclado y lcd) de usuario y los pasos que se deben seguir para generar y descargar los Reportes de Fallo.

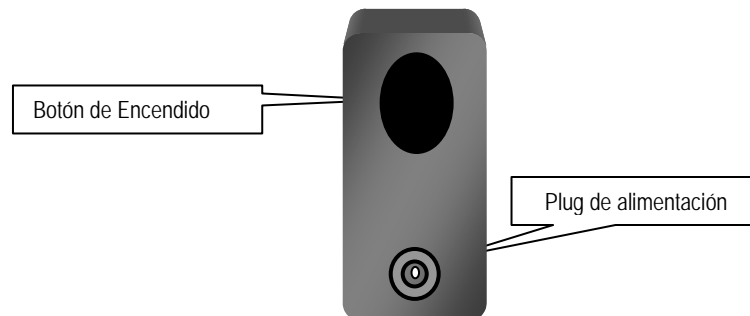
A continuación se describen en detalle las partes del prototipo y sus respectivas funciones (Figura D1).

**Figura D1.** Vista de Prototipo.



En la figura D2, vemos la parte lateral derecha del equipo y sus respectivos elementos.




**Figura D2.** Vista lateral derecha del prototipo



**D.1. CONEXIONES BÁSICAS.** El equipo utiliza los siguientes elementos externos para su funcionamiento:

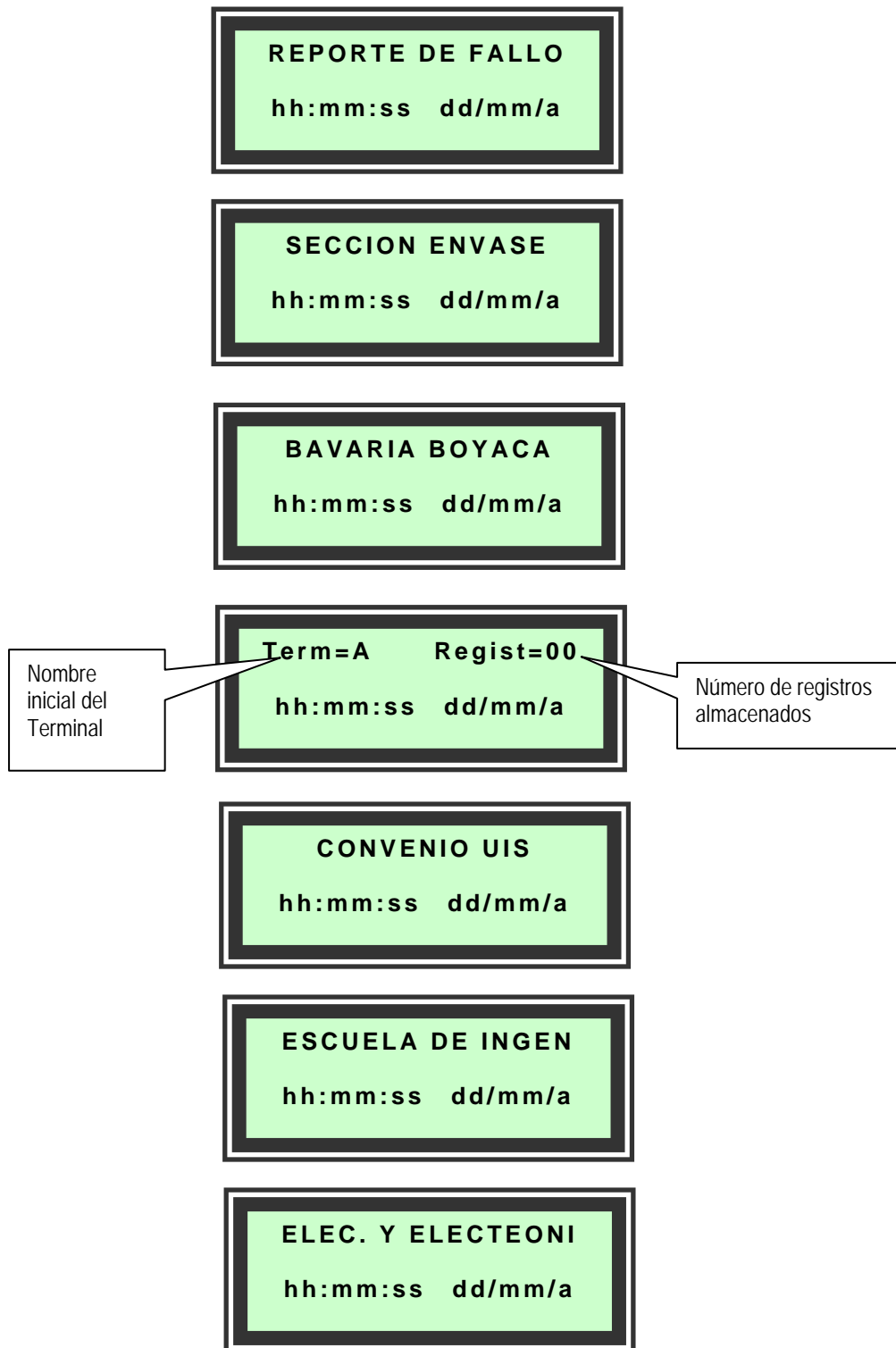
- ❖ Cable de conexión a la Bornera: utilizado para realizar la conexión entre el prototipo y el adaptador RS-485.
- ❖ Fuente de poder: El equipo se puede alimentar con un adaptador de 9V de salida.

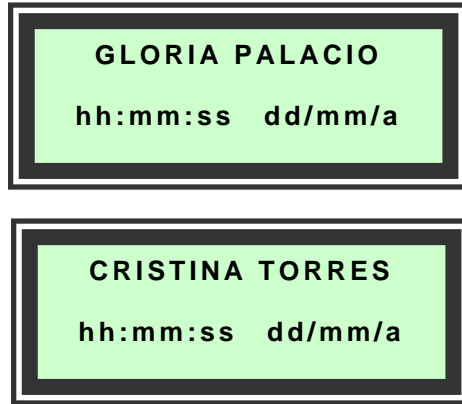
**D.2. TABLA DE EQUIVALENCIAS DEL TECLADO.**

NOMBRE	SÍMBOLO	DESCRIPCIÓN
Menú	<b>MENU</b>	<ul style="list-style-type: none"> <li>• Selección de menú u opciones de menú.</li> <li>• Avanza a la siguiente opción dentro del menú.</li> <li>• Cancela procesos</li> </ul>
Up		<ul style="list-style-type: none"> <li>• Avanza al siguiente número o letra cuando se edita un determinado carácter.</li> <li>• Avanza al siguiente menú.</li> <li>• Cancela procesos</li> </ul>
Down		<ul style="list-style-type: none"> <li>• Retrocede al anterior número o letra cuando se edita un determinado carácter.</li> <li>• Retrocede al anterior menú</li> <li>• Cancela procesos.</li> </ul>
OK		<ul style="list-style-type: none"> <li>• Aceptar o tecla Enter</li> <li>• Avanza al siguiente carácter cuando se editan los códigos del reporte.</li> </ul>

**D.3. FUNCIONAMIENTO DEL EQUIPO.** Al encender el equipo aparece en la pantalla el siguiente mensaje:

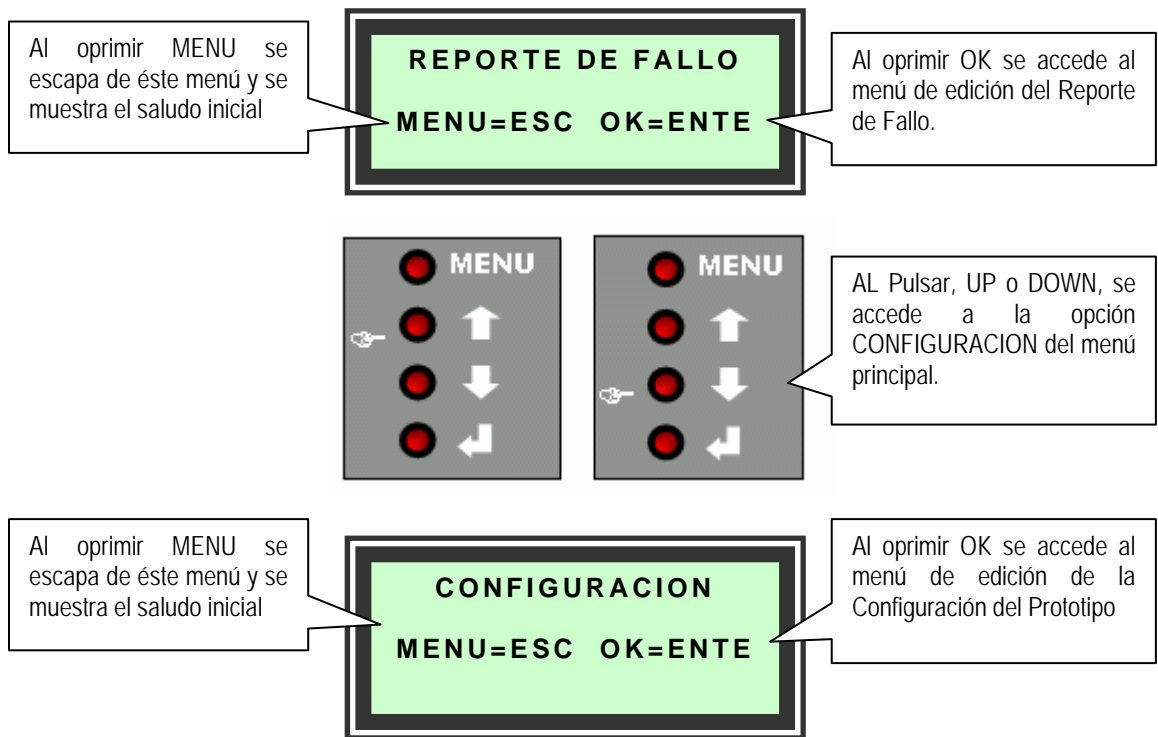
**Figura E3.** Pantalla de saludo Inicial





