

Instituto Tecnológico de Costa Rica
Escuela de Ingeniería en Electrónica



Diseño y prueba del prototipo mediador universal

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

Rafael Ángel Rojas Rodríguez

Cartago, Octubre de 2005

INSTITUTO TECNOLÓGICO DE COSTA RICA
ESCUELA DE INGENIERÍA ELECTRÓNICA

PROYECTO DE GRADUACIÓN

TRIBUNAL EVALUADOR


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

Miembros del Tribunal



Ing. Gabriela Ortiz León

Profesor lector



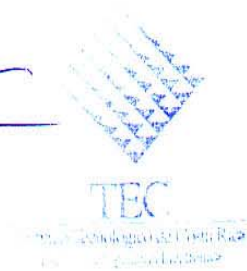
Ing. Juan Carlos Jiménez Robles

Profesor lector



Ing. Julio Stradi Granados

Profesor asesor



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

Cartago, 3 de Octubre del 2005

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

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

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

Cartago, Septiembre 2005

A handwritten signature in black ink, consisting of the letters 'R', 'A', 'R', and 'R' in a stylized, cursive font.

Rafael Ángel Rojas Rodríguez

Céd: 2-564-001

Resumen

El proyecto realizado pretende mejorar el sistema de comunicación entre los programas de control y los equipos de telecomunicaciones (SDH) de las distintas marcas. A través de rutinas de reconocimiento se multiplexan las señales, para conectar los equipos con la computadora, independiente de la configuración de pines del medio de conexión.

Con la implementación de este proyecto se busca aumentar los tiempos de acceso, para lograr una mayor eficiencia. Él cual sea capaz de identificar los equipos a través de protocolos de reconocimiento (*hand-shaking*).

El proyecto nace como iniciativa del Instituto Costarricense de Electricidad (ICE) para cubrir las necesidades de una rápida atención al mantenimiento correctivo y preventivo de los equipos.

La implementación final se trata de un circuito analógico con control digital, en el que se puede establecer (a nivel de *hardware*) la configuración de pines, y el modo de configuración Manual o Automático (a nivel de *software*); además el uso de bases de datos con *MySQL*, le dan al proyecto la posibilidad de una continuación (Final Abierto).

Palabras claves:

Protocolo;Ericsson;Lucent;Alcatel;NEC;RS-232

Summary

The carried out project seeks to improve the communication system between the control softwar's and the telecommunications equipment (SDH) of the different brands.

Through recognition routines they multiplex the signals, in order to connect the equipments to the computer, independent of the configuration, and connection.

The Development of this project pretends to increase the access times of the equipments, to achieve a bigger efficiency, at the same time be able to identify it, through recognition protocols (hand-shaking).

The project is born as an initiative of the "Instituto Costarricense de Electricidad (ICE)" to cover the needs from a quick attention to the corrective and preventive maintenance of the equipments.

The final implementation it is based in an analogical circuit with digital control, which can be establish (at hardware level) with the configuration, the configuration mode can be setup to Manual or Automatic (at software level); on the other hand the databases use is based in MySQL, that gives opportunity to continue the project (Open End).

Key words:

Protocol;Ericsson;Lucent;Alcatel;NEC;RS-232

Dedicatoria

Este trabajo lo dedico primeramente a Dios, ya que sin Él no somos nada. Sabias palabras me decía siempre mi Abuelita Melitina que en paz descansa: “Quien contra mi, si Dios está conmigo”. Dios es quien nos concede el privilegio de la vida y nos ofrece lo necesario para lograr nuestras metas. Señor Jesús, GRACIAS. Gracias te digo de todo corazón, por permitirme llegar hasta donde estoy, por las pruebas que me hacen crecer como persona y ser humano, que me acercan más a ti.

También le dedico este trabajo a mis increíbles y sabios padres, porque ellos siempre están conmigo en las buenas y en las malas; me educan, me aconsejan, me imparten valores para conducirme correctamente y me ofrecen el sabio consejo en el momento oportuno.

Además dedico este trabajo a mi novia Mariela, cuya comprensión y amor me han hecho una mejor persona, Gracias princesita.

A todas mis hermanas, que son las mujeres más especiales que Dios ha dispuesto en la Tierra, gracias por el apoyo y el cariño que me han brindado:

Ale, Nana, Naty, Nita y Pao, las amo.

A toda mi familia, a mis 7 sobrinitos.

A mis compañeros Edwin, Ruddy, Leo y amigos Willy´s y a todas aquellas personas que me apoyan, que siempre están conmigo en las buenas y en las malas.

Agradecimiento

Si nos encontramos a alguien que nos debe agradecimiento, enseguida lo recordarnos. ¡Cuántas veces nos encontramos con alguien al que debemos agradecimiento y no pensamos en ello!

Ing. Gabriela Ortiz

Ing. Juan Carlos Jiménez

Ing. Julio Stradi

Ing. Pedro Murillo

Ing. Carlos Badilla

Y a todos aquellos profesores que ayudaron en mi formación académica.

Gracias a toda mi familia y a todos mis amigos.

Gracias Mamá y Gracias Papá.

INDICE GENERAL

Capítulo 1: Introducción.....	1
1.1 Problema existente e importancia de su solución:.....	1
1.2 Solución seleccionada:	3
Capítulo 2: Meta y Objetivos	7
2.1 Meta.....	7
2.2 Objetivo general	7
2.3 Objetivos específicos.....	7
2.3.a Objetivos de hardware	7
2.3.b Objetivos de software	8
2.3.c Objetivos de documentación	8
2.3.d Objetivos de implementación	9
Capítulo 3: Marco Teórico	9
3.1 Descripción del sistema.....	10
3.2 Antecedentes Bibliográficos.....	11
3.2.1 Comunicación asíncrona.....	11
3.2.1.a El protocolo RS-232.....	11
3.2.1.b Generalidades:	12
3.2.1.c Funcionamiento:	12
3.2.1.d Limitaciones del RS-232	13
3.2.2 La comunicación USB:.....	13
3.2.2.a Generalidades	13
3.2.2.b Funcionamiento	14
3.2.3 UTP (Par trenzado no blindado):.....	14
3.2.3.a Tipos de UTP.....	14
3.2.3.b Velocidades utilizando el cable UTP.....	14
3.2.3.c Protocolo para el UTP	14
3.2.3.d Conector del cable UTP.....	15
3.2.4 Modelos de datos	15
3.2.4.a Metodología de diseño de bases de datos.....	16
3.2.4.b Modelo Entidad/Relación	18
3.3 Descripción de los principales principios de software y electrónicos relacionados con la solución del problema.....	19
3.3.a Principios de software	19
3.3.b Principios electrónicos.....	19
3.3.c Principios matemáticos.....	19
Capítulo 4: Procedimiento Metodológico.....	20
4.1 Reconocimiento y definición del problema.....	20
4.2 Obtención y análisis de información	21
4.3 Evaluación de las alternativas y síntesis de una solución.....	22
4.4 Implementación de la solución	23
4.5 Reevaluación y rediseño.....	24

Capítulo 5: Descripción detallada de la solución	25
5.1 Análisis de soluciones y selección final:	25
5.1.a Cálculo del periodo de lectura y transmisión	27
5.1.b Cálculo del tiempo transcurrido en la ejecución de la rutina	27
5.2 Descripción del Hardware	30
5.2.a Fabricación de un cable directo con terminales de monitoreo	30
5.2.b Hardware espiador	31
5.2.c Multiplexión de la señal	33
5.2.d Hardware de control implementado	35
5.3 Descripción del Software	37
5.3.a Software espiador de comunicación	37
5.3.b Software de transmisión de una trama de 6 bytes	38
5.3.c Software del Microcontrolador	39
5.3.d Comunicación del Microcontrolador con el programa en alto nivel	41
5.3.e Método de agregar nuevo equipo	42
5.3.f Almacenamiento de los datos en la EEPROM del microcontrolador	43
5.3.g Método de almacenamiento en la memoria EEPROM	43
5.3.h Método de lectura datos de la EEPROM	44
5.3.i Rutina de envío de datos a los equipos	45
5.3.j Rutina de lectura de datos	46
5.3.k Rutina de comparación de datos recibidos con los esperados	47
5.3.l Software del programa final en alto nivel (Java)	48
5.3.l.a Configuración del Dispositivo	48
5.3.l.b Espiador de Comunicación	49
5.3.l.c Lectura de Tramas de Configuración	49
5.3.l.d Agregar un equipo nuevo	50
5.3.l.e Protocolo de comunicación diseñado	51
5.3.l.f Diseño de la Base de Datos	52
Identificación de entidades	52
Identificación de Relaciones	52
Identificación de atributos	52
Equipos:	52
Datos:	52
Pines:	52
Capítulo 6: Análisis de Resultados	53
6.1 Mediciones realizadas con software de monitoreo	53
6.2 Mediciones realizadas con Osciloscopio digital <i>HP Infiniium</i>	53
6.2.a Para equipo ALCATEL	53
6.2.b Para equipo NEC-W	54
6.2.c Para equipo NEC-V	56
6.2.d Para equipo Lucent	58
6.3 Resultados obtenidos con software de monitoreo	59
6.4 Análisis	62
Capítulo 7: Conclusiones y Recomendaciones	65
7.1 Conclusiones	65
7.2 Recomendaciones	66
Bibliografía	67
Apéndice A. Medición de parámetros de la comunicación serial	68
Apéndice B. Mapa de memoria EEPROM, del PIC	69
Apéndice C. Modelo entidad-relación	70

Apéndice D. Código de la rutina de lectura en lenguaje ensamblador.....	71
Apéndice E. Configuración de Pines.....	72
Apéndice F. Manual de Usuario.....	73
Agregar nuevos equipos.....	73
Modo de Configuración.....	74
Modo de Configuración Manual.....	75
Modo de Configuración Automática.....	76
Lectura de Tramas.....	77
Trama de envío.....	77
Trama de respuesta.....	77
Españador de Puerto Serie.....	78
Apéndice G. Glosario, abreviaturas y simbología.....	79
Apéndice H. Información sobre la empresa.....	80
Apéndice I. Descripción del área donde se realizó el proyecto.....	81
Apéndice J. Diagrama de conexión con el uso de Relay's.....	82
Anexo A. Hojas de datos del multiplexor/demultiplexor CD74HC4067.....	83
Anexo B. Hojas de datos de la GAL 22V10.....	84
Anexo C. Hojas de datos del MAX233.....	85
Anexo D. Hojas de datos del microcontrolador PIC16F877A.....	87

INDICE DE FIGURAS

Figura 1.1. Conexión del sistema	1
Figura 1.2. Conexión del sistema con el dispositivo mediador	4
Figura 1.3. Diagrama de conexión del sistema.....	5
Figura 3.1. Diagrama de flujo del sistema.....	10
Figura 5.1. Diagrama de tiempos para multiplexión digital.....	28
Figura 5.11. Diagrama de Flujo de la comunicación entre la computadora y el PIC....	41
Figura 5.12. Diagrama de Flujo de la rutina de agregar nuevo equipo	42
Figura 5.13. Diagrama de Flujo del método de almacenamiento en memoria.....	43
Figura 5.14. Diagrama de Flujo de la lectura de datos de la EEPROM.....	44
Figura 5.15. Diagrama del método de envío de datos	45
Figura 5.16. Diagrama de Flujo del método de lectura de datos.....	46
Figura 5.17. Diagrama de Flujo del método de comparación de datos	47
Figura 5.18. Interfaz de configuración del dispositivo.....	48
Figura 5.19. Interfaz espía de comunicación.....	49
Figura 5.20. Interfaz de lectura de las tramas de datos	49
Figura 5.21. Interfaz para agregar un nuevo equipo.....	50
Figura 6.1. DSR respecto a los datos.....	53
Figura 6.2 DTR respecto a los datos	53
Figura 6.3. CTS respecto a los datos	54
Figura 6.4. RTS respecto a los datos	54
Figura 6.5. DSR respecto a los datos.....	54
Figura 6.6. DTR respecto a los datos	54
Figura 6.7. CTS respecto a los datos	55
Figura 6.8. RTS respecto a los datos	55
Figura 6.9. DSR respecto a CTS.	55
Figura 6.12. Dato enviado por la PC.....	56
Figura 6.13. Dato enviado por el equipo NEC-V.....	56
Figura 6.14. Comunicación entre el equipo NEC-V y la PC.	57
Figura 6.15. DTR vs DSR.	57

Figura 6.16. RTS vs CTS.	57
Figura 6.17. Dato enviado por la PC	58
Figura A.1. Software de comunicación, transmisor de una trama de 6 bytes	68
Figura B.1. Mapa de memoria de almacenamiento de la EEPROM.	69

INDICE DE TABLAS

Tabla 5.1. Diagrama de configuración de pines del cable de monitoreo.....	31
Tabla 6.1. Resultados obtenidos mediante software de monitoreo	53
Tabla 6.2. Dato de envío de la PC al equipo Alcatel.....	59
Tabla 6.3. Respuesta por parte del equipo Alcatel al microcontrolador.....	59
Tabla 6.4. Dato de envío y respuesta para el equipo NEC.	59
Tabla 6.5. Caracteres enviados por la PC al equipo Lucent.....	60
Tabla 6.6. Respuesta por parte del equipo Lucent al microcontrolador	60
Tabla 6.7. Caracteres enviados por la PC al equipo Ericsson	61
Tabla 6.8. Respuesta por parte del equipo Ericsson al microcontrolador	61
Tabla A.1. Configuración en comunicación serial	68
Tabla E.1. Configuración de pines para las 4 marcas de equipos evaluados	72

Capítulo 1: Introducción

1.1 Problema existente e importancia de su solución:

El problema a resolver tiene lugar en el área de telecomunicaciones, en el departamento de Operación y Mantenimiento del ICE, específicamente en la Sala de Transmisión, en la cual se encuentran los Switches SDH, utilizados para el desarrollo del proyecto.

Con el desarrollo de este proyecto se pretende mejorar un sistema existente, que trata de la comunicación entre una computadora y los equipos terminales de telecomunicación (SDH), en ésta se intercambia información con equipos de diferentes marcas como son: Alcatel, NEC, Erickson, Lucent, entre otros, como se ilustra en la figura 1.1, cabe destacar que para los equipos de diferente marca la conectividad y protocolo de comunicación puede ser diferente.

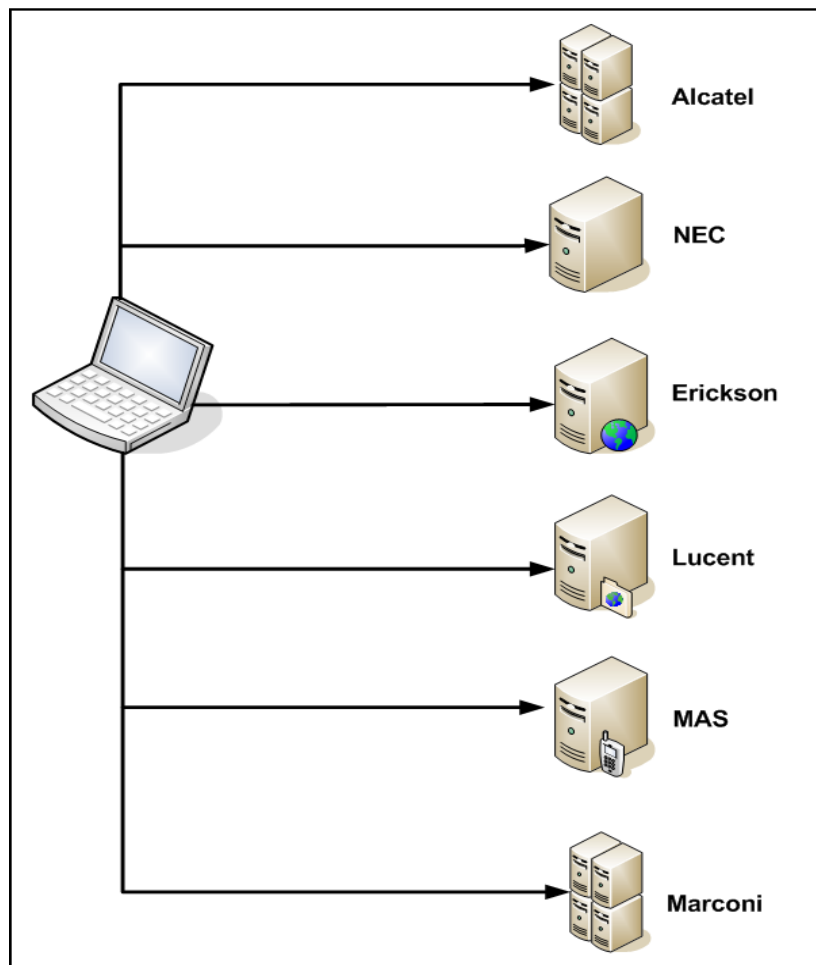


Figura 1.1. Conexión del sistema

Al haber distinta conectividad y distintos protocolos de comunicación en función de la marca con la que se opera, sobresale el problema de tener que utilizar distintas formas de conectividad, de acuerdo con el equipo que se desea operar, ya que al tratarse de distintos protocolos y diferente conectividad se debe buscar el cable de conectividad indicado de cada equipo, en caso de que se pierda o dañe un cable de conectividad, en una zona de difícil acceso a conseguir otro cable de conectividad, viene a ser un problema tanto de tiempo, económico como de eficiencia. Es evidente que esta labor hace demandar más tiempo en la desconexión, búsqueda y conexión de los cables de conectividad para los distintos protocolos utilizados por los diferentes equipos de telecomunicaciones, por lo cual es de gran ayuda un dispositivo que sirva de mediador universal, independiente de la marca del equipo con el cual se este trabajando.

Es importante hacer notar en esta sección que el marco del problema se limitará a probar con cuatro marcas de equipos, por cuestión alcances de la presente investigación.

Con la implementación de este dispositivo mediador universal se pretende lograr aumentar los tiempos de acceso a los distintos equipos, con lo cual se obtendrá una mayor eficiencia.

Los beneficios derivados de la solución del problema son los siguientes:

- Menor tiempo de acceso a los equipos.
- Mayor eficiencia en el trabajo realizado.
- Disponibilidad de un dispositivo mediador universal.
- Disponibilidad de un dispositivo inteligente, el cual sea capaz de identificar el equipo conectado, mediante protocolos de reconocimiento *hand-shaking*, para equipos predefinidos.

1.2 Solución seleccionada:

Los requerimientos establecidos por parte de la empresa en cuanto a los resultados esperados del proyecto son:

- Diseñar un dispositivo mediador evaluado con cuatro marcas, como lo son: Alcatel, NEC, Ericsson y Lucent.
- Realizar una **configuración automática**[♦] del dispositivo mediador de acuerdo al equipo que se conecta.
- Realizar protocolos de reconocimiento (*hand shaking*) de equipos.
- Comunicación serial entre la computadora y el dispositivo.
- Realizar un programa para la computadora, que maneje la configuración manual (seleccionando el equipo) ó automática (de manera inteligente) de los equipos.
- Presentar un diseño funcional para las cuatro marcas.
- El diseño deberá cumplir con estándares técnicos de los distintos fabricantes.

[♦] La Configuración Automática, se refiere al proceso de reconocer el equipo (Marca) conectado al dispositivo mediador de manera automática o inteligente.

El diseño de este dispositivo debe contemplar los estándares y protocolos de las distintas marcas. La solución seleccionada para solucionar esta problemática, es una solución que involucra la electrónica digital para su control, que entre otras funciones, principalmente debe reconocer el equipo conectado (mediante protocolos), así como realizar la ***matriz de conexión** que permita comunicar la computadora con los equipos. Para tener un mejor panorama de la solución proyectada obsérvese la siguiente figura:

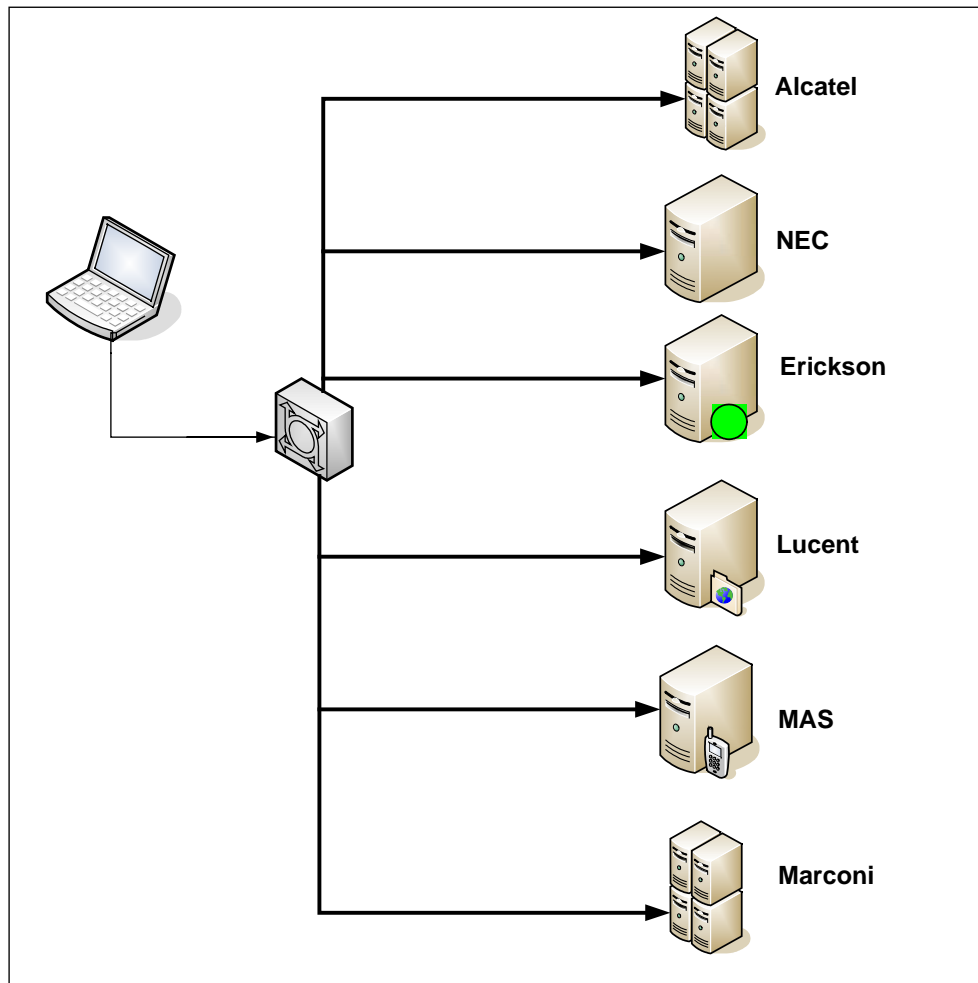


Figura 1.2. Conexión del sistema con el dispositivo mediador

Como se observa en la figura anterior la función principal del dispositivo es actuar de mediador entre la computadora y los equipos, realizando la matriz de conexión correspondiente para cada equipo.

Cabe destacar, que con las cuatro marcas con las que se evalúa la solución, los conectores (a nivel de *hardware*) que se utilizan, pueden diferir

* matriz de conexión: Se refiere a la conectividad de *Hardware* necesaria, para establecer la comunicación con los distintos equipos, o sea a la configuración de pines, de acuerdo a los distintos protocolos de comunicación.

entre marcas, ya que los equipos Alcatel y NEC, utilizan protocolo RS232, serie directo, el Ericsson, utiliza RS232 directo pero únicamente tres líneas, y el equipo Lucent presenta una particularidad y es que este tipo de protocolo posee un conector DB9 en una terminal, y un conector RJ45 en la otra terminal, la configuración de pines a nivel de *hardware* es propia del fabricante, y por ende el protocolo utilizado no es un protocolo estandarizado, presentándose el problema de un protocolo propio del fabricante, para lo cual se realizaron las pruebas necesarias para determinar dicho protocolo.

Al escogerse una solución digital para el desarrollo del proyecto, es muy importante una adecuada escogencia del microcontrolador, el cual debe tener como característica fundamental una comunicación serie, necesaria para realizar los protocolos de reconocimientos y comunicación con la computadora, es por lo tanto que se escogió el microcontrolador 16F877A, de Microchip. Además se necesita a nivel de *hardware*, multiplexar de forma analógica la señal, debido a que la conectividad de *hardware* es distinta para cada equipo, por lo tanto la matriz de conexión (a nivel de *hardware*) de cada equipo con la computadora, se implementó mediante una GAL 22V10D de Lattice. En el siguiente diagrama muestra en forma general el sistema a utilizar.

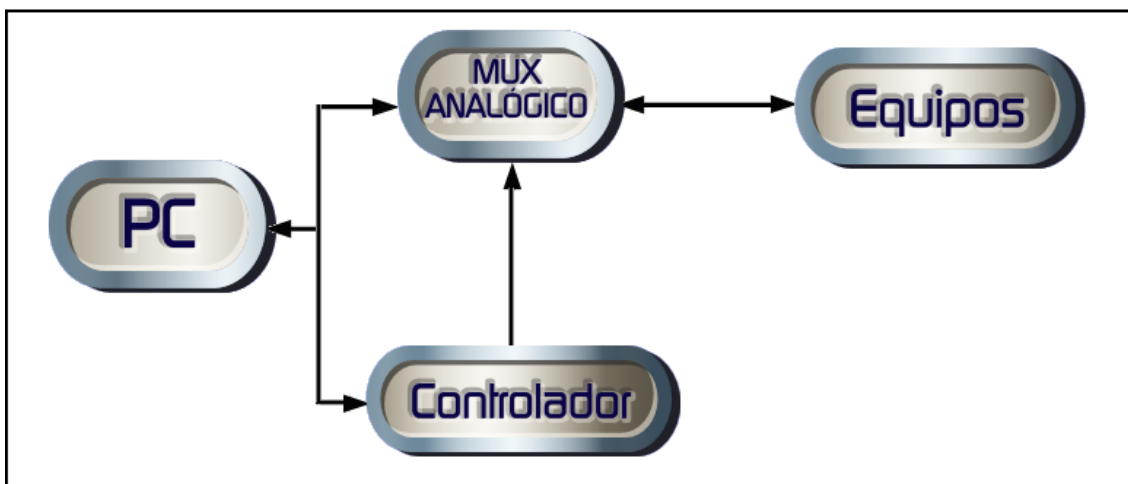


Figura 1.3. Diagrama de conexión del sistema

PC:

Este bloque representa físicamente la computadora, donde se encuentra los programas de cada equipo, así como el programa que gobierna el dispositivo, que entre otras funciones, la función principal es la selección del modo de configuración automática o manual(en la computadora se escoge el equipo).

Mux Analógico:

Este bloque representa la multiplexación de señales, y corresponde directamente a la matriz de conexión controlada por el microcontrolador, estableciendo de esta manera la comunicación con los distintos equipos, o sea a la configuración de pines, de acuerdo a los distintos protocolos de comunicación.

Controlador:

Este es el responsable directo del control del sistema, reconociendo, y configurando el puerto de acuerdo al equipo conectado, además de realizar rutinas de reconocimiento, mediante los protocolos adecuados.

Equipos:

Este bloque representa los equipos multiplexores SDH, de las distintas marcas, con el cual se debe de comunicar la computadora, para los cuales se debe de realizar la configuración de pines en el bloque de Mux analógico.

Capítulo 2: **Meta y Objetivos**

2.1 Meta

Diseñar un dispositivo para un mayor rendimiento y eficiencia en el momento de trabajar con diferentes equipos como Alcatel, NEC, Ericsson y Lucent, logrando disminuir los tiempos de interconexión con los equipos, además de automatizar el proceso de trabajo con los equipos, tomando en cuenta los distintos protocolos y estándares de comunicación que se utilizan.

2.2 Objetivo general

Diseñar un dispositivo mediador universal, el cual sirva para poder interconectar una computadora con los distintos equipos, mediante una ***matriz de comunicación** y que posea un control a través de un programa en la computadora. Ofreciendo un sistema automatizado en el proceso de trabajo con los equipos, cumpliendo con estándares y protocolos necesarios.

2.3 Objetivos específicos

2.3.a Objetivos de hardware

- Diseñar e implementar un circuito que multiplexe las señales, para la comunicación.
- Escoger el microcontrolador adecuado para la solución al problema.
- Diseñar e implementar una comunicación del dispositivo con la computadora.
 - Establecer comunicación con protocolo RS232.
 - Escoger e implementar de componentes para la comunicación.

* matriz de conexión: Se refiere a la conectividad de *Hardware* necesaria, para establecer la comunicación con los distintos equipos, o sea a la configuración de pines, de acuerdo a los distintos protocolos de comunicación.

- Determinar los estándares de comunicación con los distintos equipos.
 - Investigar los estándares utilizados por cada equipo.
 - Realizar pruebas de medición que corroboren la investigación.
- Determinar los protocolos de reconocimiento *hand shaking* de las distintas marcas.
 - Investigar con los fabricantes los protocolos de reconocimiento.
 - Realizar mediciones para corroborar los protocolos, o para determinar dichos protocolos, para los casos en que no se pueda conseguir.
- Diseñar e implementar una comunicación del dispositivo con los equipos.
 - Determinar los estándares necesarios para establecer comunicación del dispositivo de control con los equipos de comunicación.
 - Establecer comunicación con los equipos, mediante una multiplexación de las señales.

2.3.b Objetivos de software

- Diseñar un programa el cual controle el manejo ^o manual ó ^{oo} automático del dispositivo mediador, configurando directamente el equipo a utilizar.
- Diseñar las rutinas del protocolo de reconocimiento (*hand shaking*) para los distintos equipos.
- Diseñar las rutinas de comunicación entre el sistema de control y la computadora.
- Diseñar las rutinas de control de la matriz de conexión.

^o El manejo manual, quiere decir una escogencia de la marca del equipo a utilizar, en cuyo caso el dispositivo de control configura su matriz de conexión de acuerdo a los estándares y protocolos preestablecidos.

^{oo} En el manejo automático el dispositivo realiza rutinas de reconocimiento (*hand-shaking*), determinando el equipo a interconectar.

2.3.c Objetivos de documentación

- Investigar la comunicación serie y así como los distintos protocolos.
- Investigar los protocolos de reconocimiento de los distintos equipos.
- Elaborar un documento final del anteproyecto.

2.3.d Objetivos de implementación

- Realizar un dispositivo funcional que sirva de interconexión entre la computadora y los distintos equipos.
- Determinar los protocolos de reconocimiento y de comunicación con los distintos equipos.

3.1 Descripción del sistema

El proceso actual del sistema, como se ha mencionado anteriormente, consta de una computadora, un medio de transmisión y el equipo a conectar;

En este sistema existe el inconveniente de que cada equipo puede necesitar distinto tipo de estándar de conectividad, es por lo tanto que se busca mejorar éste proceso, a través de un dispositivo mediador universal, con el cual se busca automatizar este proceso, mediante la reducción de tareas efectuadas. En la figura 4.1 se muestra la secuencia de etapas del proceso.

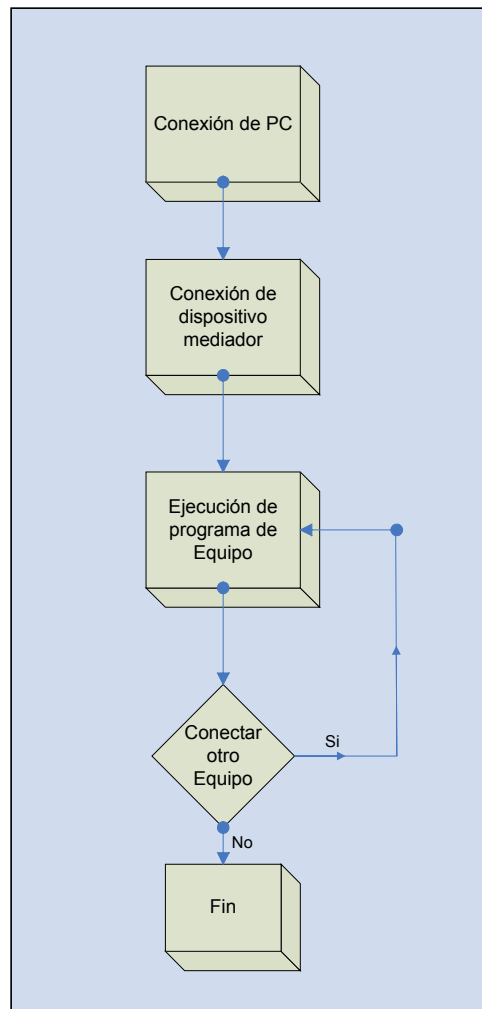


Figura 3.1. Diagrama de flujo del sistema.

3.2 Antecedentes Bibliográficos

La comunicación con los equipos viene a ser mediante comunicación serie, en algunos casos los equipos utilizan la conexión null modem, otros utilizan terminales completas de modem, mientras que otros pueden utilizar cable UTP de red, por lo tanto en esta sección se describirá ampliamente.

3.2.1 Comunicación asíncrona.

Una de las principales razones por la que estudiaremos la comunicación asíncrona es para poder tener la posibilidad de hablar con un PC a través de su puerto serie, o puerto USB a través de un '*USB to Serial*'.

3.2.1.a El protocolo RS-232.

“ Elementos necesarios para una comunicación serie asíncrona entre dos dispositivos.

Físicos.

- Los dispositivos deben tener una tierra en común, las tierras de los circuitos de los dos dispositivos deben estar conectadas.
Un cable de conexión de la salida del puerto emisor a la entrada del puerto receptor.
- Si se quiere comunicación en los dos sentidos.
Otro cable de conexión del puerto que envía del receptor al receptor del transmisor.
- Los niveles de los voltajes del transmisor deben ser aceptables por el receptor. La salida del transmisor se debe conectar a la entrada del receptor.
- La polaridad de la señal debe ser la misma.

Software.

- El *baud rate* debe ser el mismo en los dos equipos. (Velocidad de envío)
- El número de stop bits, data bits y paridad. deben ser iguales.
- Cualquier protocolo de "*Hand shaking*" necesario debe ser habilitado.”[1]

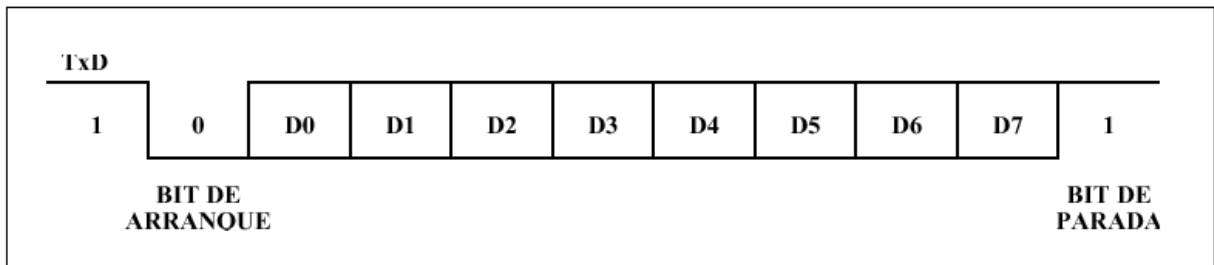


Figura 3.2. Formato de Transmisión de datos

3.2.1.b Generalidades:

- “Describe la conexión entre un equipo terminal de datos (*Data Terminal Equipment, DTE*) y un equipo transmisor de datos (*Data Carrier Equipment, DCE*) para la transmisión serie de información. El ejemplo típico es la comunicación entre el computador y el módem.
- Actualmente se utiliza casi siempre que se deben enviar datos en serie.

3.2.1.c Funcionamiento:

- Se utilizan voltajes entre 3 y 15 V para el nivel alto y entre -3 y -15 V para el nivel bajo.
- Tenemos tres tipos de funcionamiento:
 - **Simplex**: Sólo se envían datos del DTE al DCE.
 - **Half-duplex**: La comunicación funciona en ambos sentidos pero nunca al mismo tiempo.
 - **Full-duplex**: La comunicación funciona en ambos sentidos de forma simultánea porque existen dos líneas de datos, una de envío y otra de recepción.
- Se pueden utilizar conectores de 9 a 25 pines.

3.2.1.d Limitaciones del RS-232:

- Velocidad.
- Número de puertos.
- Flexibilidad.
- Facilidad de configuración, drivers.”[2]

3.2.2 ⁹La comunicación USB:

La comunicación mediante el puerto USB, a pesar de que no se implementa es importante hacer mención de éste tema, ya que se plantea como opción alternativa y complementaria a la solución planteada. Y ésta se refiere explícitamente a la comunicación del dispositivo mediador con la computadora (software de control del dispositivo). Directamente lo que se plantea es una modificación en el hardware del dispositivo mediador.

3.2.2.a Generalidades

- “USB: Universal Serial Bus
- Interfaz serie y bus externo para conectar periféricos al computador.
- Permite configurar los periféricos de forma sencilla: el computador identifica automáticamente el dispositivo que se conecta mientras opera (‘en caliente’) y lo configura sin tener que instalar *drivers* específicos.
- Además evita la necesidad de aumentar el número de puertos del ordenador.
- Permite dos tasas de transferencia diferentes: 1.5 MB/s para dispositivos lentos (joysticks, ratones,...) y 12 MB/s para dispositivos con mayor ancho de banda (discos, CDROMS,...).

⁹ Es importante realizar la salvedad de que el puerto USB, no se contempla en los objetivos de la solución proyectada.

3.2.2.b Funcionamiento:

- Su principal característica es que es rápido y muy sencillo:
Se utiliza un único conector para conectar a través de un bus serie Todos los dispositivos. Soporta la conexión de hasta 127 dispositivos.
- El bus USB tiene dos líneas para transmitir datos: D+ y D-.
- La información se transmite de forma balanceada. La señal se codifica de manera que incluya junto con los datos la información de sincronización.
- También hay dos líneas de alimentación: *Vbus* y *GND* (tierra).
- Evitan la necesidad de utilizar fuentes de alimentación externas. “[2]

3.2.3 UTP (Par trenzado no blindado):

Cable que consiste en un par o más de cables (que se utilizan en una gran variedad de aplicaciones de red) que están enfundados en plástico. UTP es popular porque es muy maleable y no ocupa tanto espacio como los STP y otros cables.