Al pulsar la tecla MENU se desplegará parte del menú principal, si luego se pulsan las teclas UP ó DOWN se podrá navegar en este menú. (Figura D4)

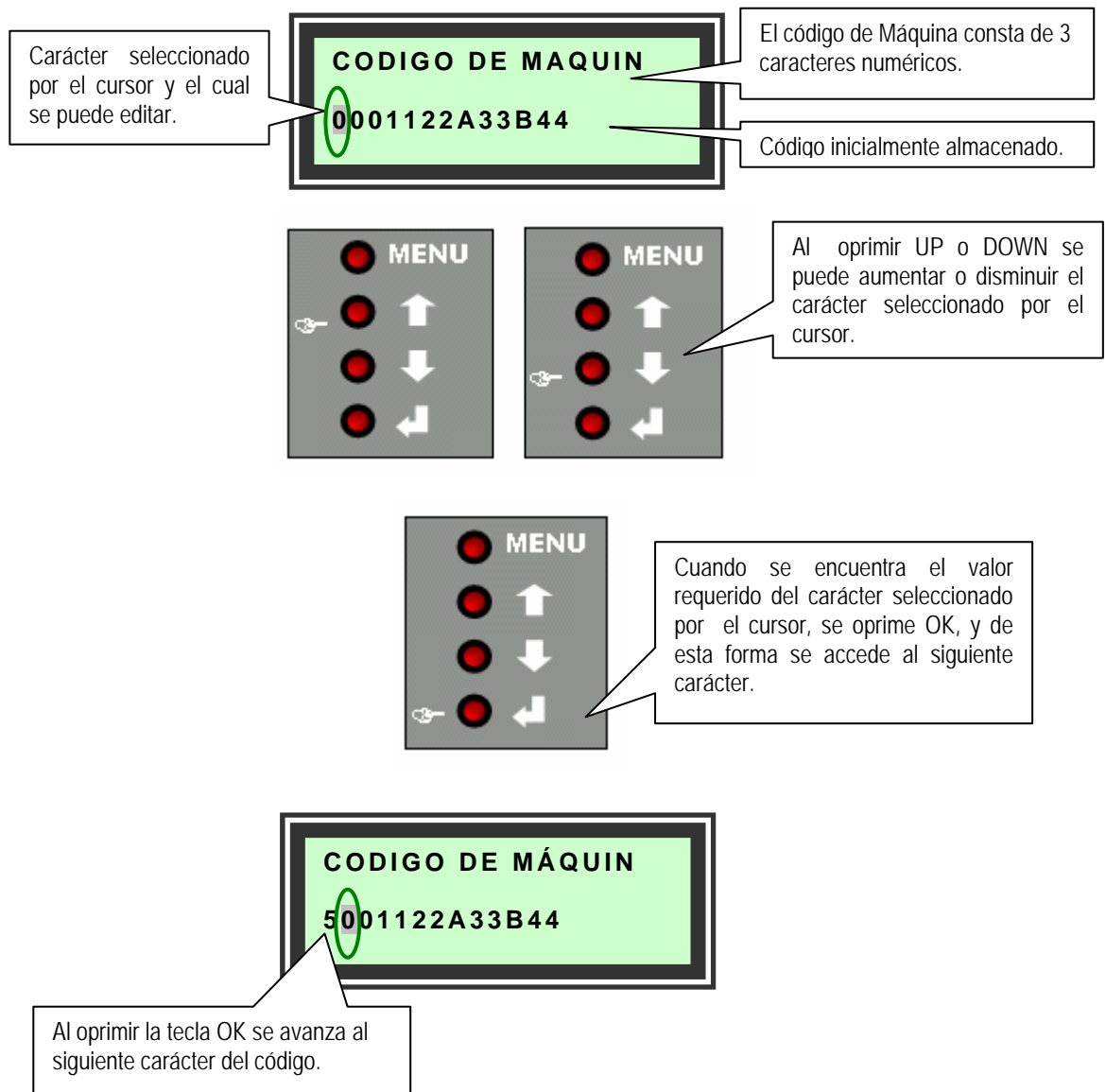
**Figura D4.** Menú principal



**D.3.1. Ingreso al Menú Reporte de Fallo.** Al ingresar al menú Reporte de Fallo aparecerá un nuevo menú en el cual el usuario puede observar la clase de códigos que conforman el Reporte de Fallo, el tamaño de cada uno y la clase de carácter.(figura D5)

Para editar cada código se utilizan las teclas UP y DOWN con las cuales se aumenta o se disminuye el valor numérico o alfabético de cada carácter

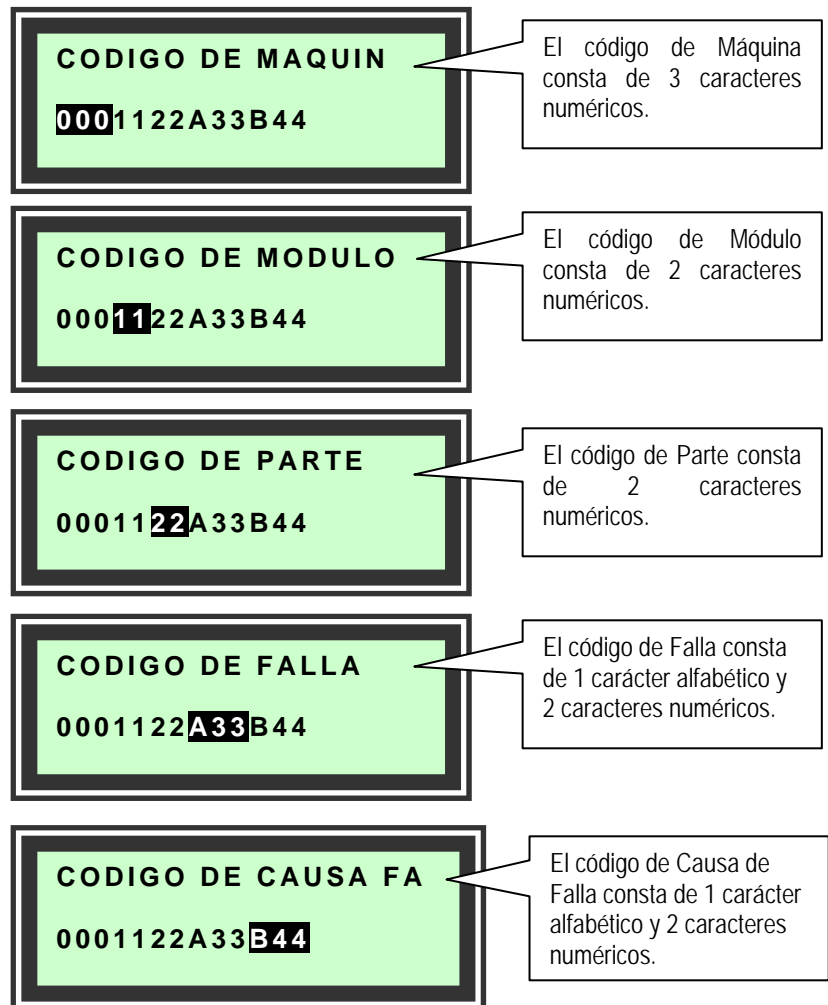
**Figura D5.** Edición de los Reportes de Fallo



El usuario puede recorrer el código del Reporte de Fallo con sólo oprimir la tecla OK, de esta forma se consigue seleccionar un carácter determinado y editarlo o corregirlo en caso de que se cometiera algún error en el momento de edición.