3.2.3.a Tipos de UTP

Tipo 3 enviar hasta 16Mhz de ancho de banda Calidad telefónica. Distancia 7-10 cm por trenza.

Tipo 4 enviar hasta 20 Mhz de ancho de banda

Tipo 5 enviar hasta 100 Mhz de ancho de banda Calidad de datos. Distancia 0,5 a 1 cm por trenza.

3.2.3.b Velocidades utilizando el cable UTP.

100 Mbps en 100 m

2 Mbps en 1500 m

60 Kbps en líneas telefónicas

3.2.3.c Protocolo para el UTP:

El protocolo utilizado para el cable UTP, es el 100BaseT y 100BaseT4 de Ethernet, para una longitud máxima de 100m

3.2.3.d Conector del cable UTP:

EIA/TIA especifica un conector de RJ-45 para el cable de UTP. Las letras RJ representan la sota registrada, y el número 45 se refiere a una sucesión de la instalación eléctrica específica. El RJ-45 que el conector del fin transparente muestra a ocho alambres coloreados. Cuatro de los alambres llevan el voltaje y son considerados “la punta” (T1 a través de T4). Los otros cuatro alambres se conectan con tierra y se llaman “el anillo” (R1 a través de R4). La Punta y anillo son condiciones que originaron en los días tempranos del teléfono. Hoy, estas condiciones se refieren al positivo y el alambre negativo en un par. Los alambres en el primer par en un cable o un conector se designan como T1 y R1. El segundo par es T2 y R2, y así sucesivamente.

3.2.4 Modelos de datos

Los modelos de datos pueden utilizarse para describir un conjunto de datos y las operaciones para manipularlos. En la literatura se puede encontrar dos tipos de modelos de datos: los modelos conceptuales y los modelos lógicos. Éstos primeros se utilizan para representar la realidad a un alto nivel de abstracción, además se puede construir una descripción de la realidad fácil de entender. En los modelos lógicos, las descripciones de los datos corresponden a la estructura física de la base de datos.

En el proceso de diseño de las bases de datos se usa primero los modelos conceptuales para lograr una descripción de alto nivel de la realidad, y luego se transforma el esquema conceptual en un esquema lógico.

Los modelos conceptuales deben ser buenas herramientas para representar la realidad, por lo que deben poseer las siguientes cualidades:

- **Expresividad:** deben tener suficientes conceptos para expresar perfectamente la realidad.
- **Simplicidad:** deben ser simples para que los esquemas sean fáciles de entender.
- **Minimalidad:** cada concepto debe tener un significado distinto.

- Formalidad: todos los conceptos deben tener una interpretación única, precisa y bien definida.

3.2.4.a Metodología de diseño de bases de datos

El diseño de una base de datos es un proceso complejo que abarca decisiones a muy distintos niveles. Lo cual se controla mejor si se descompone el problema en subproblemas y se resuelve cada uno de estos subproblemas independientemente, utilizando técnicas específicas.

Es por lo tanto que el diseño de una base de datos se descompone en diseño conceptual, diseño lógico y diseño físico, y de esta manera se puede realizar un mejor análisis y un mejor planteamiento de la solución. En la cual un modelo conceptual es un lenguaje que se utiliza para describir esquemas conceptuales. El objetivo del diseño conceptual es describir el contenido de información de la base de datos y no las estructuras de almacenamiento que se necesitarán para manejar esta información. Y el modelo lógico se utiliza para especificar esquemas lógicos (modelo relacional, modelo de red, etc.).

“Cada uno de los esquemas conceptuales se componen de entidades, relaciones, atributos, dominios de atributos e identificadores. El esquema conceptual también tendrá una documentación, que se irá produciendo durante su desarrollo. Las tareas a realizar en el diseño conceptual son las siguientes:

1. Identificar las entidades.
2. Identificar las relaciones.
3. Identificar los atributos y asociarlos a entidades y relaciones.
4. Determinar los dominios de los atributos.
5. Determinar los identificadores.
6. Determinar las jerarquías de generalización (si las hay).
7. Dibujar el diagrama entidad-relación.
8. Revisar el esquema conceptual local con el usuario. “[3].

1. Identificar las entidades

En primer lugar hay que definir los principales objetos que interesan al usuario. Estos objetos serán las entidades. Una forma de identificar las entidades es examinar las especificaciones de requisitos de usuario. En estas especificaciones se buscan los nombres o los sintagmas nominales que se mencionan (por ejemplo: número de empleado, nombre de empleado, número de inmueble, dirección del inmueble, alquiler, número de habitaciones). También se buscan objetos importantes como personas, lugares o conceptos de interés, excluyendo aquellos nombres que sólo son propiedades de otros objetos. Por ejemplo, se pueden agrupar el número de empleado y el nombre de empleado en una entidad denominada empleado, y agrupar número de inmueble, dirección del inmueble, alquiler y número de habitaciones en otra entidad denominada inmueble.

2. Identificar las relaciones

Una vez definidas las entidades, se deben definir las relaciones existentes entre ellas. Del mismo modo que para identificar las entidades se buscaban nombres en las especificaciones de requisitos, para identificar las relaciones se suelen buscar las expresiones verbales (por ejemplo: oficina tiene empleados, empleado gestiona inmueble, cliente visita inmueble). Si las especificaciones de requisitos reflejan estas relaciones es porque son importantes para la empresa y, por lo tanto, se deben reflejar en el esquema conceptual.

3. Identificar los atributos y asociarlos a entidades y relaciones

Son atributos los nombres que identifican propiedades, cualidades, identificadores o características de entidades o relaciones.

Lo más sencillo es preguntarse, para cada entidad y cada relación, ¿qué información se quiere saber de...? La respuesta a esta pregunta se debe encontrar en las especificaciones de requisitos.

3.2.4.b Modelo Entidad/Relación

El modelo entidad-relación es el modelo conceptual más utilizado para el diseño conceptual de bases de datos. Fue introducido por Peter Chen en 1976. El modelo entidad-relación está formado por un conjunto de conceptos que permiten describir la realidad mediante un conjunto de representaciones gráficas y lingüísticas.

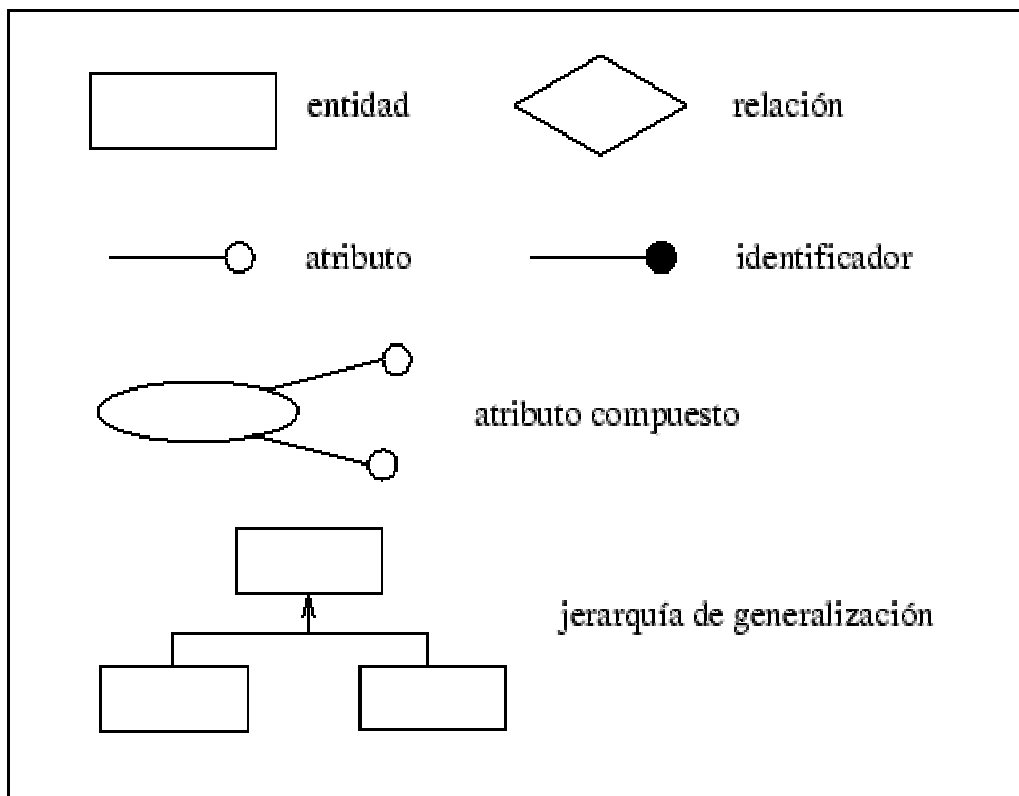


Figura 3.3. Conceptos del modelo entidad-relación extendido[4]

3.3 Descripción de los principales principios de software y electrónicos relacionados con la solución del problema

3.3.a Principios de software:

En cuanto a los principios de software, cabe hacer notar la importancia de los protocolos de comunicación, es por ende que en el sistema digital debe contar con algoritmos que se comuniquen con el protocolo necesario según sea el equipo que se conecte.

También se debe de realizar algoritmos que logren la comunicación de un programa de alto nivel de interacción con el usuario, con el dispositivo mediador universal.

3.3.b Principios electrónicos:

Los principios electrónicos relacionados con la solución del problema, radican principalmente en el control de la ^ζmatriz de conexión y la matriz misma, por lo tanto en cuanto al control de la matriz, los principios electrónicos relacionados son principios de electrónica digital, como un microcontrolador, su alimentación, sus puertos y sus líneas de control.

3.3.c Principios matemáticos:

Los principios matemáticos se concentran principalmente en el cálculo de la matriz de conexión, así como el valor de algunos valores relacionados de algunos componentes.

^ζ matriz de conexión: Se refiere a la conectividad de *Hardware* necesaria, para establecer la comunicación con los distintos equipos, o sea a la configuración de pines, de acuerdo a los distintos protocolos y estándares de comunicación.

Capítulo 4: Procedimiento Metodológico

4.1 Reconocimiento y definición del problema

La caracterización de protocolos constituye un paso trascendental para la determinación de los diferentes equipos, y es por tanto fundamental realizar pruebas de campo, que tienen como fin realizar mediciones de protocolos de reconocimiento (*hand-shaking*) para cada uno de ellos, fundamental para el sensado y configuración del equipo conectado. También paralelo a esta actividad son importantes las entrevistas que se puedan realizar al personal que los utiliza.

El proceso de determinación de protocolos, es un pilar fundamental del proyecto, ya que una medición errónea de este dato, llevaría a un mal desarrollo por lo que se procedió a realizar distintas formas de medición.

En un principio, se realizaron mediciones por medio de un programa en alto nivel (lenguaje Java), y se leyeron los datos en forma bidireccional, pero como se explicará más adelante, hay una pérdida de bits en esta lectura, por lo que se procedió a realizarlas por medio de un osciloscopio, con el que se obtuvieron datos precisos, pero resultó un inconveniente para la lectura del dato a 38400 baudios, ya que ésta es muy rápida con respecto a la escala del osciloscopio, por lo que se buscó una opción alternativa y más eficiente, mediante un microcontrolador, en el que se corroboraron los datos leídos con las mediciones realizadas en un proceso iterativo.

En el proceso anterior fue importante la ayuda brindada por los operadores de la sala de Transmisión del departamento de Operación y Mantenimiento, tanto en la búsqueda y uso de equipos, como en sus respectivos programas de control.

4.2 Obtención y análisis de información

El proyecto cuenta con la mejor disposición del sector de Operación y Mantenimiento del ICE, por lo que mucha información se obtuvo por medio de entrevistas con personal de la Institución, especialmente con el uso de equipos, así como de sus respectivos software de control.

La información preliminar se obtuvo especialmente del personal de la Institución, aunque parte de la más relevante, que se intentó investigar con los proveedores no fue posible conseguirla, por lo que se buscaron soluciones alternas para obtener dicha información, que se explicarán con más detalle en el siguiente capítulo.

La evaluación de los protocolos de comunicación se realizó mediante mediciones de prueba, y comprobación de respuesta por parte de los equipos; además, la investigación de distintas configuraciones y de los diferentes estándares, se apoyó con el uso de Internet, consultas bibliográficas, pero principalmente con los manuales y hojas de datos de los equipos.

En orden cronológico, se investigó con proveedores de las diferentes marcas, paralelamente en los manuales técnicos de los equipos SDH, con lo que no se obtuvo la información necesaria para desarrollar el proyecto, por lo que se procedió a realizar mediciones, con un programa espiador en alto nivel (Java), después con un Osciloscopio y por último con el circuito de control.

Éstas representaron la información necesaria para el desarrollo del proyecto, ya que con éstas se obtuvieron los datos de envío y de respuesta de cada uno de los equipos, con lo que el trabajo de su reconocimiento se pudo lograr de manera satisfactoria.

4.3 Evaluación de las alternativas y síntesis de una solución

El proyecto está destinado a la elaboración de un dispositivo mediador universal. Por consiguiente, la elección de las soluciones considera los resultados de las sugerencias de los asesores, investigación bibliográfica, pruebas de medición, disposición de equipos y análisis de costos.

Desde un principio la solución planteada, trata de un control digital, que involucra una multiplexación de las señales, en donde fue de gran ayuda las sugerencias de los profesores lectores del proyecto, que encaminaron dicha solución entre otras alternativas, a una solución mediante un dispositivo lógico programable (GAL), en el que la propiedad combinacional es la característica necesaria para el módulo de multiplexación.

Durante el desarrollo del proyecto, se implementó temporalmente una solución digital sincrónica, mediante un microcontrolador para la multiplexación, y en este proceso se encontró un problema en los tiempos de retardo de las señales, pues el microcontrolador es un dispositivo sincrónico que realiza una única tarea por cada ciclo de instrucción, y como son 9 líneas del puerto serie, se producirá un retardo **mínimo** de 9 ciclos de instrucción, para que una señal o un pin del microcontrolador se refresque a su valor actual. Como el ciclo de instrucción del microcontrolador seleccionado, tiene un valor de 200ns, se produce un retardo **mínimo** de 1,8 μ s, con lo que se evidencia, que para una velocidad de 38400 baudios, el significado de éste retardo no es tolerable. Con lo que se puede concluir que para un microcontrolador con ese periodo de ciclo de instrucción, no se puede realizar ésta solución, por lo que se discriminó, optando por una lógica combinacional programable; además de ser una solución de menor costo.

Para verificar la validez de la solución se realizaron pruebas de medición, tanto en el laboratorio del ITCR, como en la sala de transmisión del ICE.

La evaluación de una solución propuesta radica en el tamaño, rendimiento y en el cumplimiento de estándares y protocolos de comunicación, del dispositivo mediador (hardware de control).

4.4 Implementación de la solución

Como se mencionó anteriormente la solución inicial propuesta, sufrió un ligero cambio en la forma de multiplexar la señal, y es que en esta tarea se utilizó un dispositivo lógico programable (GAL 22V10).

Los mecanismos utilizados para evaluar la propuesta de solución, consisten esencialmente en pruebas de campo, ya que estas son importantes para certificar la funcionalidad del proyecto.

En el hardware de control, se procedió a realizar mediciones modulares, lo cual quiere decir que se verificó el hardware de control por módulos, primero se probó el módulo de comunicación con la computadora, así como rutinas de pruebas del hardware de control, y seguidamente se comprobó la comunicación con el software de control.

Esto pasó a ser un proceso iterativo, principalmente en las pruebas de comunicación con un segundo puerto de la computadora, el cual simulaba ser el equipo de Telecomunicaciones, debido a la velocidad de transmisión y de lectura por parte del hardware de control. A pesar de que se calculó y se ajustó el tiempo que transcurre en efectuar dichas rutinas no fue suficiente, por lo que se procedió a realizar un proceso iterativo, reduciendo éste tiempo hasta alcanzar una comunicación exitosa.

Después se realizaron mediciones en la Sala de Transmisión, del cuarto piso del antiguo Edificio del ICE, las cuales consistieron principalmente en medición de datos, tanto de transmisión hacia los equipos, como los de respuesta, mediante el dispositivo mediador universal (ó hardware de control).

Seguidamente se realizaron pruebas en el los laboratorios del ITCR, las cuales consistían en comprobar el funcionamiento de la GAL 22V10.

Y por último las pruebas de comprobación consistían en transmitir el dato de saludo inicial de *hand-shaking*, respectivo para cada equipo, y comprobar con el dato esperado de respuesta por parte del equipo, interactuando de manera directa con la computadora, mediante un software de control, dichas mediciones fueron realizadas en el ICE.

El medio de difusión será la documentación final que solicita la Institución, donde se indicarán los resultados obtenidos, restricciones y recomendaciones, y se facilitará un manual de usuario para poder operar el hardware de control.

4.5 Reevaluación y rediseño

El dispositivo se podría comunicar a través del puerto USB ó infra-rojo de la computadora, que vendría a ser muy importante, debido a que en las computadoras modernas especialmente las *laptop* se está eliminando el puerto serie, con el que se debe de realizar un programa que redireccione el puerto de comunicación, de los programas de los equipos de puerto serie a puerto *USB* o infra-rojo. También se podría hacer de forma tal que la alimentación del dispositivo la tome del medio de transmisión, como sería el caso del puerto *USB*.

Una alternativa de una posible mejora al proyecto, se podría dar especialmente, en que el dispositivo reconozca comandos, y no únicamente el saludo de *hand-shaking*, y que vendría a ser de gran importancia, ya que la comunicación con los equipos multiplexores SDH, se podría realizar remotamente, y de esta manera poder chequear fallas, y realizar *troubleshooting*, desde un solo punto, sin necesidad de desplazarse. Vale la pena resaltar que al haberse implementado la solución con bases de datos, se deja la posibilidad de poder acceder la información de los equipos, con un software de comunicación remota, y por lo tanto se puede concluir que dicho proyecto deja las puertas abiertas a una continuación (final abierto).

Capítulo 5: Descripción detallada de la solución

La solución implementada, se explicará en forma cronológica, de manera que se pueda detallar paso por paso, el proceso de solución.

5.1 Análisis de soluciones y selección final:

En esta sección, se dará como premisa una descripción de las soluciones alternas, diseñadas para obtener aspectos relevantes en la solución final del problema mismo.

Primeramente, como parte de la solución, se fabricó un cable de monitoreo, cuyo objetivo es leer los datos de comunicación intercambiados entre la computadora y los equipos de Telecomunicaciones, con lo que se busca, realizar la lectura del protocolo de *hand-shaking* entre ambos, y de esta manera poder reproducirla para el reconocimiento, mediante el *hardware* de control; el diagrama del cable mencionado se muestra en la siguiente sección en la figura 5.2, y su configuración se puede observar de manera más detallada en la tabla 5.1.

Con la fabricación del cable es indispensable un software de monitoreo, que sea capaz de leer de manera simultánea, los dos puertos de la computadora, por lo tanto se diseñó de forma tal, que fuese capaz de observar los datos de transmisión de forma bidireccional, además de ciertas señales de control, como lo son: CTS (*Clear To Send*), DTR (*Data Terminal Ready*), RI (*Ring*), y CD (*Carrier Detect*); este software se explica más en detalle en la sección de descripción de *software* de este capítulo.