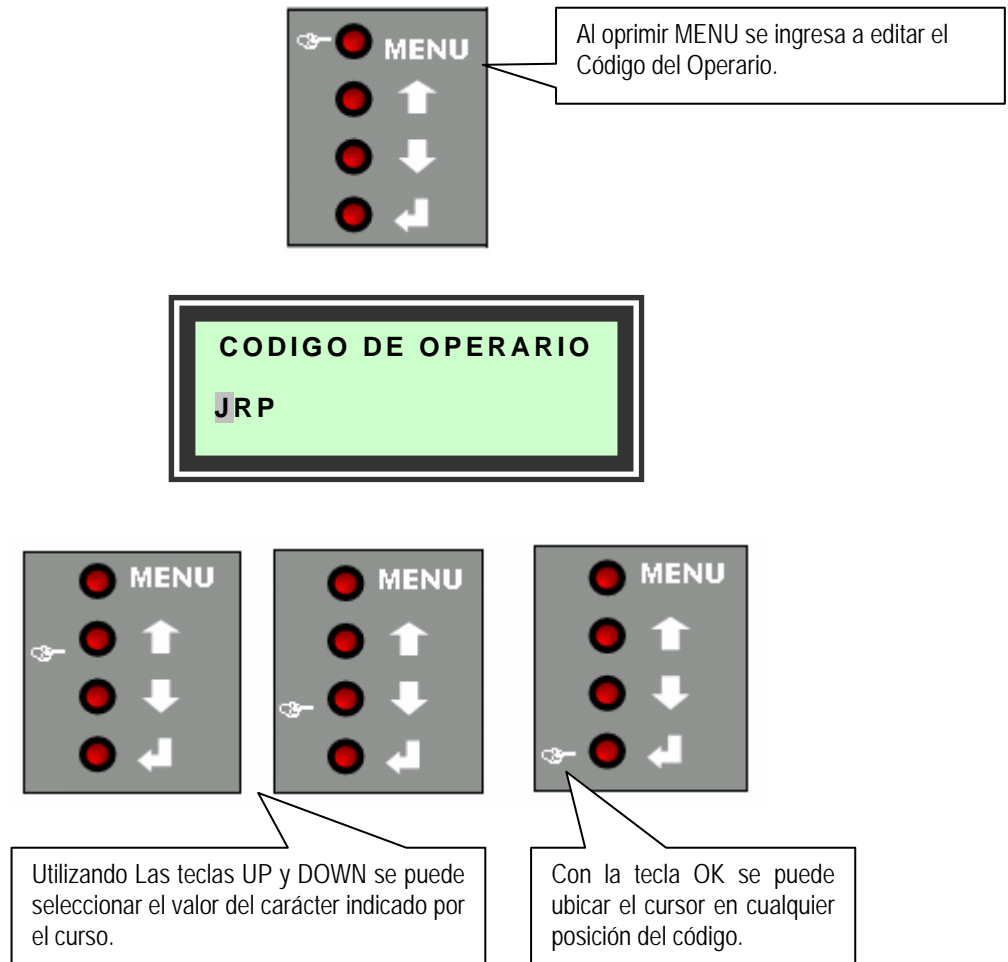
Al comienzo, y durante cada código, se le indica al usuario el nombre de la clase de código al cual se está accediendo, dentro del Reporte de Fallo, y que se encuentra disponible para ser editado (figura D6).

**Figura D6.** Recorrido dentro del código del Reporte de Fallo.

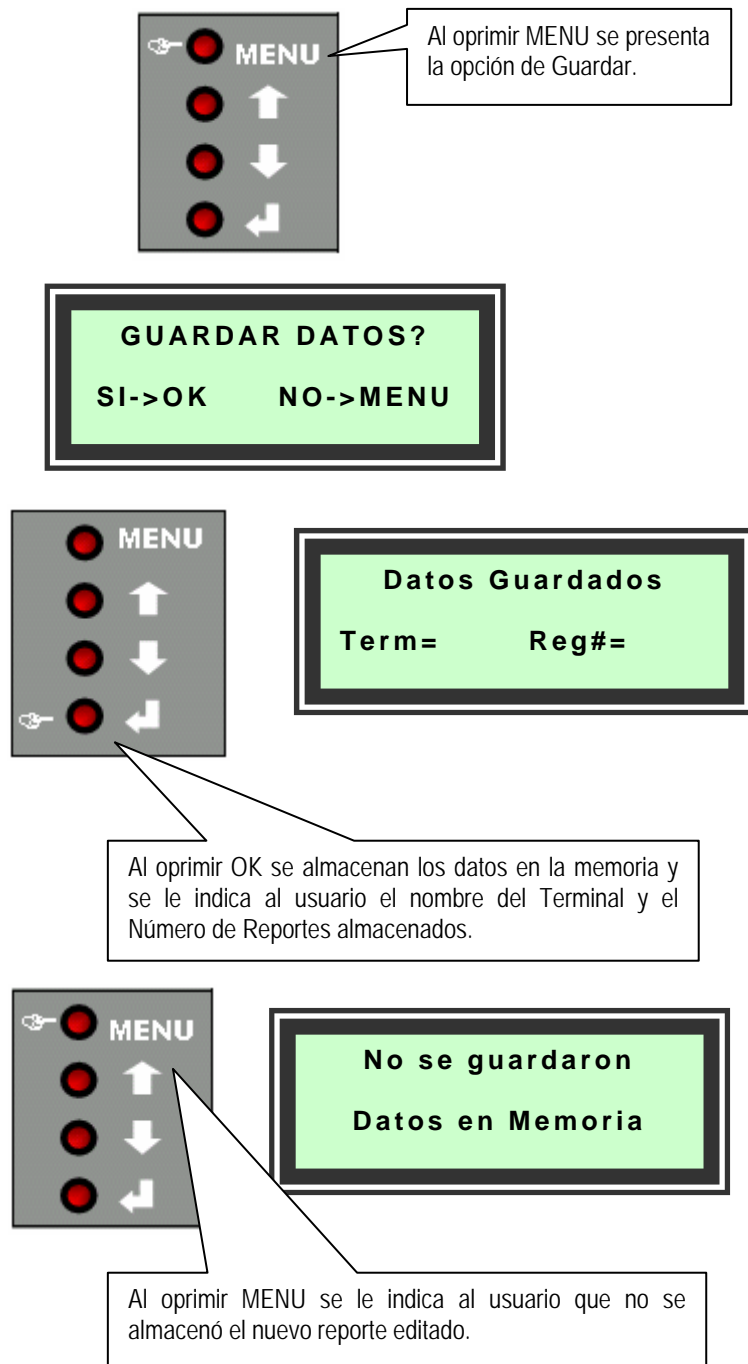


Cuando se ha finalizado la edición del reporte de fallo y los errores se han corregido, en caso de presentarse, se debe oprimir la tecla MENU, y editar el código del operario que está reportando la falla (figura D7). La edición de este código se realiza de la misma forma como se editaron los códigos anteriores. Al terminar la edición de éste último código se debe pulsar de nuevo la tecla MENU con la cual se accede a un nuevo pantallazo, allí se le pregunta al usuario si desea o no almacenar los nuevos datos en la memoria (figura D8).

**Figura D7.** Edición Código del Operario.



**Figura D8.** Almacenar los nuevos Reportes.

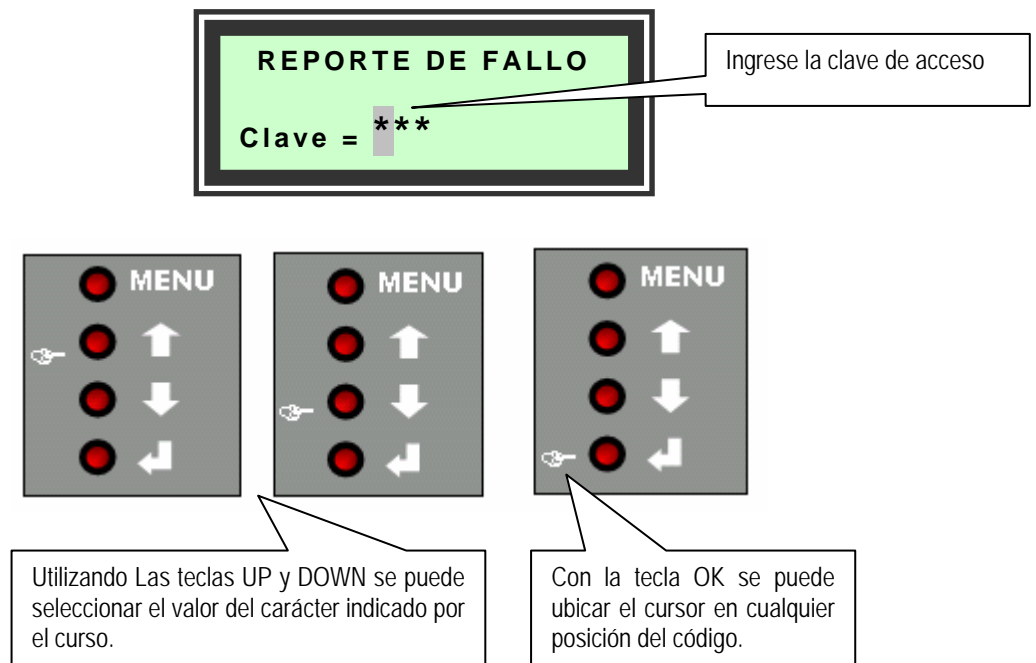


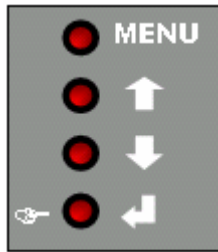
**D.3.2. Ingreso al Menú Configuración.** El acceso a la Configuración del prototipo se hace desde el menú principal, por medio de la cual se pueden modificar datos importantes como lo son:

- ❖ Nombre del terminal desde el que se está reportando el fallo.
- ❖ Fecha y hora que el prototipo genera, con el fin de almacenarla junto con cada reporte de fallo, para de ésta forma tener un mayor grado de confiabilidad de tal reporte.
- ❖ Clave de acceso la cual es indispensable para ingresar a éste menú (figura D9 y D11).