Con la comunicación establecida, y con el conocimiento de la trama que envía la computadora a los equipos, se puede establecer una interacción con el software diseñado, y de esta forma verificar que el dato recibido, sea el mismo que el dato espiado previamente a las distintas velocidades respectivamente, para así discriminar la velocidad de comunicación de los equipos.

Los resultados obtenidos en esta prueba no fueron satisfactorios, ya que a distintas velocidades, se obtuvo una misma respuesta, la cual es un cero (0) en código ASCII, lo que no corresponde al dato esperado, espiado previamente.

Y con base en estos resultados, se investigó el protocolo de la trama de comunicación para el puerto serie, para los casos en que se envía más de un carácter en la trama, o sea para los casos en que el dato es superior a los 8 bits (1 byte), pero en las investigaciones realizadas especialmente en Internet, y en libros, no se encontró ninguna información que ayudara en éste tema, por lo que se procedió a realizar dicha medición en el laboratorio del ITCR. A través del diseño de un software, en el que se enviaron tramas de más de 1 byte (refiérase al Apéndice A), controlando así características como, tamaño de la palabra, bits de parada, control de flujo y paridad, y mediante un osciloscopio, se realizaron mediciones, para poder observar los bits de espaciamento, entre los bits correspondientes al dato mismo, construyéndose así la tabla A.1(refiérase al apéndice A), la cual contiene información necesaria para poder caracterizarlos. De ésta manera, se puede entender mejor aún la trama, y por lo tanto el protocolo de comunicación para cada equipo.

Con los datos obtenidos, se descarta la solución planteada inicialmente mediante el software de control, para leer o espiar la trama de comunicación, ya que de antemano no tenemos las características de comunicación de los equipos (como bits de parada, longitud de la palabra, etc), por lo tanto se vuelve necesario, la búsqueda de una solución alterna.

En consecuencia se realizaron mediciones por medio de un Osciloscopio (resultados mostrados en el siguiente capítulo), con las que se obtuvieron los datos reales y exactos de la trama de comunicación, que constituyó una pieza clave en el desarrollo del proyecto, y por tanto parte importante en el proceso de solución.

A pesar de su exactitud, el proceso de lectura se hizo realmente difícil para los equipos Lucent, ya que los Lucent, poseen una velocidad de comunicación de 38400 baudios, lo cual representa una tarea difícil de realizar, por que la escala del Osciloscopio, debe ser lo suficientemente amplia, para determinar el dato, además de que se debe hacer un seguimiento de lectura de bits a través de toda la trama. Y en especial el dato de respuesta por parte del equipo se vuelve más difícil, ya que no se puede capturar la trama de respuesta.

Como última opción y solución final, se pensó en una lectura de datos menos lenta (menor tiempo en realizar la lectura de la trama de datos), y más

exacta, la cual se implemento mediante un microcontrolador. Con éste se logra realizar una medición de la trama completa, contemplando todos los bits del protocolo para cada equipo, que representa de manera implícita* el dato de *hand-shaking*.

Durante el desarrollo de la solución digital, se presentaron problemas de comunicación entre el dispositivo (*hardware* de control), y los equipos de Telecomunicaciones, y mediante pruebas de búsqueda de error, se determinó que dichos problemas eran producidos por un desfase en los tiempos de lectura por parte del microcontrolador, respecto a los tiempos de la señal de transmisión de la computadora, por lo que se calculó el tiempo de espera entre bits, y el tiempo transcurrido en la ejecución de la rutina del microcontrolador, como se muestra:

5.1.a Cálculo del periodo de lectura y transmisión:

Periodo de 1 bit para equipos Alcatel, Ericsson y NEC W:

$$T_b = \frac{1}{9600} = 104 \mu s$$

Periodo de 1 bit para equipos NEC V:

$$T_b = \frac{1}{4800} = 208 \mu s$$

Periodo de 1 bit para equipos Lucent:

$$T_b = \frac{1}{38400} = 26 \mu s$$

5.1.b Cálculo del tiempo transcurrido en la ejecución de la rutina:

Ciclo de instrucción = 200ns.

Cantidad de instrucciones = 29^Ψ.

$$T_T = 29 * 200 = 5,8 \mu s$$

Los tiempos calculados anteriormente, fueron fundamentales para lograr una comunicación exitosa, a pesar de que las pruebas con los equipos de Telecomunicaciones, no resultara como lo esperado en el primer intento, fue mediante un proceso iterativo que se logró dicha comunicación de manera

* El dato se encuentra de manera implícita, ya que al realizar una lectura de la trama completa, se contemplan también los bits de espaciado entre bytes de datos.

Ψ Refiérase al Apéndice D.

satisfactoria. Es entonces que la solución digital, mediante el microcontrolador, pasa a ser un módulo final, de la solución modular completa.

Siguiendo con la solución modular planteada, se procedió a solucionar la etapa de multiplexación de la comunicación entre la computadora y los equipos de Telecomunicaciones, como se expone seguidamente.

En ésta etapa se plantean 3 posibles soluciones, las cuales se pasan por un proceso de discriminación, para implementar la más factible.

Como primera solución, se plantea solucionar mediante integrados con función de multiplexor/demultiplexor analógicos, controlados digitalmente, seleccionando los integrados CD74HC4067(hojas de datos se muestran en el Anexo A), los cuales realizan la tarea necesaria, pero se encontró un inconveniente económico, y es que el precio de cada integrado oscila entre \$15 a \$20, y como el medio de comunicación serial requiere de 9 unidades, la implementación pasaría a ser una solución muy costosa, alrededor de \$150, sin tomar el cuenta el hardware extra, por necesitar mayor cantidad en las líneas de control (4 por Integrado).

La segunda opción de solución planteada fue mediante un microcontrolador, el cual es el responsable directo de realizar la multiplexación de la señal, pero ésta no satisface los resultados esperados, ya que la función de multiplexación/demultiplexación el microcontrolador la realiza de manera sincrónica, además de que por su arquitectura interna únicamente puede realizar una única instrucción por ciclo de instrucción, y como mínimo se deben de realizar 9, por lo que el periodo de refrescamiento de una señal es como mínimo 9 ciclos de instrucción, equivalente a 1,8 μ s, como se muestra en la figura 5.1

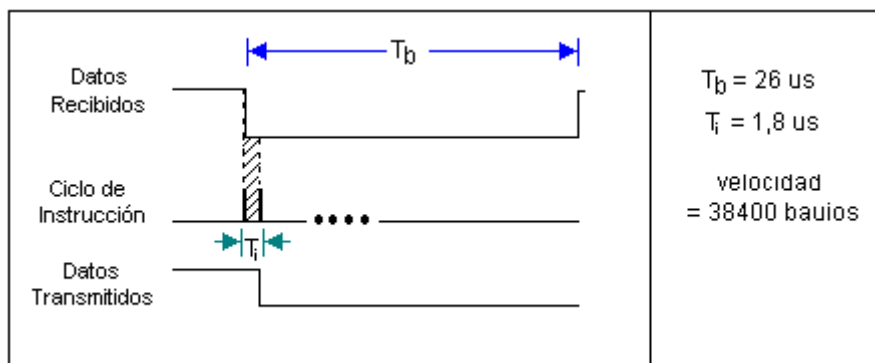


Figura 5.1. Diagrama de tiempos para multiplexión digital

Como se observa en la figura 5.1, el tiempo de inactividad (T_i), viene a ser significativo, sobre todo por que el estándar RS232, especifica que la transición entre un '1' lógico a un '0', o viceversa, no debe sobrepasar el 4% del Periodo de un bit(T_b), por lo que dicho valor debe ser:

$$T_{4\%} \leq T * 0,04$$

$$\text{Con} \Rightarrow T = 26\mu s$$

$$T_{4\%} \leq 1,04\mu s$$

En la figura 5.1, se muestra que este tiempo tiene un valor mínimo de $1,8\mu s$, que es superior al máximo establecido por el estándar de $1,04\mu s$, por lo tanto la solución digital planteada, no se puede implementar por incumplimientos con el estándar RS232.

Con base en esta conclusión, se debe buscar una solución combinacional, por lo que la implementación de una GAL, viene a ser una la solución más óptima para la multiplexación. La última solución planteada e implementada, es la solución con una GAL22V10D, la cual da la posibilidad de multiplexar la señal, sin tener problemas de retardo de la señal.

Como último paso, y con la unificación de los módulos de solución propuestos, se implementaron mediante un software de control, dando como resultado el cumplimiento satisfactorio de los objetivos planteados del proyecto.

5.2 Descripción del Hardware

5.2.a Fabricación de un cable directo con terminales de monitoreo:

Dicho cable posee dos terminales extra, las cuales sirven para monitorear el trasiego de información entre la computadora terminal y los equipos de telecomunicaciones. Como puede observarse en la figura 5.2, los datos enviados de la computadora terminal a los equipos de telecomunicaciones, se leen mediante la computadora espía a través el puerto 1, y los datos enviados de los equipos a la computadora, se leen a través del puerto 2 de la computadora espía.

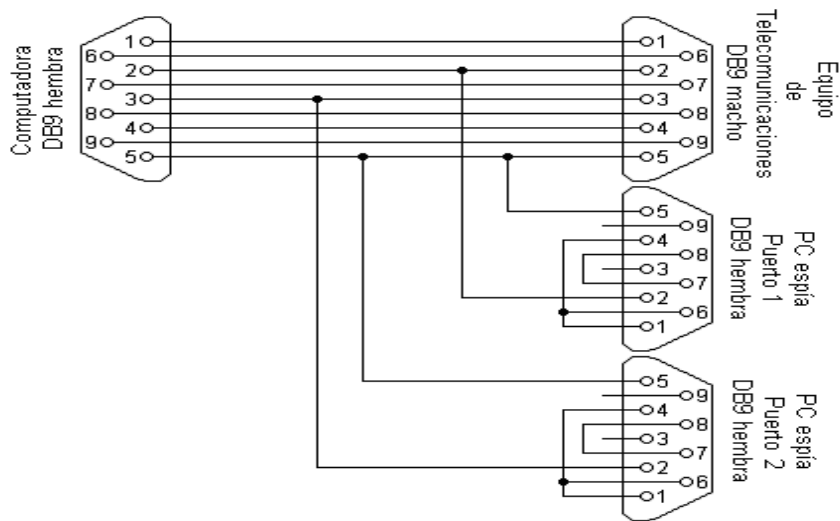


Figura 5.2. Diagrama de conexión del cable de monitoreo

Los datos obtenidos se muestran en el capítulo 6, pero al intentar reproducirlos para recibir una respuesta por parte de los equipos de Telecomunicaciones, no se tuvo éxito, debido a la pérdida de bits significativos, que mediante la lectura por software no se podían observar.

Computadora	Equipo de Telecom.	Puerto 1 Espía	Puerto 2 Espía	Descripción
1	1	-	-	Carrier Detect
2	2	2	-	Rx » Rx ₁
3	3	-	2	Tx » Rx ₂
4	4	-	-	Data Terminal ready
5	5	5	5	Ground
6	6	-	-	Data set ready
7	7	-	-	Request to send
8	8	-	-	Clear to send
9	9	-	-	Ring indicator

Tabla 5.1. Diagrama de configuración de pines del cable de monitoreo

Como puede observarse en la tabla 5.1, los datos enviados de la computadora al equipo, se leen a través de la señal Rx1, o sea el pin receptor del puerto serie del puerto 1, y en el caso de que la información fluya del equipo de telecomunicación hacia la computadora, la información se lee a través de la señal Rx2, que es el pin receptor de datos del puerto 2.

Este tipo de medio de conexión con configuración directa se utiliza por los equipos Alcatel, NEC y Ericsson.

5.2.b Hardware espiador

Como se mencionó en la sección anterior, se diseñó una opción más rápida, y más eficiente, para leer las tramas de comunicación, la cual consta de una solución digital, mediante un microcontrolador y un MAX233, cuya configuración se puede observar en la figura 5.2, en la que se muestra claramente que mediante el microcontrolador, se monitorea la trama que va en dirección desde la computadora, hacia el equipo de telecomunicaciones. Y de esta forma se obtuvo el valor de dichas tramas (los resultados se muestran en el siguiente capítulo), las cuales se corroboran con las obtenidas con el osciloscopio. Por lo que esta solución, pasa a ser parte integral de la implementación final.

A pesar de que la lectura inteligente de esta trama no estaba dentro de los objetivos planteados, se creyó conveniente hacer el dispositivo lo más automatizado e inteligente posible, para que pueda tener una continuación de proyecto sin ningún problema de compatibilidad con otros equipos.

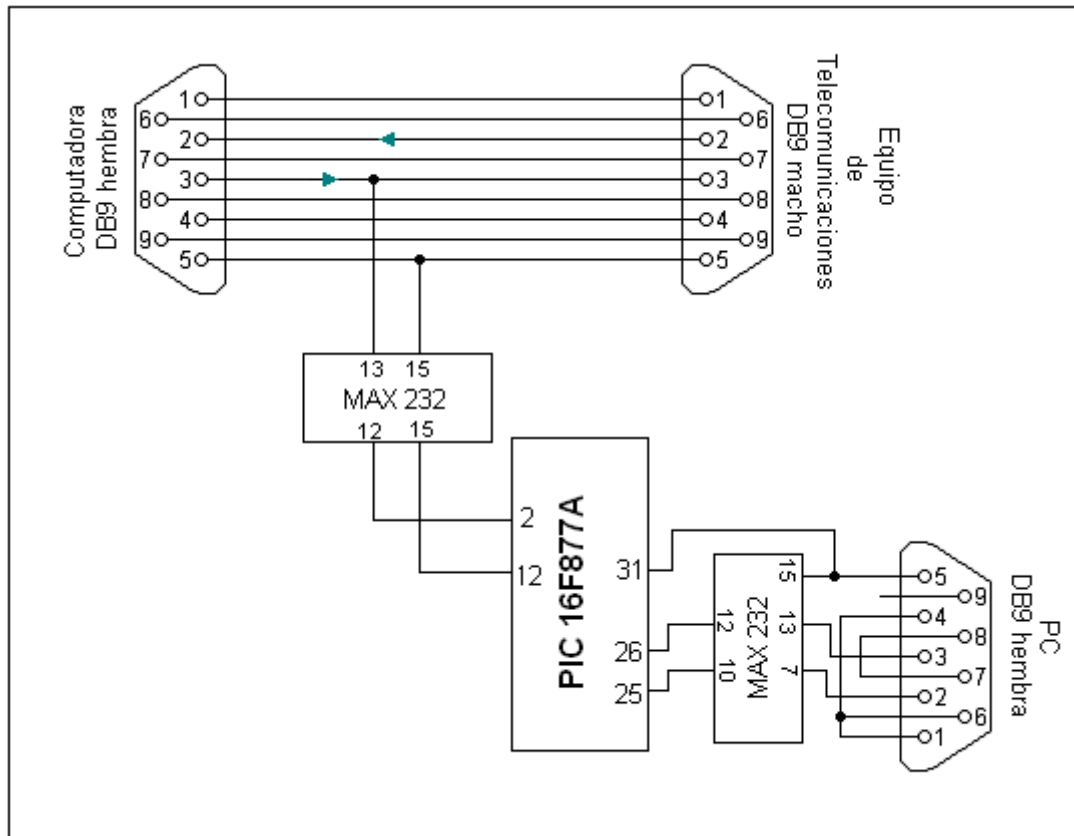


Figura 5.3. Diagrama de solución digital para el monitoreo de comunicación

En la figura 5.3, se muestra el microcontrolador *16F877A, que es el responsable de la lectura de la trama de datos, a través de rutinas de programación, en las que se cumplió los estándares de software del RS232. Además se muestra el *MAX233, con el cual cumple con estándares eléctricos del puerto serie, que es a su vez la etapa de interfase, para convertir los niveles bajos de voltaje y los niveles alternos, entre el microcontrolador y las señales alternas del RS232.

* Sus Hojas de datos se muestran en el Anexo D
 • Sus Hojas de datos se muestran en el Anexo C

5.2.c Multiplexión de la señal

Para realizar la multiplexación de la señal se utilizó un dispositivo lógico programable (*GAL22V10D), especialmente por sus características combinatoriales, ya que se espera que el tiempo de retardo por la multiplexación de las señales tienda a cero.

A través de este dispositivo, se logró establecer la comunicación; en él que además de multiplexar la señal con la computadora, también multiplexa las señales del PIC, utilizadas para el reconocimiento de los equipos, como se observa en la figura 5.3.

* Sus Hojas de datos se muestran en el Anexo B.

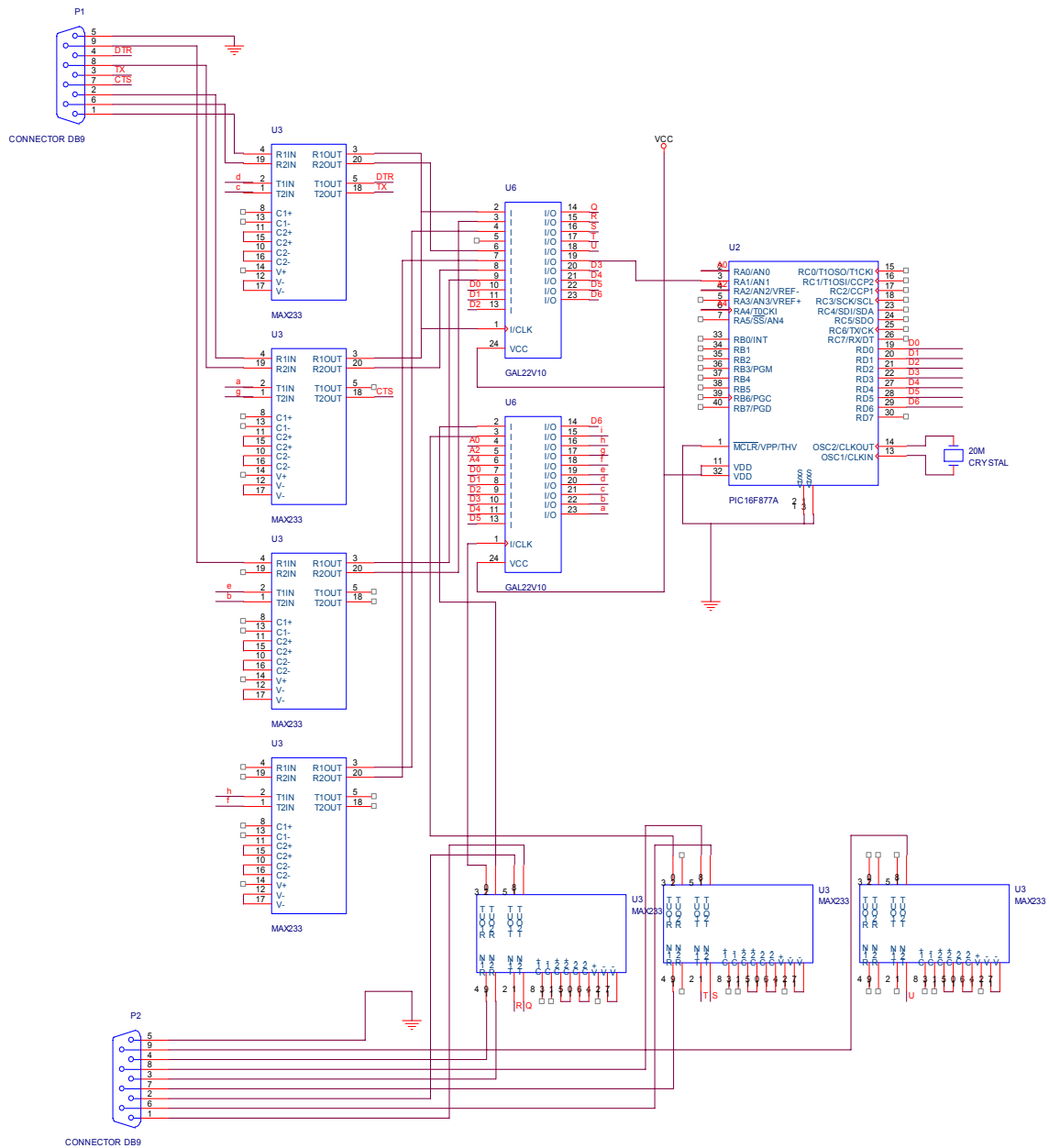


Figura 5.4. Diagrama de multiplexión de las señales

A través de las líneas D1-D7, el PIC indica la multiplexación de las señales, en el que la menos significativa selecciona si la comunicación se realiza con la computadora, o con el microcontrolador. Y a través de las otras líneas realiza el protocolo correspondiente.

Con las pruebas realizadas, se determinó que los integrados utilizados (MAX233), producen un problema de impedancias cuando se colocan dos transmisores de digital a analógico en paralelo, como se observa en la figura 5.4. Y el dato no lee correctamente, por lo que la comunicación no se puede realizar en forma bidireccional, de esta manera.

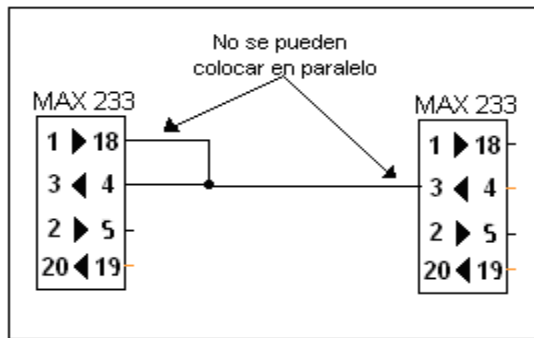


Figura 5.5. Diagrama de comunicación bidireccional

En conclusión, la forma de realizar la comunicación bidireccional, es mediante un sistema mecánico, como un *relay*, como se observa en la figura 5.5.

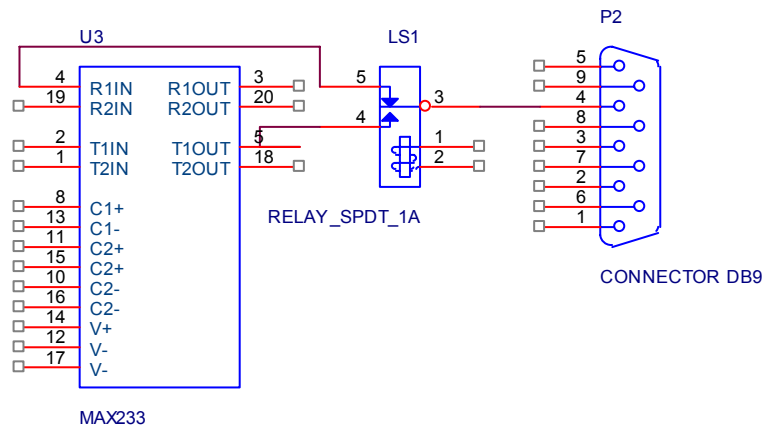


Figura 5.6. Diagrama de comunicación bidireccional con *Relay*.

A pesar de que no fue necesaria la implementación de la solución con *Relay*'s en el Apéndice J, se muestra el diagrama final del circuito, con el uso de estos componentes.

5.2.d Hardware de control implementado

El diagrama general implementado, para la realización de las distintas pruebas de comunicación con los diferentes equipos, se muestra en la figura 5.6

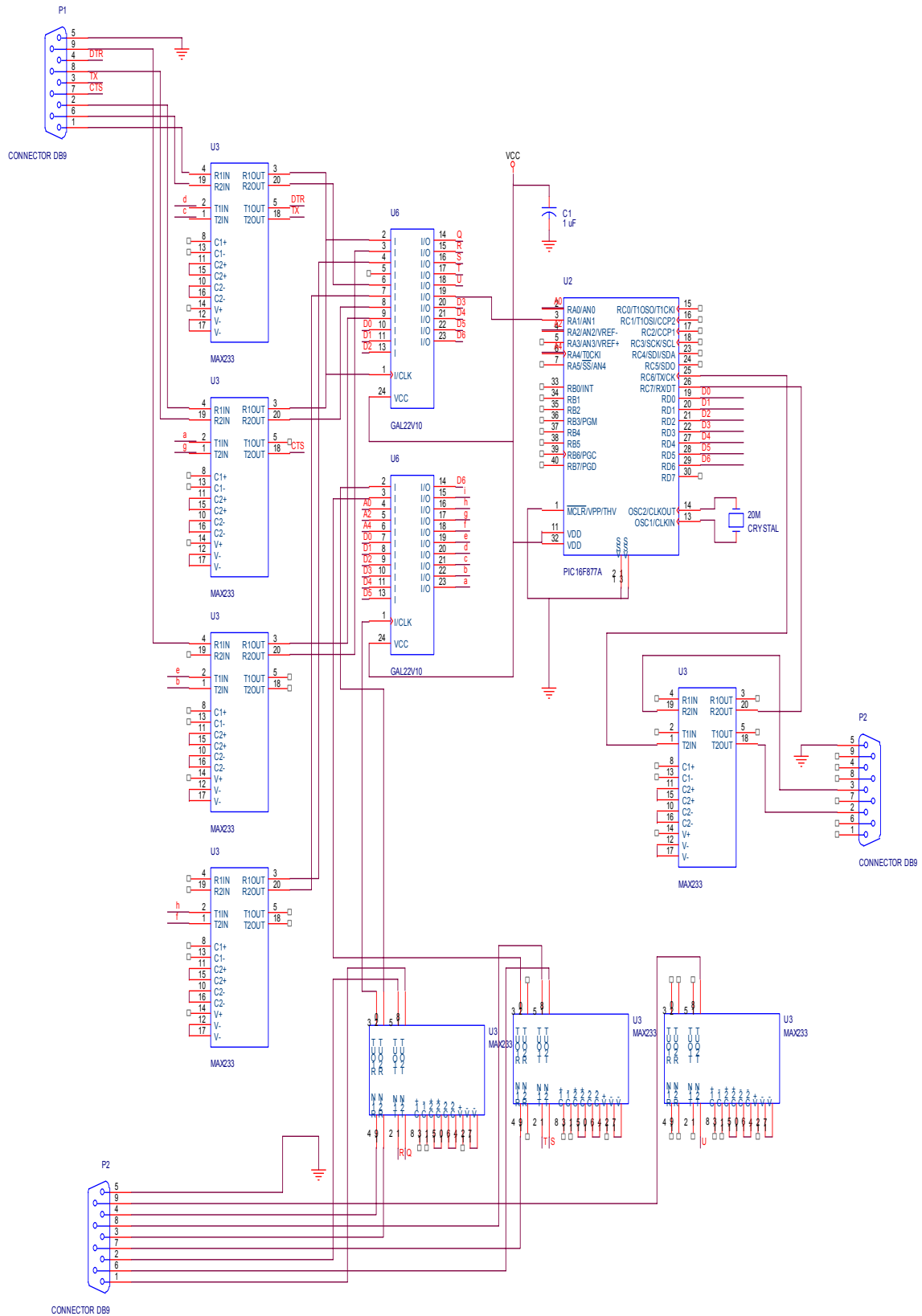


Figura 5.7. Diagrama final implementado

5.3 Descripción del Software

5.3.a Software espiador de comunicación

Se diseñó un software de Monitoreo, que despliega la información recibida en forma sincrónica, además se muestran algunas señales de control de flujo, para un mejor análisis y monitoreo de las señales en la comunicación entre la computadora y los equipos, como se observa en la Figura 5.3, En los cuadros superiores se leen los datos tanto de envío, como los de respuesta (mediante la lectura de los dos puertos). Por último en el cuadro inferior se muestra el comportamiento de las señales, a través del tiempo en forma discreta.

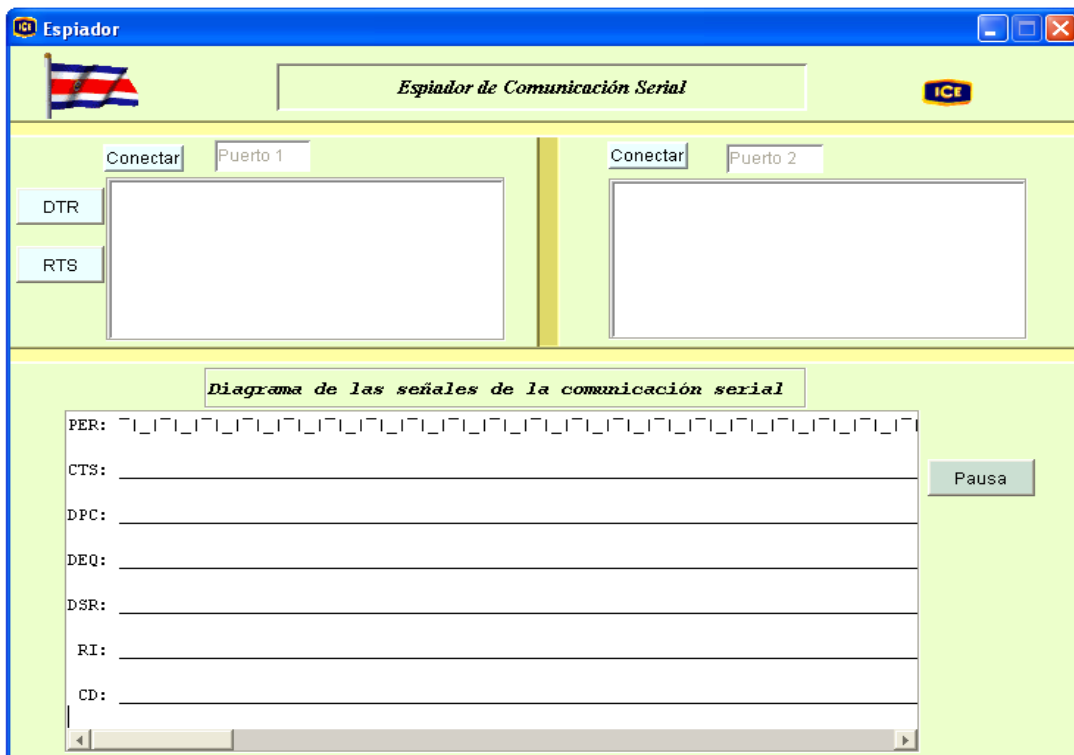


Figura 5.8. Software de monitoreo de comunicación

Éste software, también posee la particularidad de enviar datos a los equipos, para poder recibir respuesta por parte de los mismos, pero en dicha prueba no se obtuvieron los resultados esperados, por lo que se procedió a realizar la medición y transmisión de datos, de otras maneras, explicadas previamente.

5.3.b Software de transmisión de una trama de 6 bytes

En el apéndice A, específicamente la figura A, se muestra la interfaz del software de comunicación, la cual tiene por objeto realizar mediciones de la trama de datos, con diferentes combinaciones de configuración.

Este software posee dos métodos muy importantes, generados a través de los eventos del botón de Propiedades, y del botón de Enviar.

EL de propiedades tiene como finalidad, realizar una configuración del puerto en ejecución, para poder realizar cambios de configuración de forma dinámica. Y en el de enviar, se busca enviar una trama de 6 bytes, con la configuración seleccionada, a través del botón de propiedades. En éste lo que se realiza, a grandes rasgos, es enviar una variable de arreglo tipo byte, mediante un método de envío de la clase *outputBuffer* de JAVA.

Este software fue muy importante para el desarrollo mismo del proyecto, ya que éste envía los datos con una particularidad, y es que mediante la interfaz lo que se escribe es el código ASCII, en lugar del carácter, facilitando de esta manera el envío de caracteres no imprimibles a través del teclado.

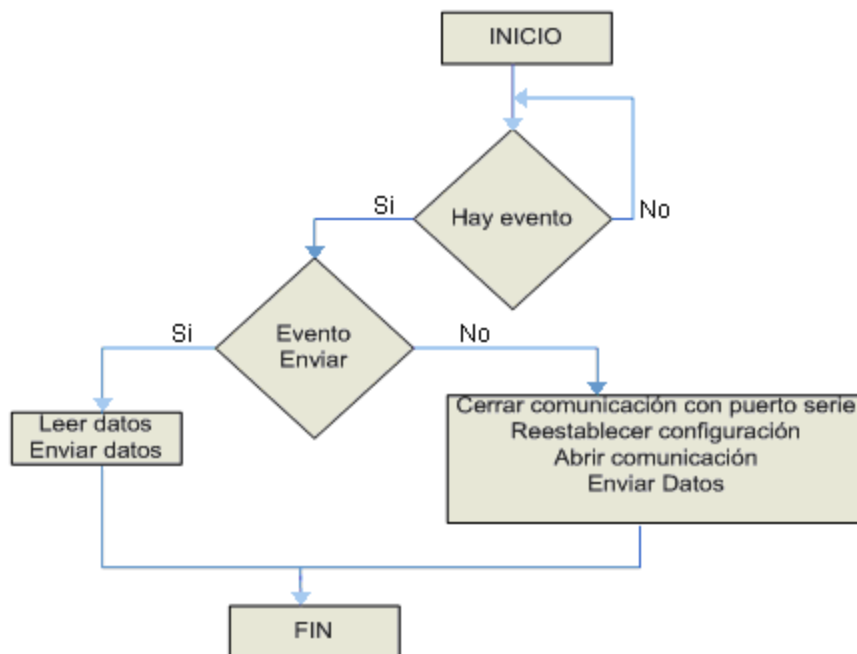


Figura 5.9. Diagrama de flujo para los métodos de configuración, y envío de tramas.

5.3.c Software del Microcontrolador

El software del microcontrolador cuenta con características muy particulares, como lo es su diseño automatizado y generalizado.

La figura 5.4 muestra el diagrama de flujo de la secuencia de operación, del hardware de control, en la que existen 4 estados principales de operación; que son: Automático, Manual, leer Trama y leer Trama con dato.

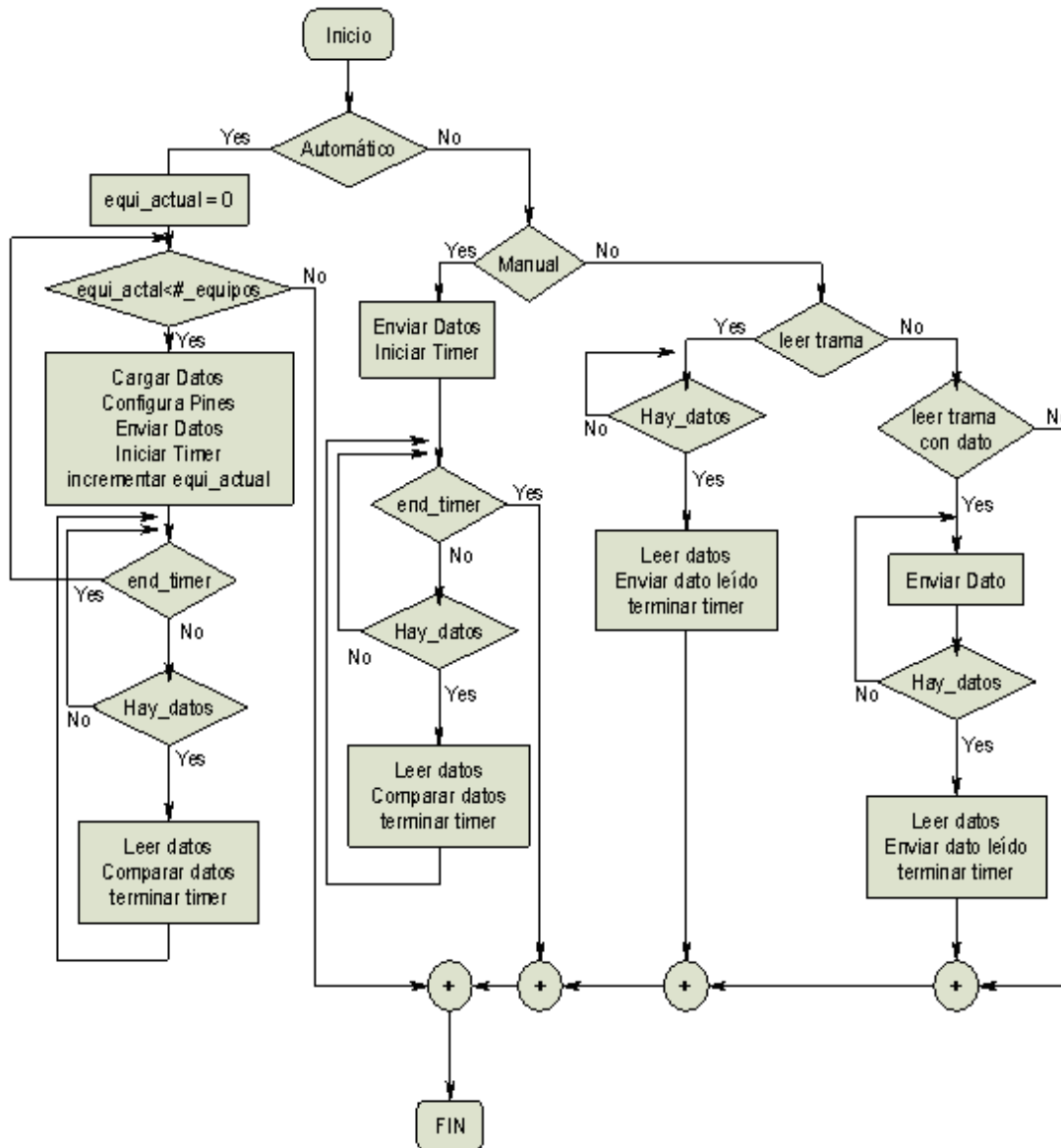


Figura 5.10. Diagrama de Flujo del software del microcontrolador

El modo “Automático”, lo que representa es que el modo de configuración y reconocimiento de los equipos, se debe realizar en forma automática, o sea, el dispositivo debe reconocer el equipo conectado, realizando la configuración correspondiente para enviar la trama de *hand-shaking*, y por último recibir el dato de respuesta del equipo, para compararlo con el valor esperado. Para los casos en que no se reciba dato de respuesta por parte del equipo, se configura con el siguiente, que fue previamente aprendido, y así sucesivamente, hasta llegar al último, luego se retorna al primero, volviendo el ciclo en una lista circular. Es importante señalar que se utilizó un *Timer*, que tiene la función de dar un tiempo prudencial, para la espera del dato de respuesta por parte del equipo, con un valor de 1,56 [s].