Esta interfaz le permite al usuario equivocarse hasta 2 veces durante el ingreso de la clave de acceso, de tal forma que cuando el usuario no digite la clave correctamente en dos oportunidades la interfaz reinicia de nuevo y se visualizará el saludo inicial

**Figura D9** Ingreso de la Clave de acceso





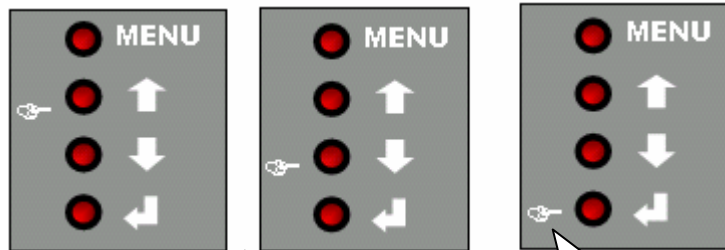
Al oprimir OK se detecta si la clave fue o no digitada correctamente

Si la clave de acceso fue correcta, el usuario puede ingresar al menú Configuración para realizar los respectivos cambios. (figura D10)

**Figura D10.** Menú Configuración

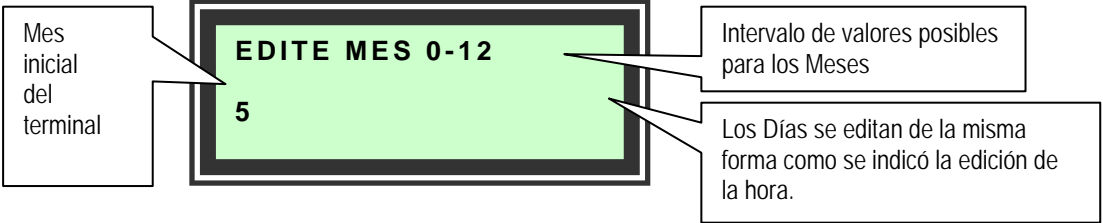
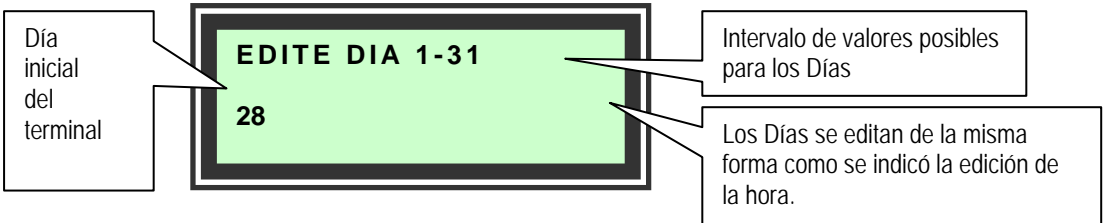
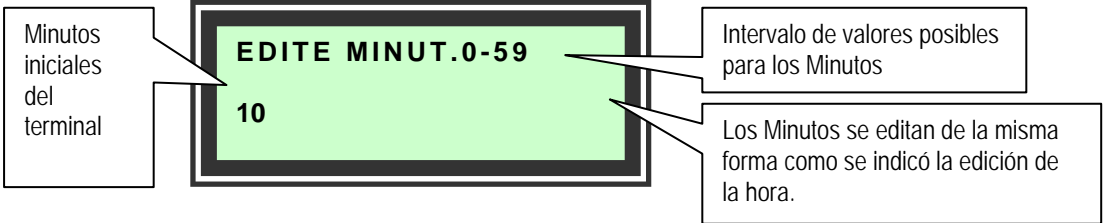
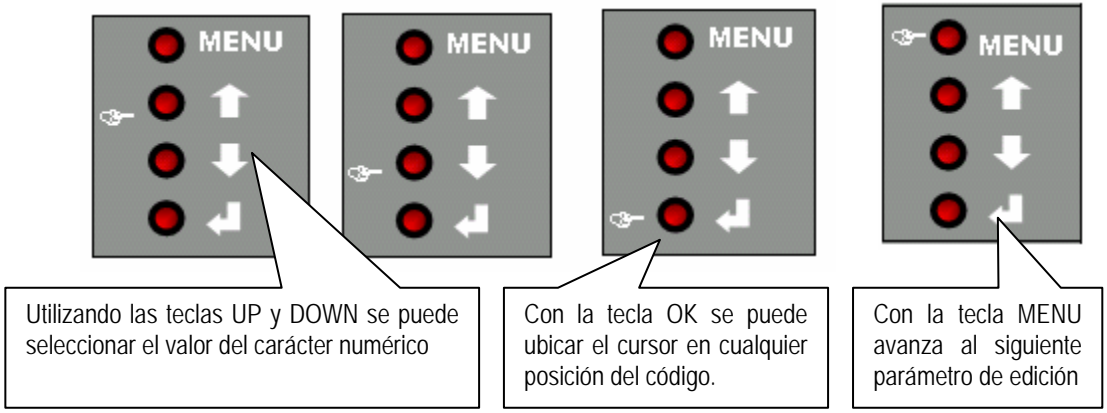
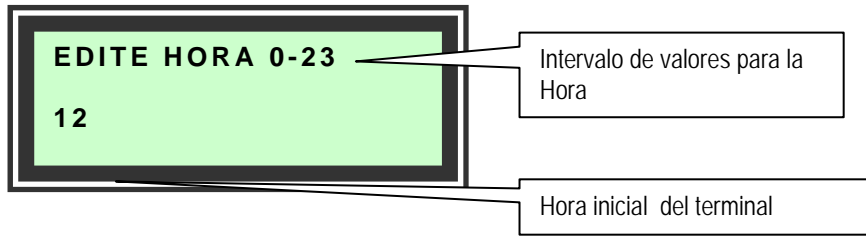


Nombre inicial del terminal



Utilizando las teclas UP y DOWN se puede seleccionar el valor del carácter alfabético

Con la tecla OK avanza al siguiente parámetro de edición



Año inicial del terminal

**EDITE AÑO 00-99**  
5

Intervalo de valores posibles para los Años

Los Años se editan de la misma forma como se indicó la edición de la hora.

Al oprimir MENU no se ingresa a éste submenú y se continúa con el menú de Configuración.

**CAMBIO DE CLAVE**  
**MENU=ESC OK=ENTE**

Al oprimir OK se accede al menú de edición de la Configuración del Prototipo

**BORRAR MEMORIA**  
**NO**

MENU

MENU

MENU

Al oprimir UP o DOWN se puede seleccionar Borrar o no la Memoria

Con la tecla MENU avanza al siguiente parámetro de edición

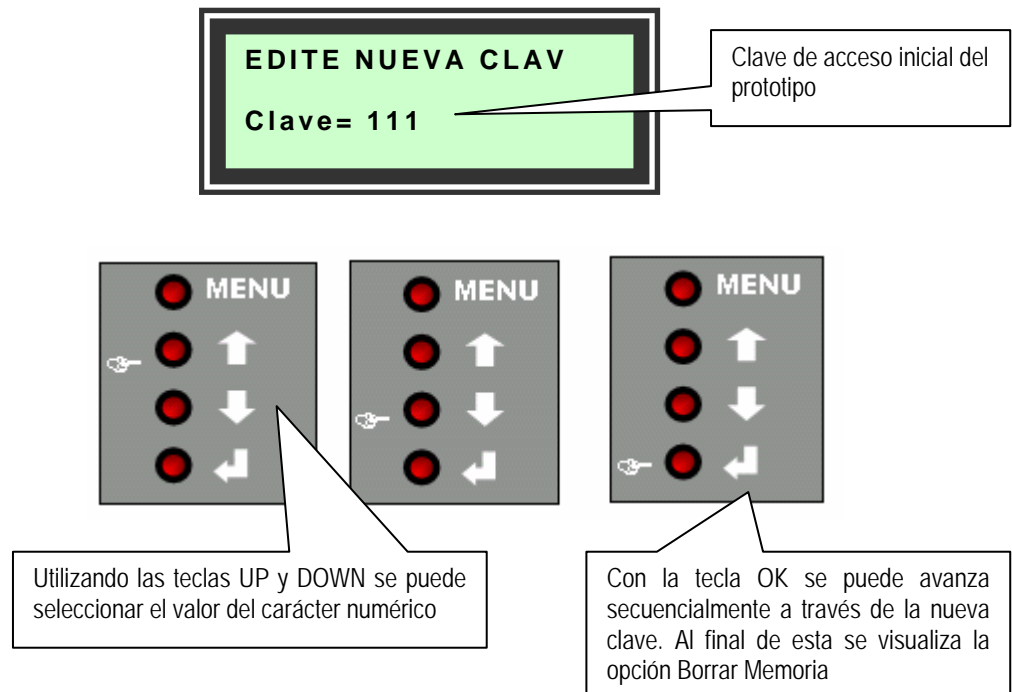
Al oprimir OK se almacena la configuración y para volver al Saludo Inicial se oprime MENU