El modo “Manual”, se configura para un único equipo, enviándole el dato correspondiente, y de igual manera que en el modo Automático, se espera un Tiempo prudencial (definido por el *Timer*), para leer el dato de respuesta por parte del equipo, en cuyo caso se compara con el valor esperado, para comprobar si es el equipo a reconocer.

El método de “Leer trama”, espera a que haya un dato para leerlo, y enviarlo al software de alto nivel. Este método tiene gran importancia, en el proceso de automatización del dispositivo, y es que mediante esta rutina, se logra conocer el dato de *hand-shaking*, de envío de la computadora hacia el equipo a conectar.

Por último el método de “Leer trama con dato”, busca conocer el dato de respuesta del Equipo, enviándole el dato de saludo (*hand-shaking*), conocido por el método anterior de “Leer trama”.

Es importante señalar, que con la implementación de estos dos últimos métodos, se logró confirmar las tramas de datos de los equipos, medidas con el Osciloscopio, así como conocer la trama de respuesta del Equipo Lucent, ya que no se había podido realizar su lectura, por razones de velocidad de transmisión, y tamaño de la trama.

5.3.d Comunicación del Microcontrolador con el programa en alto nivel

Esta comunicación, a nivel de software se diseñó mediante estados, que representan instrucciones que realiza el microcontrolador.

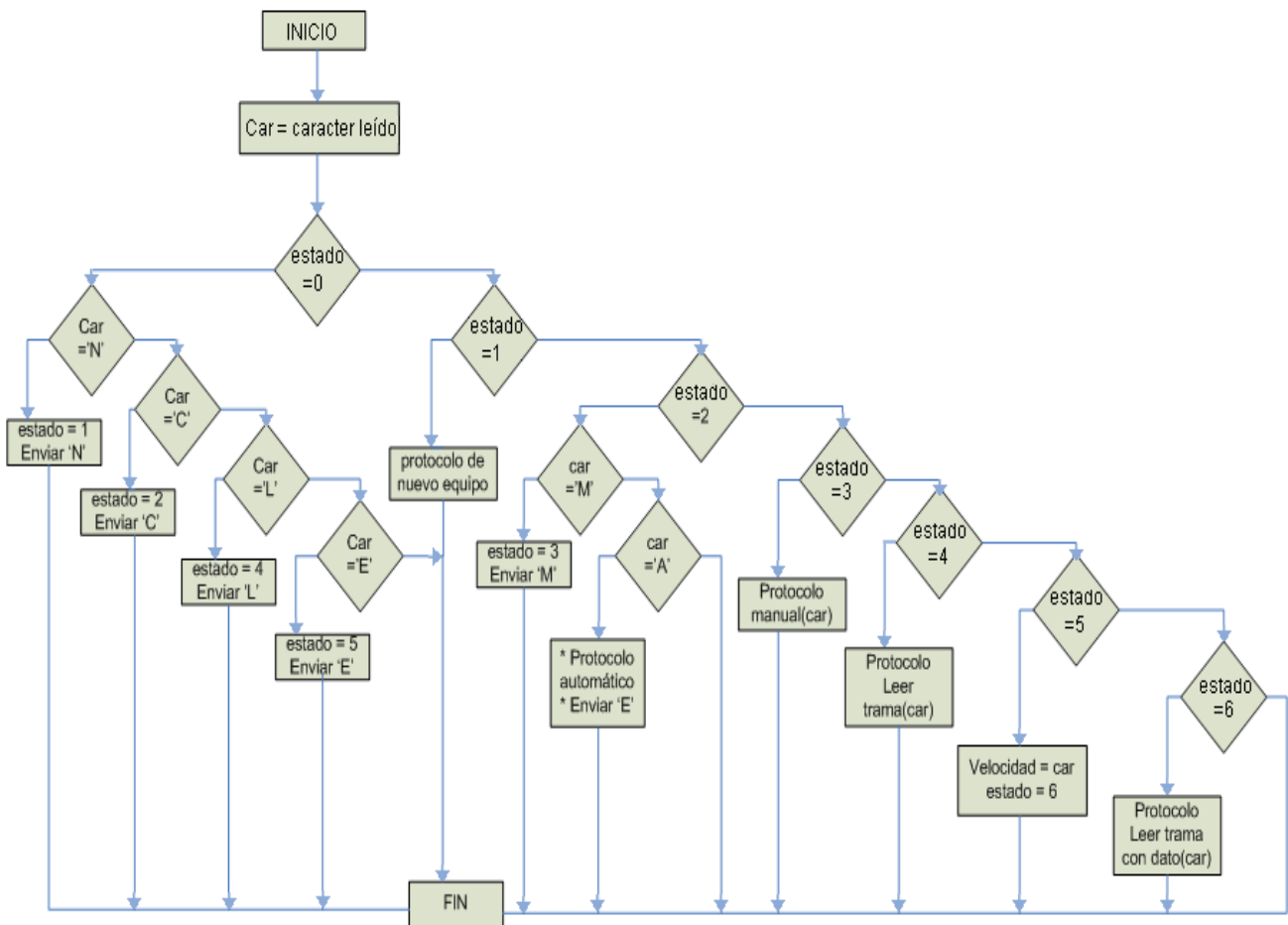


Figura 5.11. Diagrama de Flujo de la comunicación entre la computadora y el PIC

El estado 0, representa el estado inicial, en el que se selecciona el modo de operación del PIC, como se observa en la figura 5.5.

El PIC opera en cuatro estados fundamentales, que se seleccionan cuando se envían los caracteres, al puerto serie del PIC.

El estado 1, se selecciona cuando se recibió previamente en el puerto serie una 'N', durante el estado 0, éste realiza una llamada a la rutina del método de agregar nuevo equipo, que se explica con más detalle en la siguiente sección.

El estado 2, se selecciona en el estado 0, cuando se recibe una 'C', en él se escoge el modo de configuración, ya sea Manual enviando una 'M', ó Automático cuando se envía una 'A', del software de alto nivel. Para el caso del modo de configuración automático, se hace una llamada a la rutina de

configuración, explicada en la sección anterior. En el caso que se recibe una 'M', se cambia al estado 3.

En éste se configura el dispositivo en modo manual, recibiendo como parámetro el código del equipo que se desea configurar.

Para que el dispositivo pase al estado 4, se debe enviar una 'L' en el estado inicial (estado 0); en el que a grandes rasgos, la función que realiza es la lectura de la trama de envío al equipo, necesaria para aprender los datos de configuración de un equipo que se necesite agregar. Por lo que esta trama es también utilizada para obtener el dato de respuesta del equipo, del *hand-shaking*.

El estado 5, se selecciona con una 'E' en el estado inicial. En éste se recibe la velocidad de comunicación con el equipo, pasando así al estado 6, para recibir los datos a enviar al equipo en su protocolo de *hand-shaking*.

Éstos se envían al equipo, para recibir una respuesta, y tener de ésta manera los datos de configuración (*hand-shaking*), del equipo que se desea agregar.

5.3.e Método de agregar nuevo equipo

En esta rutina se reciben todos los parámetros del equipo que se desea agregar, en el orden en que se muestra en la figura 5.6

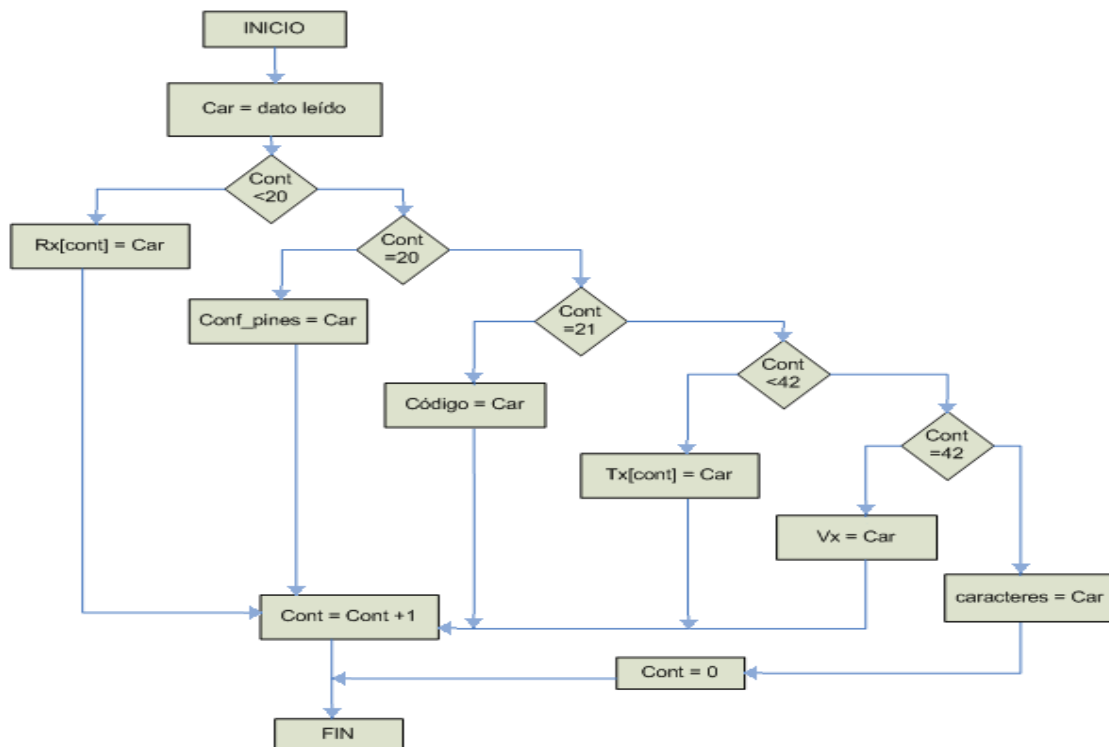


Figura 5.12. Diagrama de Flujo de la rutina de agregar nuevo equipo

5.3.f Almacenamiento de los datos en la EEPROM del microcontrolador

Para guardar los datos de configuración de los equipos, en la memoria del microcontrolador, se procedió a almacenar en la EEPROM, que tiene un espacio de almacenamiento de 256 bytes.

Los datos de configuración de un equipo nuevo, se mostraron anteriormente en la figura 5.6, en ella se observa que la cantidad de bytes correspondientes para cada equipo es de 44 bytes, por lo que se puede almacenar la configuración de 5 equipos, que es suficiente para cumplir con los objetivos de reconocimiento de 4 configuraciones de equipos.

Con base en estos parámetros, se diseñó un mapeo de la memoria de datos principal (EEPROM), como se muestra en la figura B.1, del Apéndice B, con bloques de memoria por equipo de 44 bytes.

A partir del mapeo de memoria, se facilita el diseño de rutinas tanto de almacenamiento como de lectura de la memoria.

5.3.g Método de almacenamiento en la memoria EEPROM

Esta rutina constituye en la adición de nuevos equipos, que tienen un espacio en memoria de 44 bytes, y se deben guardar según el orden del mapeo de memoria diseñado, mostrado en la figura B.1, del Apéndice B.

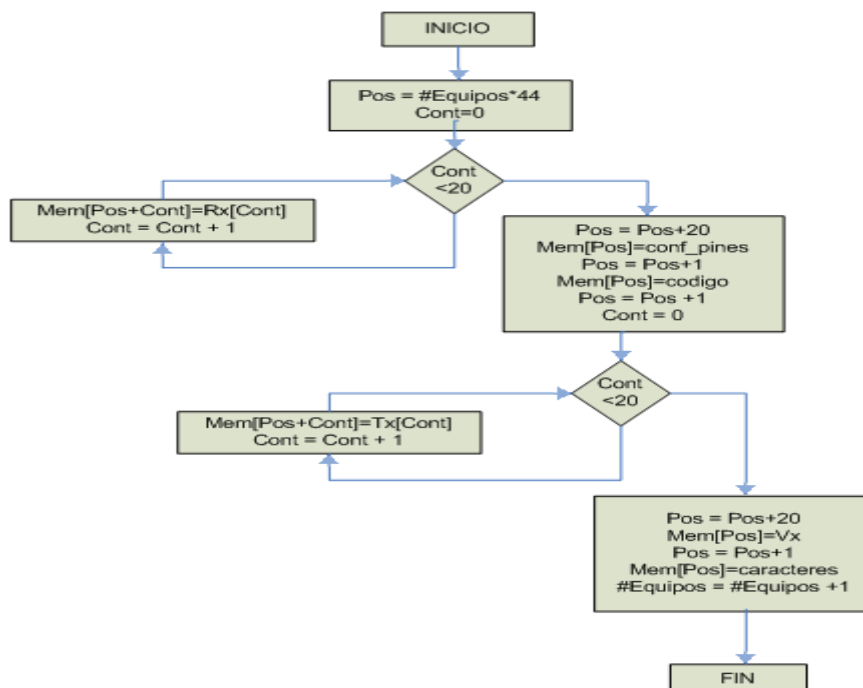


Figura 5.13. Diagrama de Flujo del método de almacenamiento en memoria

5.3.h Método de lectura datos de la EEPROM

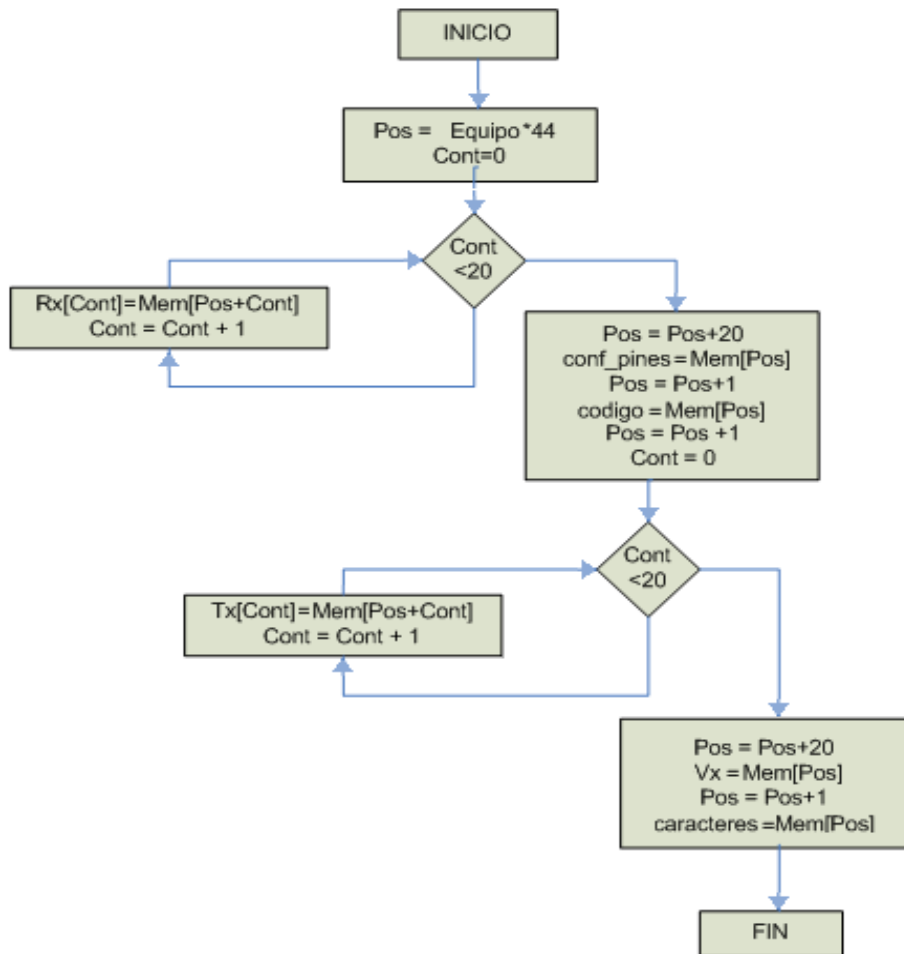


Figura 5.14. Diagrama de Flujo de la lectura de datos de la EEPROM

El método de lectura de datos de la EEPROM, lee los datos de configuración directamente de la memoria, almacenándolos en variables, utilizadas en otros métodos para el reconocimiento de los equipos.

Los parámetros leídos de memoria, son los de configuración del equipo, por lo que este método recibe como parámetro el número de equipo a cargar.

5.3.i Rutina de envío de datos a los equipos

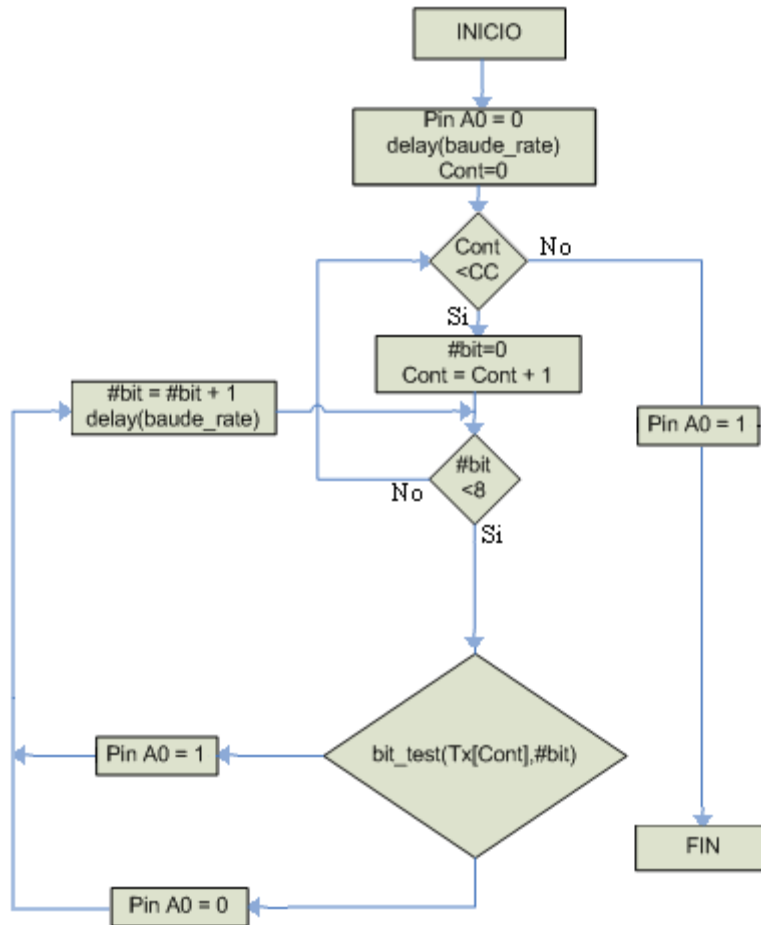


Figura 5.15. Diagrama del método de envío de datos

En el diagrama de flujo de la figura 5.9, se muestra el método de envío de la trama de datos, en la que es importante resaltar que su longitud es variable, respecto del equipo (en la figura 5.9 se representa con la variable CC). También es importante destacar que en la rutina se contemplan características del estándar RS232, como lo es el bit de inicio, y el bit de parada.

5.3.j Rutina de lectura de datos

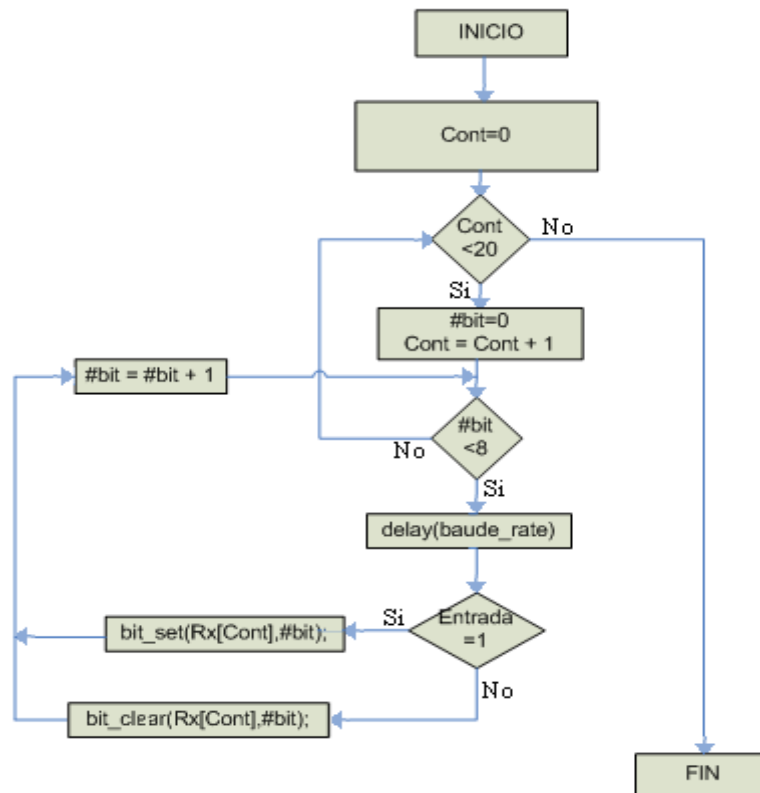


Figura 5.16. Diagrama de Flujo del método de lectura de datos

En la figura 5.10, se muestra el método de lectura de la trama, diseñada para leer los datos en forma serial. A pesar de que el microcontrolador posee ésta característica, se tienen varios inconvenientes, para realizar la lectura mediante la USART del microcontrolador, como lo es la velocidad variable, así como la necesidad de dos módulos de comunicación serial; uno que se comunica con la computadora, específicamente con el software de alto nivel (software de control), y otro módulo que se comunica con los equipos de Telecomunicaciones, para su respectivo reconocimiento.

El diagrama de flujo de la figura 5.10, realiza la lógica de la lectura de tramas de 20 bytes de datos, cuyo resultado se almacena en un arreglo de 20 bytes, llamado Rx.

Cabe destacar que con la implementación de éste método, se tiene la ventaja de poder cambiar la velocidad de lectura de los datos, únicamente con el cambio en la variable `baude_rate`, que es el periodo de espera de la lectura de los bits.

5.3.k Rutina de comparación de datos recibidos con los esperados

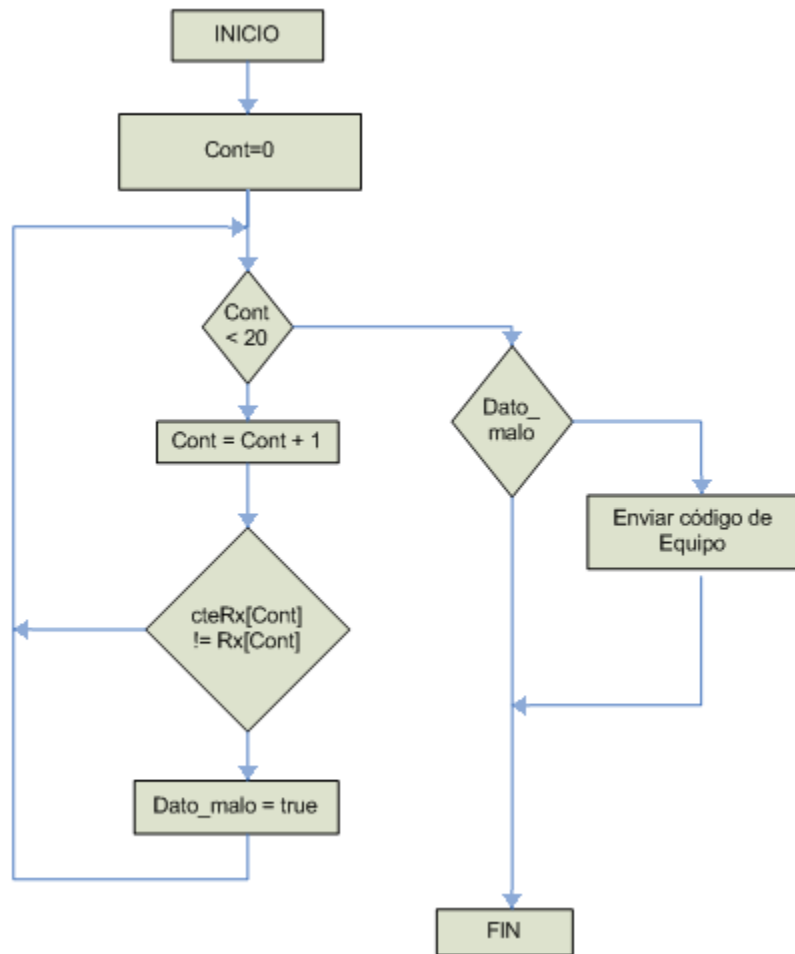


Figura 5.17. Diagrama de Flujo del método de comparación de datos

El método, ilustrado en el diagrama de flujo de la figura 5.11, se utiliza para comparar la trama de datos leída con la que se espera recibir, cuyo valor fue leído en el método de la figura 5.8, y él de la trama de respuesta se leyó en el método de la figura 5.10.

Por ende, con los datos previamente cargados, se procede a comparar las tramas en dicha rutina, y en el caso en que las tramas sean iguales, se envía el código del equipo, al software de alto nivel, para una posterior consulta en la base de datos.

5.3.1 Software del programa final en alto nivel (Java)

En ésta sección, se muestra el programa en alto nivel en Java, que es la responsable directa, de una interacción con el usuario, que consta de cuatro funciones principales.

5.3.1.a Configuración del Dispositivo

La configuración del dispositivo, es una de las funciones del software, que tiene por objeto, seleccionar el modo de configuración del dispositivo, como se observa en la figura 5.12. En ésta se puede seleccionar el equipo a configurar directamente, seleccionando el tipo de configuración Manual, donde el software inicia un protocolo de comunicación con el hardware, transmitiendo así los parámetros necesarios para la configuración.

También como se observa en la figura 5.12, existe el tipo de configuración Automática, que inicia otro protocolo de comunicación

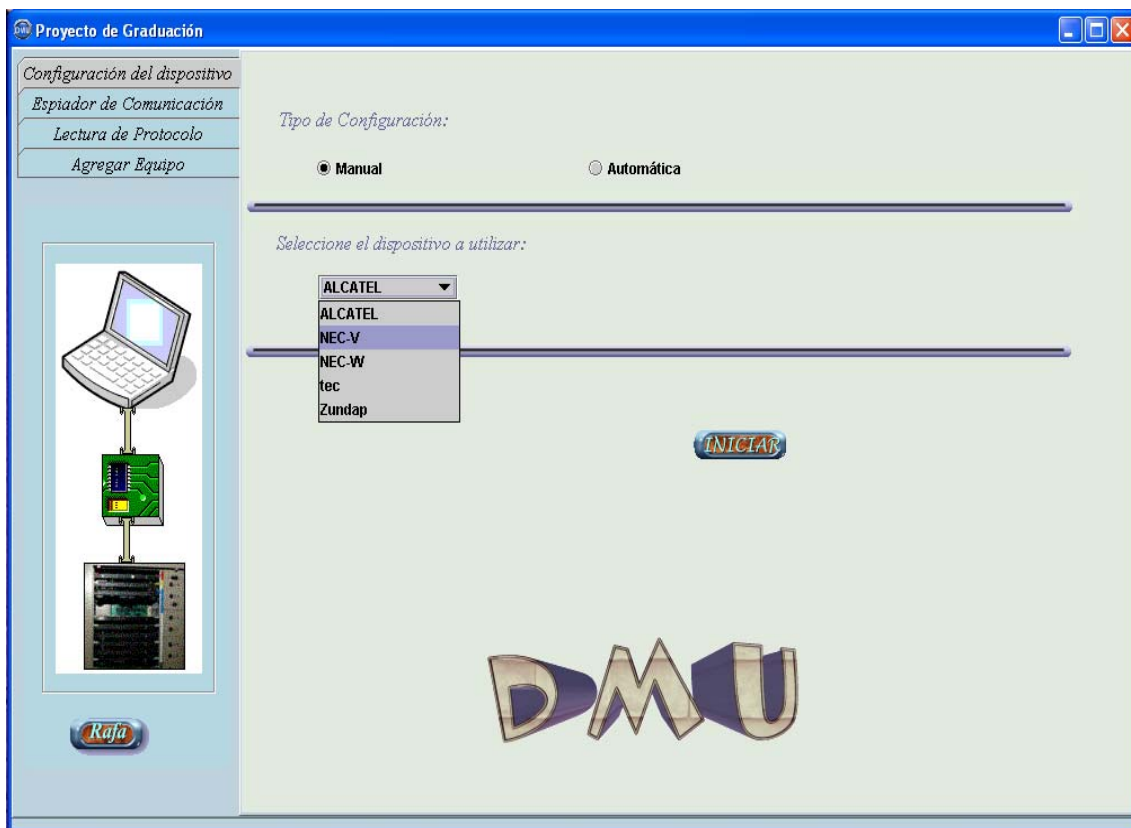


Figura 5.18. Interfaz de configuración del dispositivo

5.3.1.b Espiador de Comunicación

Este subprograma, espía la comunicación entre la computadora y el equipo, mediante el cable espíador, a través del cual se monitorea la comunicación en forma bidireccional.

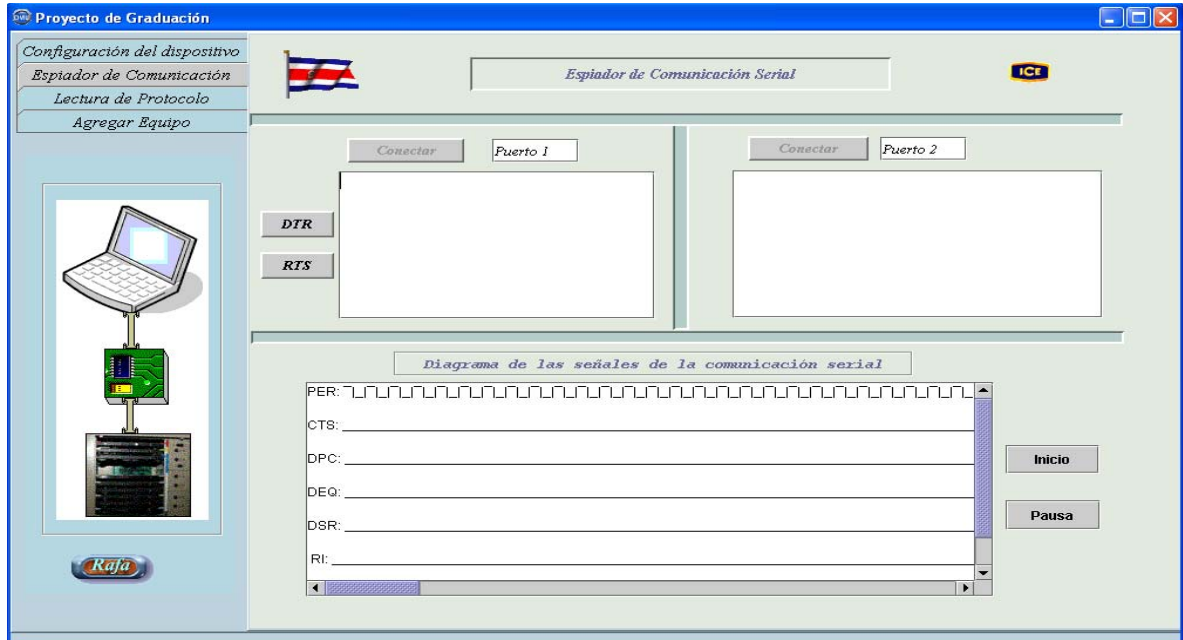


Figura 5.19. Interfaz espía de comunicación

5.3.1.c Lectura de Tramas de Configuración

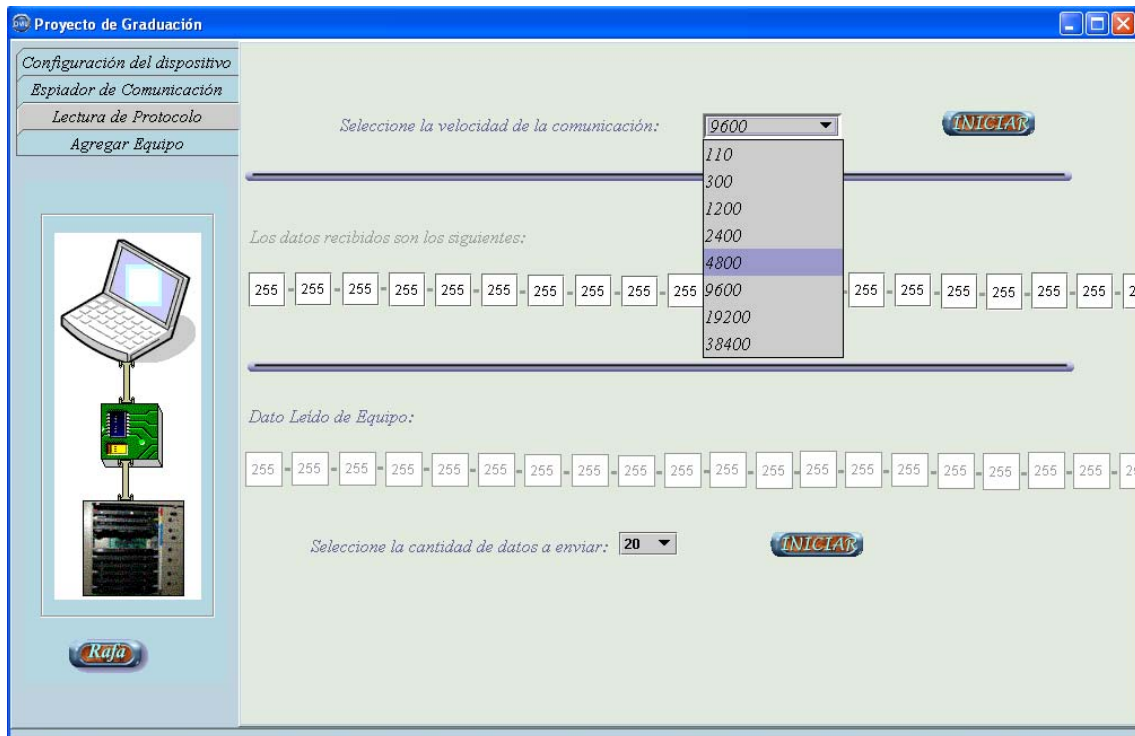


Figura 5.20. Interfaz de lectura de las tramas de datos

Con esta interfaz, se determinan tanto los datos de envío, como los de respuesta de los equipos, a la velocidad seleccionada; en la primera fila de la figura 5.14, se despliega la trama recibida de la computadora, que corresponde a los datos de envío hacia él equipo en cuestión.

En la segunda fila de datos de la figura 5.14, se muestran los datos recibidos por parte del equipo.

5.3.1.d Agregar un equipo nuevo

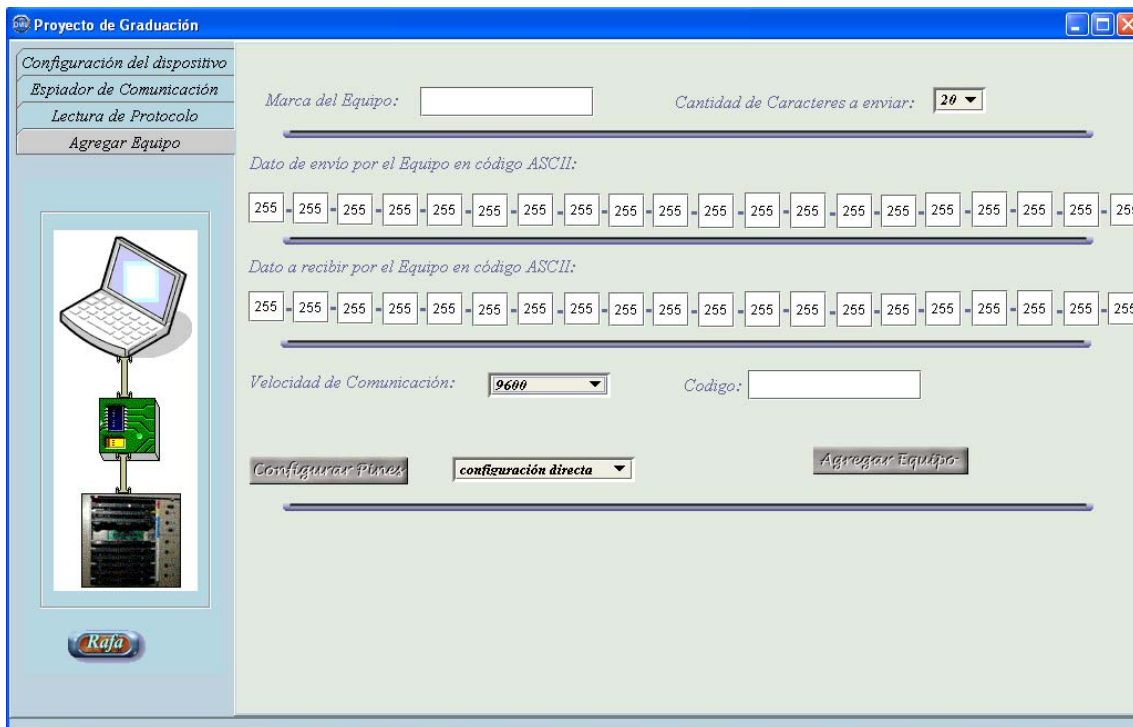


Figura 5.21. Interfaz para agregar un nuevo equipo

En esta última interfaz, el software realiza la función de agregar un nuevo equipo, para el cual ya se conocen previamente sus características de configuración, como se muestra en la figura 5.15, y a través del botón derecho de Agregar Equipo, se inicia un protocolo de comunicación con el dispositivo, enviándole los parámetros del equipo nuevo, y de esta manera agregarlo a la memoria EEPROM del microcontrolador.

5.3.1.e Protocolo de comunicación diseñado

Debido a que el dispositivo (hardware), debe de realizar varias funciones, se diseñó un protocolo de comunicación, para que el software del microcontrolador pueda realizar distintas funciones y recibir los datos ordenadamente.