**GUARDAR CONFIG?**  
**SI->OK NO->MENU**

Al oprimir MENU no se almacena la configuración y para volver al Saludo Inicial se oprime MENU

Al submenú Cambio Clave.se ingresa mediante el menú Configuración al seleccionarlo por medio de la tecla OK

**Figura D11.** Submenú Cambio de Clave.



Si el usuario deja de interactuar con la interfaz durante un minuto, se visualiza el siguiente mensaje:



Y después de otro minuto se visualiza el Saludo Inicial.

## ANEXO E. HOJAS DE DATOS

### E1. HOJA DE DATOS DEL MICROCONTROLADOR



## General Description

### 1.2 Introduction

The MC68HC908GP32 is a member of the low-cost, high-performance M68HC08 Family of 8-bit microcontroller units (MCUs). All MCUs in the family use the enhanced M68HC08 central processor unit (CPU08) and are available with a variety of modules, memory sizes and types, and package types.

### 1.3 Features

For convenience, features have been organized to reflect:

- Standard features of the MC68HC908GP32
- Features of the CPU08

#### 1.3.1 Standard Features of the MC68HC908GP32

- High-performance M68HC08 architecture optimized for C-compilers
- Fully upward-compatible object code with M6805, M146805, and M68HC05 Families
- 8-MHz internal bus frequency
- FLASH program memory security<sup>1</sup>
- On-chip programming firmware for use with host personal computer which does not require high voltage for entry
- In-system programming
- System protection features:
  - Optional computer operating property (COP) reset
  - Low-voltage detection with optional reset and selectable trip points for 3.0-V and 5.0-V operation
  - Illegal opcode detection with reset
  - Illegal address detection with reset

<sup>1</sup> No security feature is absolutely secure. However, Motorola's strategy is to make reading or copying the FLASH difficult for unauthorized users.

## General Description

- Oscillator stop mode enable bit (OSCSTOPENB) in the CONFIG register to allow user selection of having the oscillator enabled or disabled during stop mode
- 8-bit keyboard wakeup port
- 5-mA maximum current injection on all port pins to maintain input protection
- 40-pin plastic dual-in-line package (PDIP), 42-pin shrink dual-in-line package (SDIP), or 44-pin quad flat pack (QFP)
- Specific features of the MC68HC908GP32 in 40-pin PDIP are:
  - Port C is only 5 bits: PTC0–PTC4
  - Port D is only 6 bits: PTD0–PTD5; single 2-channel TIM module
- Specific features of the MC68HC908GP32 in 42-pin SDIP are:
  - Port C is only 5 bits: PTC0–PTC4
  - Port D is 8 bits: PTD0–PTD7; dual 2-channel TIM modules
- Specific features of the MC68HC908GP32 in 44-pin QFP are:
  - Port C is 7 bits: PTC0–PTC6
  - Port D is 8 bits: PTD0–PTD7; dual 2-channel TIM modules

### 1.3.2 Features of the CPU08

Features of the CPU08 include:

- Enhanced HC05 programming model
- Extensive loop control functions
- 16 addressing modes (eight more than the HC05)
- 16-bit index register and stack pointer
- Memory-to-memory data transfers
- Fast  $8 \times 8$  multiply instruction
- Fast 16/8 divide instruction
- Binary-coded decimal (BCD) instructions
- Optimization for controller applications
- Efficient C language support

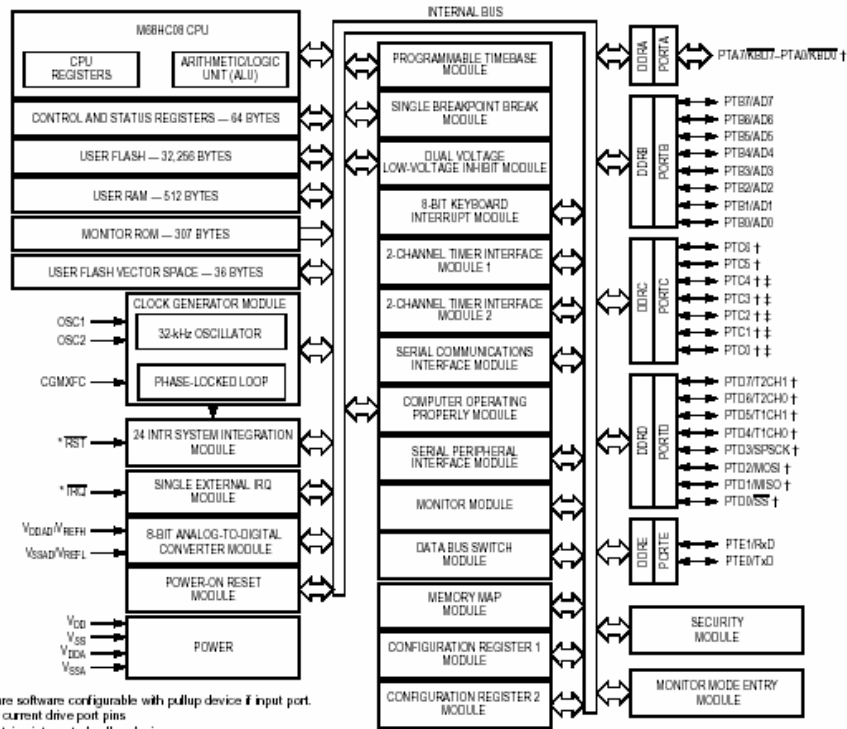
## General Description

- Oscillator stop mode enable bit (OSCSTOPENB) in the CONFIG register to allow user selection of having the oscillator enabled or disabled during stop mode
- 8-bit keyboard wakeup port
- 5-mA maximum current injection on all port pins to maintain input protection
- 40-pin plastic dual-in-line package (PDIP), 42-pin shrink dual-in-line package (SDIP), or 44-pin quad flat pack (QFP)
- Specific features of the MC68HC908GP32 in 40-pin PDIP are:
  - Port C is only 5 bits: PTC0–PTC4
  - Port D is only 6 bits: PTD0–PTD5; single 2-channel TIM module
- Specific features of the MC68HC908GP32 in 42-pin SDIP are:
  - Port C is only 5 bits: PTC0–PTC4
  - Port D is 8 bits: PTD0–PTD7; dual 2-channel TIM modules
- Specific features of the MC68HC908GP32 in 44-pin QFP are:
  - Port C is 7 bits: PTC0–PTC6
  - Port D is 8 bits: PTD0–PTD7; dual 2-channel TIM modules

### 1.3.2 Features of the CPU08

Features of the CPU08 include:

- Enhanced HC05 programming model
- Extensive loop control functions
- 16 addressing modes (eight more than the HC05)
- 16-bit index register and stack pointer
- Memory-to-memory data transfers
- Fast 8 × 8 multiply instruction
- Fast 16/8 divide instruction
- Binary-coded decimal (BCD) instructions
- Optimization for controller applications
- Efficient C language support



† Ports are software configurable with pullup device if input port.  
 ‡ Higher current drive port pins  
 \* Pin contains integrated pullup device

Figure 1-1. MCU Block Diagram

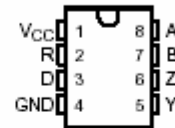
## E2. TRANSCEIVER SN75179B

### SN75179B DIFFERENTIAL DRIVER AND RECEIVER PAIR

SLLS003E – OCTOBER 1985 – REVISED JUNE 1995

- Meets or Exceeds the Requirements of TIA/EIA-422-B, TIA/EIA-485-A, and ITU Recommendation V.11
- Bus Voltage Range . . . -7 V to 12 V
- Positive- and Negative-Current Limiting
- Driver Output Capability . . . 60 mA Max
- Driver Thermal-Shutdown Protection
- Receiver Input Impedance . . . 12 k $\Omega$  Min
- Receiver Input Sensitivity . . .  $\pm 200$  mV
- Receiver Input Hysteresis . . . 50 mV Typ
- Operates From Single 5-V Supply
- Low Power Requirements

D OR P PACKAGE  
(TOP VIEW)



#### description

The SN75179B is a differential driver and receiver pair designed for balanced transmission-line applications and meets TIA/EIA-422-B, TIA/EIA-485-A, and ITU Recommendation V.11. It is designed to improve the performance of full-duplex data communications over long bus lines.

The SN75179B driver output provides limiting for both positive and negative currents. The receiver features high input impedance, input hysteresis for increased noise immunity, and input sensitivity of  $\pm 200$  mV over a common-mode input voltage range of -7 V to 12 V. The driver provides thermal shutdown for protection from line fault conditions. Thermal shutdown is designed to occur at a junction temperature of approximately 150°C. The SN75179B is designed to drive current loads of up to 60 mA maximum.

The SN75179B is characterized for operation from 0°C to 70°C.

#### Function Tables

##### DRIVER

INPUT D	OUTPUTS	
	Y	Z
H	H	L
L	L	H

##### RECEIVER

DIFFERENTIAL INPUTS A - B	OUTPUT R
$V_{ID} \geq 0.2$ V	H
$-0.2$ V $< V_{ID} < 0.2$ V	?
$V_{ID} \leq -0.2$ V	L
Open	?

H = high level, L = low level, ? = indeterminate



Please be aware that an important notice concerning availability, standard warranty, and use in critical applications of Texas Instruments semiconductor products and disclaimers thereto appears at the end of this data sheet.

PRODUCTION DATA information is current as of publication date. Products conform to specifications per the terms of Texas Instruments standard warranty. Production process may change without notice and does not necessarily indicate testing of all parameters.

**TEXAS  
INSTRUMENTS**

POST OFFICE BOX 655303 • DALLAS, TEXAS 75265

Copyright © 1995, Texas Instruments Incorporated

1

**absolute maximum ratings over operating free-air temperature range (unless otherwise noted)†**

Supply voltage, $V_{CC}$ (see Note 1)	7 V
Voltage range at any bus terminal	-10 V to 15 V
Differential input voltage, $V_{ID}$ (see Note 2)	$\pm 25$ V
Package thermal impedance, $\theta_{JA}$ (see Note 3): D package	197°C/W
P package	104°C/W
Storage temperature range, $T_{stg}$	-65°C to 150°C
Lead temperature 1,6 mm (1/16 inch) from case for 10 seconds	260°C

† Stresses beyond those listed under "absolute maximum ratings" may cause permanent damage to the device. These are stress ratings only, and functional operation of the device at these or any other conditions beyond those indicated under "recommended operating conditions" is not implied. Exposure to absolute-maximum-rated conditions for extended periods may affect device reliability.

- NOTES: 1. All voltage values, except differential input voltage, are with respect to network ground terminal.  
 2. Differential input voltage is measured at the noninverting input with respect to the corresponding inverting input.  
 3. The package thermal impedance is calculated in accordance with JEDEC 51, except for through-hole packages, which use a trace length of zero.

**recommended operating conditions**

		MIN	NOM	MAX	UNIT
Supply voltage, $V_{CC}$		4.75	5	5.25	V
High-level input voltage, $V_{IH}$	Driver	2			V
Low-level input voltage, $V_{IL}$	Driver			0.8	V
Common-mode input voltage, $V_{IC}$		-7‡		12	V
Differential input voltage, $V_{ID}$				$\pm 12$	V
High-level output current, $I_{OH}$	Driver			-60	mA
	Receiver			-400	$\mu$ A
Low-level output current, $I_{OL}$	Driver			60	mA
	Receiver			8	mA
Operating free-air temperature, $T_A$		0		70	°C

‡ The algebraic convention, where the less positive (more negative) limit is designated minimum, is used in this data sheet for common-mode input voltage and threshold voltage.



# SN75179B DIFFERENTIAL DRIVER AND RECEIVER PAIR

SLLS003E – OCTOBER 1985 – REVISED JUNE 1995

## DRIVER SECTION

electrical characteristics over recommended ranges of supply voltage and operating free-air temperature (unless otherwise noted)

PARAMETER	TEST CONDITIONS	MIN	TYP†	MAX	UNIT
$V_{IK}$ Input clamp voltage	$I_I = -18 \text{ mA}$			-1.5	V
$V_O$ Output voltage	$I_O = 0$	0		6	V
$ V_{OD1} $ Differential output voltage	$I_O = 0$	1.5		6	V
$ V_{OD2} $ Differential output voltage	$R_L = 100 \Omega$ See Figure 1	$1/2 V_{OD1}$ or $2^\ddagger$			V
	$R_L = 54 \Omega$ See Figure 1	1.5	2.5	5	V
$ V_{OD3} $ Differential output voltage	See Note 4	1.5		5	V
$\Delta V_{OD} $ Change in magnitude of common-mode output voltages§				$\pm 0.2$	V
$V_{OC}$ Common-mode output voltage	$R_L = 54 \Omega$ or $100 \Omega$ See Figure 1			$\begin{matrix} 3 \\ -1 \end{matrix}$	V
$\Delta V_{OC} $ Change in magnitude of common-mode output voltages§				$\pm 0.2$	V
$I_O$ Output current	$V_{CC} = 0, V_O = -7 \text{ V to } 12 \text{ V}$			$\pm 100$	$\mu\text{A}$
$I_{IH}$ High-level input current	$V_I = 2.4 \text{ V}$			20	$\mu\text{A}$
$I_{IL}$ Low-level input current	$V_I = 0.4 \text{ V}$			-200	$\mu\text{A}$
$I_{OS}$ Short-circuit output current	$V_O = -7 \text{ V}$			-250	mA
	$V_O = V_{CC}$ or $12 \text{ V}$			250	mA
$I_{CC}$ Supply current (total package)	No load		57	70	mA

† All typical values are at  $V_{CC} = 5 \text{ V}$  and  $T_A = 25^\circ\text{C}$ .

‡ The minimum  $V_{OD2}$  with  $100\text{-}\Omega$  load is either  $1/2 V_{OD2}$  or  $2 \text{ V}$ , whichever is greater.

§  $\Delta|V_{OD}|$  and  $\Delta|V_{OC}|$  are the changes in magnitude of  $V_{OD}$  and  $V_{OC}$ , respectively, that occur when the input changes from a high level to a low level.

NOTE 4: See TIA/EIA-485-A, Figure 3.5, Test Termination Measurement 2.

switching characteristics,  $V_{CC} = 5 \text{ V}$ ,  $T_A = 25^\circ\text{C}$

PARAMETER	TEST CONDITIONS	MIN	TYP	MAX	UNIT
$t_{d(OD)}$ Differential output delay time	$R_L = 54 \Omega$ , See Figure 3		15	22	ns
$t_{t(OD)}$ Differential output transition time			20	30	ns

### Symbol Equivalents

DATA-SHEET PARAMETER	TIA/EIA-422-B	TIA/EIA-485-A
$V_O$	$V_{Oa}, V_{Ob}$	$V_{Oa}, V_{Ob}$
$ V_{OD1} $	$V_O$	$V_O$
$ V_{OD2} $	$V_I (R_L = 100 \Omega)$	$V_I (R_L = 54 \Omega)$
$ V_{OD3} $		$V_I$ (Test Termination Measurement 2)
$\Delta V_{OD} $	$  V_I  -  V_I  $	$  V_I  -  V_I  $
$V_{OC}$	$ V_{Os} $	$ V_{Os} $
$\Delta V_{OC} $	$ V_{Os} - V_{Os} $	$ V_{Os} - V_{Os} $
$I_{OS}$	$ I_{Sa} ,  I_{Sb} $	
$I_O$	$ I_{Sa} ,  I_{Sb} $	$I_{Sa}, I_{Sb}$

### E3. TRANSCEIVER DS275



## DS275 Line-Powered RS-232 Transceiver Chip

[www.dalsemi.com](http://www.dalsemi.com)

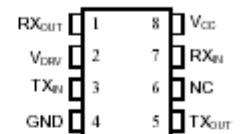
#### FEATURES

- Low-power serial transmitter/receiver for battery-backed systems
- Transmitter steals power from receive signal line to save power
- Ultra-low static current, even when connected to RS-232-E port
- Variable transmitter level from +5 to +12 volts
- Compatible with RS-232-E signals
- Available in 8-pin, 150 mil wide SOIC package (DS275S)
- Low-power CMOS

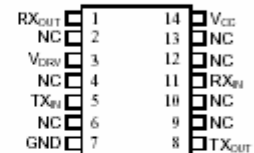
#### ORDERING INFORMATION

DS275	8-pin DIP
DS275S	8-pin SOIC
DS275E	14-pin TSSOP

#### PIN ASSIGNMENT



DS275 8-Pin DIP (300-mil)  
DS275 8-Pin SOIC (150-mil)



DS275E 14-Pin TSSOP

#### PIN DESCRIPTION

RX <sub>OUT</sub>	- RS-232 Receiver Output
V <sub>DRV</sub>	- Transmit driver +V
TX <sub>IN</sub>	- RS-232 Driver Input
GND	- System Ground (0V)
TX <sub>OUT</sub>	- RS-232 Driver Output
NC	- No Connection
RX <sub>IN</sub>	- RS-232 Receive Input
V <sub>CC</sub>	- System Logic Supply (+5V)

#### DESCRIPTION

The DS275 Line-Powered RS-232 Transceiver Chip is a CMOS device that provides a low-cost, very low-power interface to RS-232 serial ports. The receiver input translates RS-232 signal levels to common CMOS/TTL levels. The transmitter employs a unique circuit which steals current from the receive RS-232 signal when that signal is in a negative state (marking). Since most serial communication ports remain in a negative state statically, using the receive signal for negative power greatly reduces the DS275's static power consumption. This feature is especially important for battery-powered systems such as laptop computers, remote sensors, and portable medical instruments. During an actual communication session, the DS275's transmitter will use system power (5-12 volts) for positive transitions while still employing the receive signal for negative transitions.

RX <sub>IN</sub>	±15 volts
TX <sub>IN</sub>	-0.3 to V <sub>CC</sub> + 0.3 volts
TX <sub>OUT</sub>	±15 volts
RX <sub>OUT</sub>	-0.3 to V <sub>CC</sub> + 0.3 volts
Storage Temperature	-55°C to +125°C
Operating Temperature	0°C to 70°C

- \* This is a stress rating only and functional operation of the device at these or any other conditions above those indicated in the operation sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods of time may affect reliability.

#### RECOMMENDED DC OPERATING CONDITIONS

PARAMETER	SYMBOL	MIN	TYP	MAX	UNITS	NOTES
Logic Supply	V <sub>CC</sub>	4.5	5.0	5.5	V	1
Transmit Driver Supply	V <sub>DRV</sub>	4.5	5-12	13.0	V	1
Logic 1 Input	V <sub>IH</sub>	2.0		V <sub>CC</sub> +0.3	V	2
Logic 0 Input	V <sub>IL</sub>	-0.3		+0.8	V	
RS-232 Input Range (RX <sub>IN</sub> )	V <sub>RS</sub>	-15		+15	V	
Dynamic Supply Current						
TX <sub>IN</sub> – V <sub>CC</sub>	I <sub>DRV1</sub>		400	800	μA	3
	I <sub>CC1</sub>		40	100	μA	
TX <sub>IN</sub> – GND	I <sub>DRV1</sub>		3.8	5.0	μA	
	I <sub>CC1</sub>		40	100	μA	
Static Supply Current						
TX <sub>IN</sub> – V <sub>CC</sub>	I <sub>DRV2</sub>		1.5	10.0	μA	4
	I <sub>CC2</sub>		10.0	15.0	μA	
TX <sub>IN</sub> – GND	I <sub>DRV2</sub>		3.8	5.0	mA	
	I <sub>CC2</sub>		10.0	20.0	μA	
Driver Leakage Current (V <sub>CC</sub> =0V)	I <sub>DRV3</sub>		0.05	1.0	μA	5

**DC ELECTRICAL CHARACTERISTICS** (0°C to 70°C;  $V_{CC} = V_{DRV} = 5V \pm 10\%$ )

PARAMETER	SYMBOL	MIN	TYP	MAX	UNITS	NOTES
TX <sub>OUT</sub> Level High	V <sub>OIXH</sub>	3.5	4.0	5.0	V	6
TX <sub>OUT</sub> Level Low	V <sub>OIXL</sub>	-8.5	-9.0		V	7
TX <sub>OUT</sub> Short Circuit Current	I <sub>SC</sub>		+60	+85	mA	
TX <sub>OUT</sub> Output Slew Rate	t <sub>SR</sub>			30	V/μs	
Propagation Delay	t <sub>PD</sub>		5		μs	8
RX <sub>IN</sub> Input Threshold Low	V <sub>TL</sub>	0.8	1.2	1.6	V	
RX <sub>IN</sub> Input Threshold High	V <sub>TH</sub>	1.6	2.0	2.4	V	
RX <sub>IN</sub> Threshold Hysteresis	V <sub>HYS</sub>	0.5	0.8		V	9
RX <sub>OUT</sub> Output Current @ 2.4V	I <sub>OH</sub>	-1.0			mA	
RX <sub>OUT</sub> Output Current @ 0.4V	I <sub>OL</sub>			3.2	mA	

**NOTES:**

1. V<sub>DRV</sub> must be greater than or equal to V<sub>CC</sub>.
2. V<sub>CC</sub> = V<sub>DRV</sub> = 5V ± 10%.
3. See test circuit in Figure 4.
4. See test circuit in Figure 5.
5. See test circuit in Figure 6.
6. TX<sub>IN</sub> = V<sub>TL</sub> and TX<sub>OUT</sub> loaded by 3 kΩ to ground.
7. TX<sub>IN</sub> = V<sub>TH</sub>, RX<sub>IN</sub> = -10 volts and TX<sub>OUT</sub> loaded by 3 kΩ to ground.
8. TX<sub>IN</sub> to TX<sub>OUT</sub> - see Figure 7.
9. V<sub>HYS</sub> = V<sub>TH</sub> - V<sub>TL</sub>.

## E4. MEMORIA SERIAL EEPROM 24FC256

  
**MICROCHIP**     **24AA256/24LC256/24FC256**  


---

**256K I<sup>2</sup>C™ CMOS Serial EEPROM**

Device Selection Table

Part Number	VCC Range	Max. Clock Frequency	Temp. Ranges
24AA256	1.8-5.5V	400 kHz <sup>(1)</sup>	I
24LC256	2.5-5.5V	400 kHz	I, E
24FC256	1.8-5.5V	1 MHz <sup>(2)</sup>	I

Note 1: 100 kHz for VCC < 2.5V.

Note 2: 400 kHz for VCC < 2.5V.

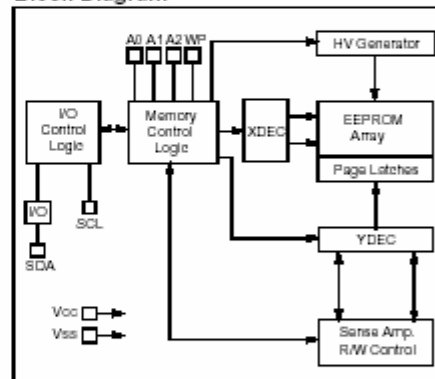
### Features:

- Low-power CMOS technology:
  - Maximum write current 3 mA at 5.5V
  - Maximum read current 400 µA at 5.5V
  - Standby current 100 nA, typical at 5.5V
- 2-wire serial interface bus, I<sup>2</sup>C™ compatible
- Cascadable for up to eight devices
- Self-timed erase/write cycle
- 64-byte Page Write mode available
- 5 ms max. write cycle time
- Hardware write-protect for entire array
- Output slope control to eliminate ground bounce
- Schmitt Trigger inputs for noise suppression
- 1,000,000 erase/write cycles
- Electrostatic discharge protection > 4000V
- Data retention > 200 years
- 8-pin PDIP, SOIC, SOIC, TSSOP, MSOP and DFN packages, 14-lead TSSOP package
- Pb-free finishes available
- Temperature ranges:
  - Industrial (I): -40°C to +85°C
  - Automotive (E): -40°C to +125°C

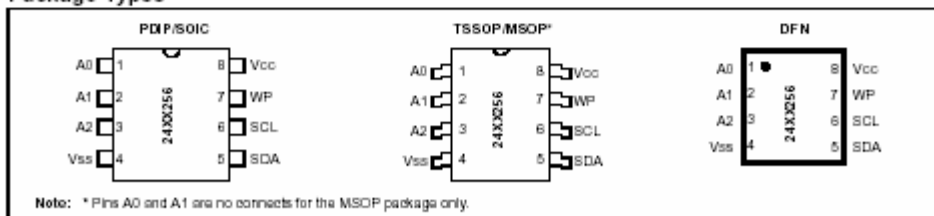
### Description:

The Microchip Technology Inc. 24AA256/24LC256/24FC256 (24XX256\*) is a 32K x 8 (256 Kbit) Serial Electrically Erasable PROM, capable of operation across a broad voltage range (1.8V to 5.5V). It has been developed for advanced, low-power applications such as personal communications or data acquisition. This device also has a page write capability of up to 64 bytes of data. This device is capable of both random and sequential reads up to the 256K boundary. Functional address lines allow up to eight devices on the same bus, for up to 2 Mbit address space. This device is available in the standard 8-pin plastic DIP, SOIC, TSSOP, MSOP and DFN packages.

### Block Diagram



### Package Types



\*24XX256 is used in this document as a generic part number for the 24AA256/24LC256/24FC256 devices.

# 24AA256/24LC256/24FC256

## 1.0 ELECTRICAL CHARACTERISTICS

### Absolute Maximum Ratings<sup>(†)</sup>

V <sub>CC</sub> .....	6.5V
All inputs and outputs w.r.t. V <sub>SS</sub> .....	-0.6V to V <sub>CC</sub> +1.0V
Storage temperature.....	-65°C to +150°C
Ambient temperature with power applied.....	-40°C to +125°C
ESD protection on all pins.....	≥ 4 kV

† NOTICE: Stresses above those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other conditions above those indicated in the operational listings of this specification is not implied. Exposure to Absolute Maximum Rating conditions for extended periods may affect device reliability.

TABLE 1-1: DC CHARACTERISTICS

DC CHARACTERISTICS			Electrical Characteristics: Industrial (I): V <sub>CC</sub> = +1.8V to 5.5V T <sub>A</sub> = -40°C to +85°C Automotive (E): V <sub>CC</sub> = +2.5V to 5.5V T <sub>A</sub> = -40°C to +125°C			
Param. No.	Sym.	Characteristic	Min.	Max.	Units	Conditions
D1	—	A0, A1, A2, SCL, SDA and WP pins:	—	—	—	—
D2	V <sub>H</sub>	High-level input voltage	0.7 V <sub>CC</sub>	—	V	—
D3	V <sub>L</sub>	Low-level input voltage	—	0.3 V <sub>CC</sub> 0.2 V <sub>CC</sub>	V V	V <sub>CC</sub> ≥ 2.5V V <sub>CC</sub> < 2.5V
D4	V <sub>HYS</sub>	Hysteresis of Schmitt Trigger inputs (SDA, SCL pins)	0.05 V <sub>CC</sub>	—	V	V <sub>CC</sub> ≥ 2.5V (Note)
D5	V <sub>OL</sub>	Low-level output voltage	—	0.40	V	I <sub>OL</sub> = 3.0 ma @ V <sub>CC</sub> = 4.5V I <sub>OL</sub> = 2.1 ma @ V <sub>CC</sub> = 2.5V
D6	I <sub>LI</sub>	Input leakage current	—	±1	µA	V <sub>IN</sub> = V <sub>SS</sub> or V <sub>CC</sub> , WP = V <sub>SS</sub> V <sub>IN</sub> = V <sub>SS</sub> or V <sub>CC</sub> , WP = V <sub>CC</sub>
D7	I <sub>LO</sub>	Output leakage current	—	±1	µA	V <sub>OUT</sub> = V <sub>SS</sub> or V <sub>CC</sub>
D8	C <sub>IN</sub> , C <sub>OUT</sub>	Pin capacitance (all inputs/outputs)	—	10	pF	V <sub>CC</sub> = 5.0V (Note) T <sub>A</sub> = 25°C, F <sub>CLK</sub> = 1 MHz
D9	I <sub>CC</sub> Read	Operating current	—	400	µA	V <sub>CC</sub> = 5.5V, SCL = 400 kHz
	I <sub>CC</sub> Write		—	3	mA	V <sub>CC</sub> = 5.5V
D10	I <sub>CCS</sub>	Standby current	—	1	µA	T <sub>A</sub> = -40°C to +85°C SCL = SDA = V <sub>CC</sub> = 5.5V A0, A1, A2, WP = V <sub>SS</sub>
			—	5	µA	T <sub>A</sub> = -40°C to +125°C SCL = SDA = V <sub>CC</sub> = 5.5V A0, A1, A2, WP = V <sub>SS</sub>

Note: This parameter is periodically sampled and not 100% tested.

# 24AA256/24LC256/24FC256

## 1.0 ELECTRICAL CHARACTERISTICS

### Absolute Maximum Ratings<sup>(†)</sup>

V <sub>CC</sub> .....	6.5V
All inputs and outputs w.r.t. V <sub>SS</sub> .....	-0.6V to V <sub>CC</sub> +1.0V
Storage temperature.....	-65°C to +150°C
Ambient temperature with power applied.....	-40°C to +125°C
ESD protection on all pins.....	≥ 4 kV

† NOTICE: Stresses above those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other conditions above those indicated in the operational listings of this specification is not implied. Exposure to Absolute Maximum Rating conditions for extended periods may affect device reliability.

TABLE 1-1: DC CHARACTERISTICS

DC CHARACTERISTICS			Electrical Characteristics: Industrial (I): V <sub>CC</sub> = +1.8V to 5.5V T <sub>A</sub> = -40°C to +85°C Automotive (E): V <sub>CC</sub> = +2.5V to 5.5V T <sub>A</sub> = -40°C to +125°C			
Param. No.	Sym.	Characteristic	Min.	Max.	Units	Conditions
D1	—	A0, A1, A2, SCL, SDA and WP pins:	—	—	—	—
D2	V <sub>H</sub>	High-level input voltage	0.7 V <sub>CC</sub>	—	V	—
D3	V <sub>L</sub>	Low-level input voltage	—	0.3 V <sub>CC</sub> 0.2 V <sub>CC</sub>	V V	V <sub>CC</sub> ≥ 2.5V V <sub>CC</sub> < 2.5V
D4	V <sub>HYS</sub>	Hysteresis of Schmitt Trigger inputs (SDA, SCL pins)	0.05 V <sub>CC</sub>	—	V	V <sub>CC</sub> ≥ 2.5V (Note)
D5	V <sub>OL</sub>	Low-level output voltage	—	0.40	V	I <sub>OL</sub> = 3.0 ma @ V <sub>CC</sub> = 4.5V I <sub>OL</sub> = 2.1 ma @ V <sub>CC</sub> = 2.5V
D6	I <sub>LI</sub>	Input leakage current	—	±1	µA	V <sub>IN</sub> = V <sub>SS</sub> or V <sub>CC</sub> , WP = V <sub>SS</sub> V <sub>IN</sub> = V <sub>SS</sub> or V <sub>CC</sub> , WP = V <sub>CC</sub>
D7	I <sub>LO</sub>	Output leakage current	—	±1	µA	V <sub>OUT</sub> = V <sub>SS</sub> or V <sub>CC</sub>
D8	C <sub>IN</sub> , C <sub>OUT</sub>	Pin capacitance (all inputs/outputs)	—	10	pF	V <sub>CC</sub> = 5.0V (Note) T <sub>A</sub> = 25°C, F <sub>CLK</sub> = 1 MHz
D9	I <sub>CC</sub> Read	Operating current	—	400	µA	V <sub>CC</sub> = 5.5V, SCL = 400 kHz
	I <sub>CC</sub> Write		—	3	mA	V <sub>CC</sub> = 5.5V
D10	I <sub>CCS</sub>	Standby current	—	1	µA	T <sub>A</sub> = -40°C to +85°C SCL = SDA = V <sub>CC</sub> = 5.5V A0, A1, A2, WP = V <sub>SS</sub>
			—	5	µA	T <sub>A</sub> = -40°C to +125°C SCL = SDA = V <sub>CC</sub> = 5.5V A0, A1, A2, WP = V <sub>SS</sub>

Note: This parameter is periodically sampled and not 100% tested.

## 24AA256/24LC256/24FC256

AC CHARACTERISTICS (Continued)			Electrical Characteristics:			
			Industrial (I):		VCC = +1.8V to 5.5V	TA = -40°C to +85°C
			Automotive (E):		VCC = +2.5V to 5.5V	TA = -40°C to +125°C
Param. No.	Sym.	Characteristic	Min.	Max.	Units	Conditions
13	TAA	Output valid from clock (Note 2)	—	3500	ns	1.8V ≤ VCC < 2.5V 2.5V ≤ VCC ≤ 5.5V 1.8V ≤ VCC < 2.5V 24FC256 2.5V ≤ VCC ≤ 5.5V 24FC256
14	TBUF	Bus free time: Time the bus must be free before a new transmission can start	4700 1300 1300 500	—	ns	1.8V ≤ VCC < 2.5V 2.5V ≤ VCC ≤ 5.5V 1.8V ≤ VCC < 2.5V 24FC256 2.5V ≤ VCC ≤ 5.5V 24FC256
15	Tof	Output fall time from VIH minimum to VIL maximum CB ≤ 100 pF	10 + 0.1CB	250 250	ns	All except, 24FC256 (Note 1)
16	TSP	Input filter spike suppression (SDA and SCL pins)	—	50	ns	All except, 24FC256 (Notes 1 and 3)
17	TWC	Write cycle time (byte or page)	—	5	ms	—
18	—	Endurance	1,000,000	—	cycles	25°C (Note 4)

- Note 1: Not 100% tested. CB = total capacitance of one bus line in pF.
- 2: As a transmitter, the device must provide an internal minimum delay time to bridge the undefined region (minimum 300 ns) of the falling edge of SCL to avoid unintended generation of Start or Stop conditions.
- 3: The combined TSP and VHYS specifications are due to new Schmitt Trigger inputs, which provide improved noise spike suppression. This eliminates the need for a TI specification for standard operation.
- 4: This parameter is not tested but ensured by characterization. For endurance estimates in a specific application, please consult the Total Endurance™ Model, which can be obtained from Microchip's web site at [www.microchip.com](http://www.microchip.com).

FIGURE 1-1: BUS TIMING DATA