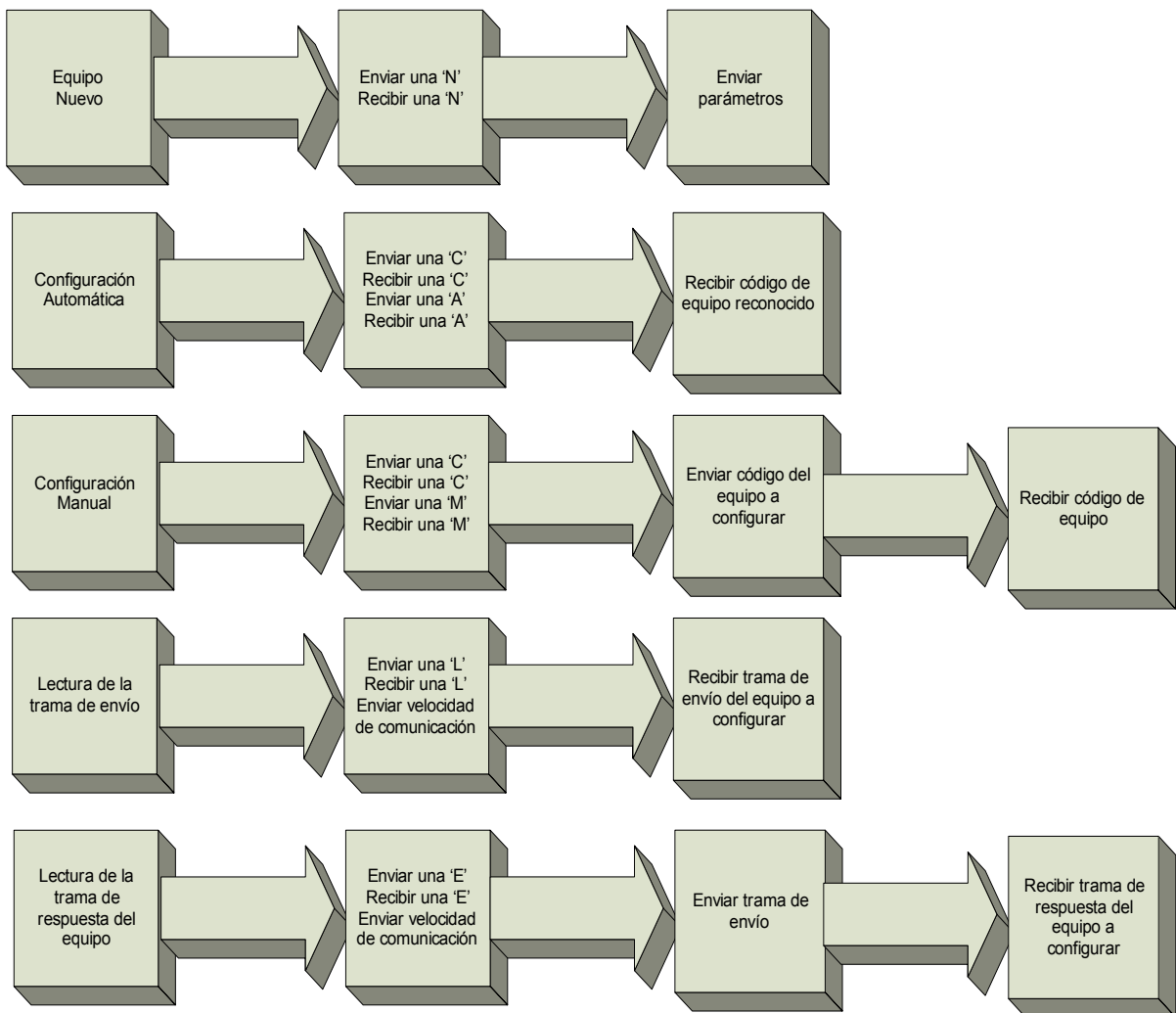


Figura 5.22. Diagrama de bloques del protocolo de comunicación con el PIC

En el diagrama de bloques de la figura 5.16, se muestra el protocolo de comunicación diseñado, con el que se logra una comunicación ordenada, entre el hardware y el software de control.

5.3.1.f Diseño de la Base de Datos

El diseño de la base de datos, se realizó con el modelo entidad-relación, cuyo diagrama se muestra en el Apéndice C, en el que se muestra las entidades, y su relación. Como metodología de diseño, se siguieron los siguientes pasos.

Identificación de entidades

Las entidades identificadas, son:

- Equipos
- Datos
- Pines

Identificación de Relaciones

Las relaciones encontradas, para las entidades identificadas son:

- Datos \Rightarrow Equipos: la relación es 1 a n, ya que un dato podría pertenecer a más de un equipo.
- Equipos \Rightarrow Datos: la relación es 1 a 1, ya que un equipo únicamente puede contener 1 dato.
- Pines \Rightarrow Equipos: la relación es 1 a n, ya que una configuración de pines podría pertenecer a más de un equipo.
- Equipos \Rightarrow Pines: la relación es 1 a 1, ya que un equipo únicamente puede contener 1 una única configuración de pines.

Identificación de atributos

Equipos:

- Marca: identifica el nombre o la marca del equipo.
- Código: Representa un código representativo único para cada equipo.
- Velocidad: almacena la velocidad de comunicación de cada equipo.
- Carácter de envío: identifica la cantidad de caracteres de envío.

Datos:

- 20 caracteres: representa los datos de saludo inicial
- Dirección: identifica si los caracteres son de envío o de respuesta.

Pines:

- Configuración: representa la configuración de pines

Capítulo 6: Análisis de Resultados

6.1 Mediciones realizadas con software de monitoreo

Velocidad (baudios)	Trama de la PC al equipo		Trama del equipo a la PC	
	Caracteres	Código ASCII	Caracteres	Código ASCII
1200	ä, x, ☉, ` , ♣ , ♦, ♦, ä	252 132, 158, 2, 96, 5, 4, 4, 132	Ë ?, ø	203 136, 248
2400	□, ü , D, l, l, %, , ?	3, 252 19, 68, 73, 108, 37, 4, 136	?, ò #, ", l, ê	151, 242 35, 34, 73, 234
9600	~, ☉, s,,u,~ ~, ☉, 0,,,,,3,...	126, 2, 115, 0, 117, 126 126,2,48,0,0,0,63,51,0....	~, ☉, ò,,ò,ò,~ ~, ☉, ►,,,ò,?, ☉, ,,,,	126,2,147,0,149,126 126,2,16,0,0,0,147,63,
19200	ø, ▲,,▲,↑,,ð,f,, ø, ▲,,x,,À, ,,,	248,30,0,30,24,0,240,102,,, 248,30,0, 20,0,192,0,,,	ø,~,▲,?,▲,ø... ,,,,,	248,126,6,128,30,248,,, ,,,,,

Tabla 6.1. Resultados obtenidos mediante software de monitoreo

6.2 Mediciones realizadas con Osciloscopio digital *HP Infiniium*

6.2.a Para equipo ALCATEL:

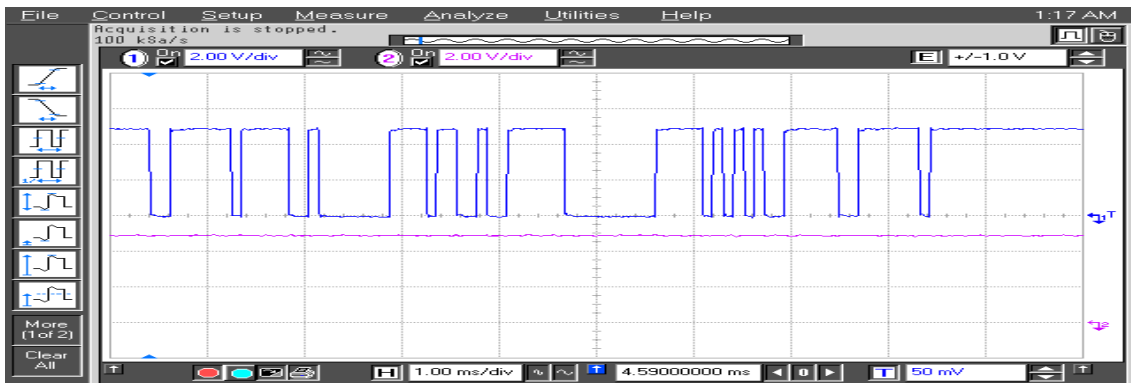


Figura 6.1. DSR respecto a los datos

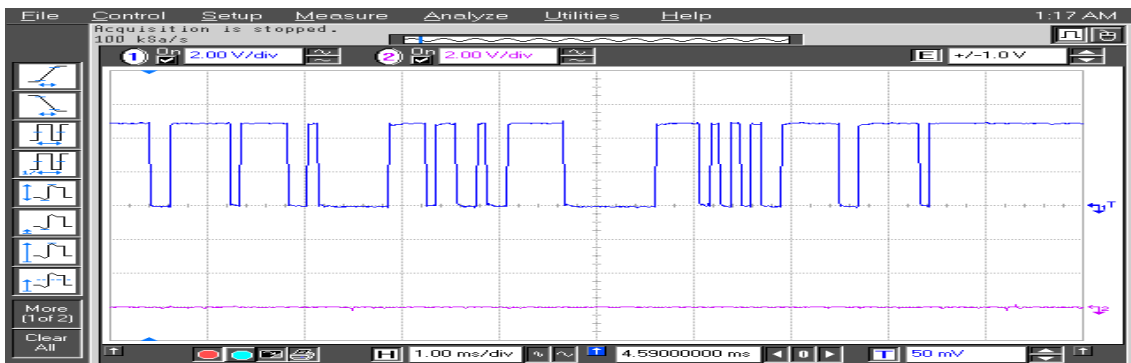


Figura 6.2 DTR respecto a los datos

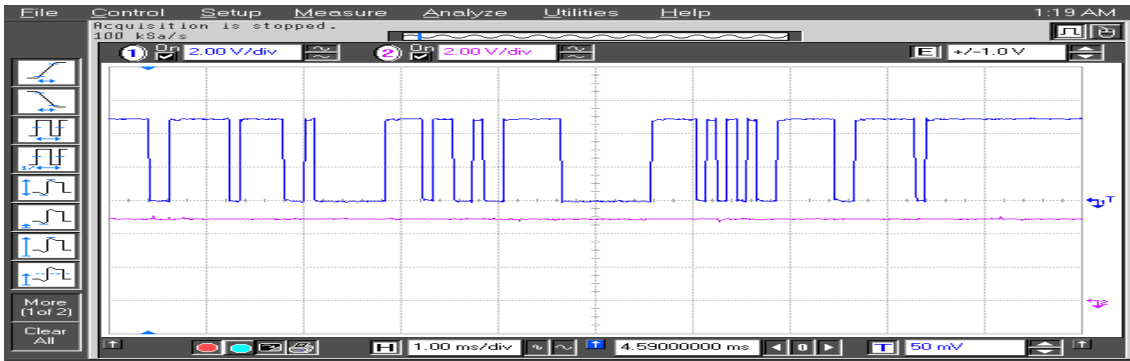


Figura 6.3. CTS respecto a los datos

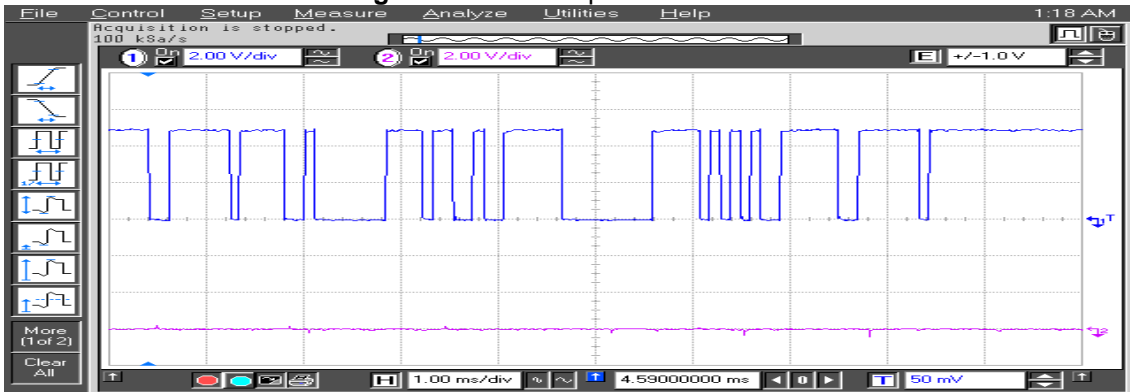


Figura 6.4. RTS respecto a los datos

6.2.b Para equipo NEC-W:

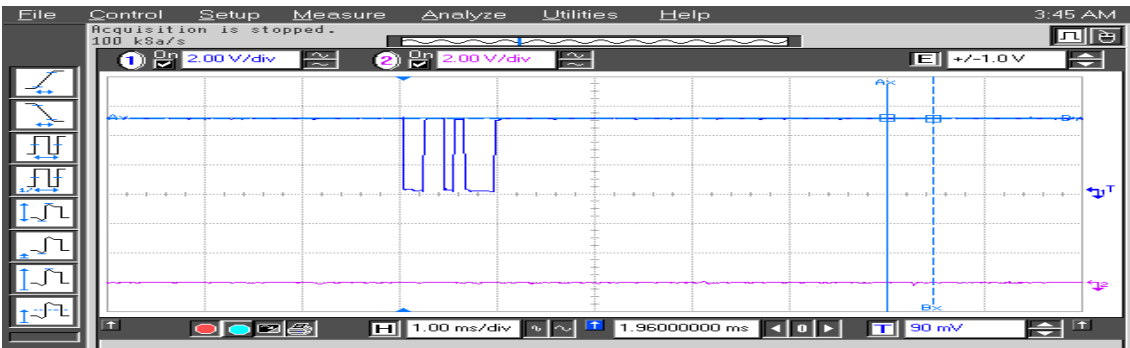


Figura 6.5. DSR respecto a los datos

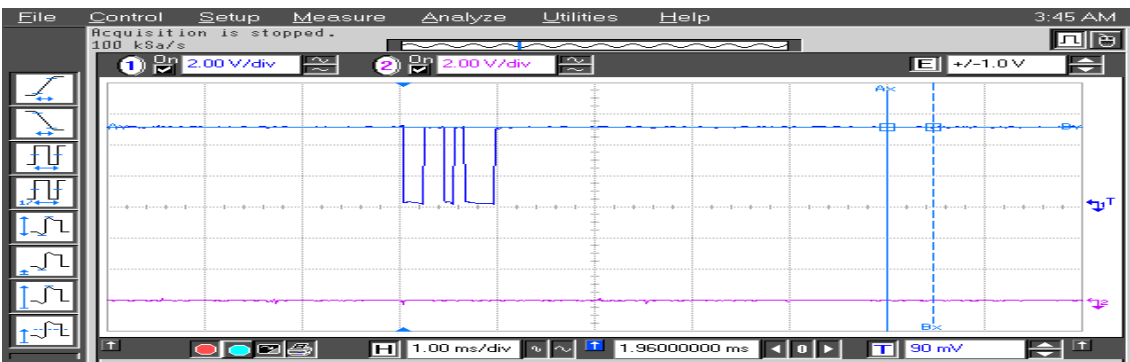


Figura 6.6. DTR respecto a los datos

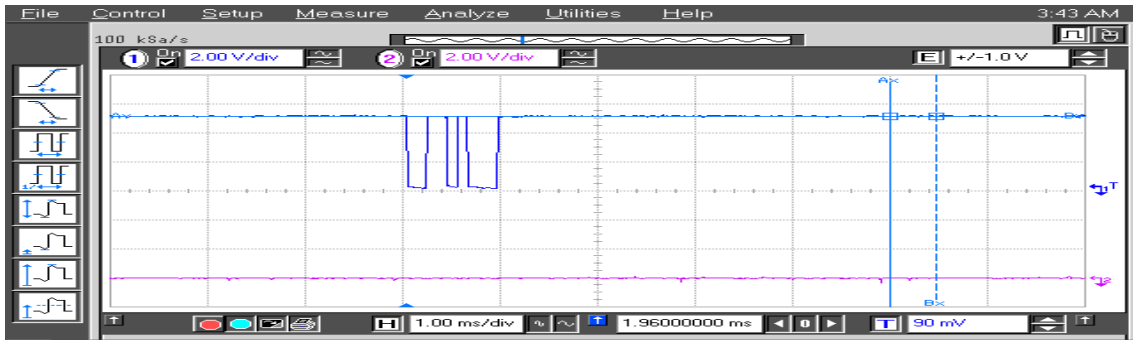


Figura 6.7. CTS respecto a los datos

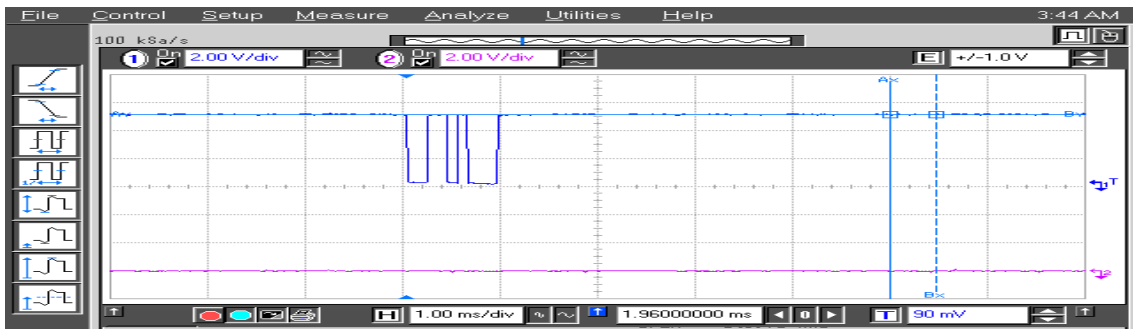


Figura 6.8. RTS respecto a los datos

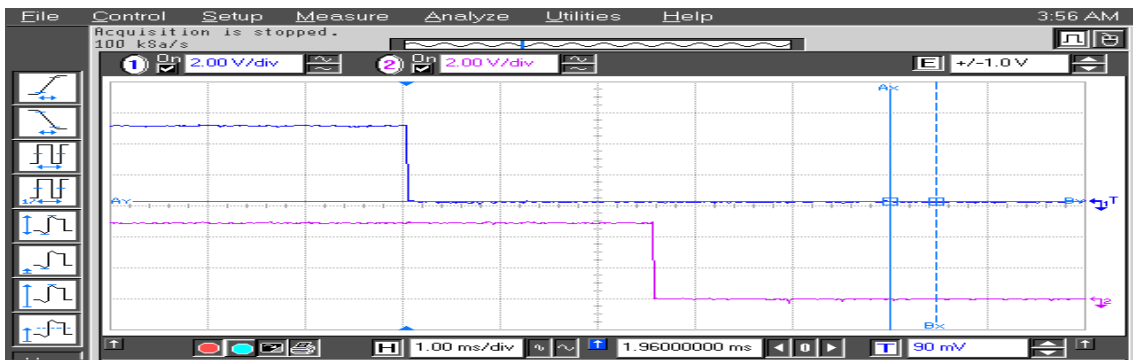


Figura 6.9. DSR respecto a CTS.

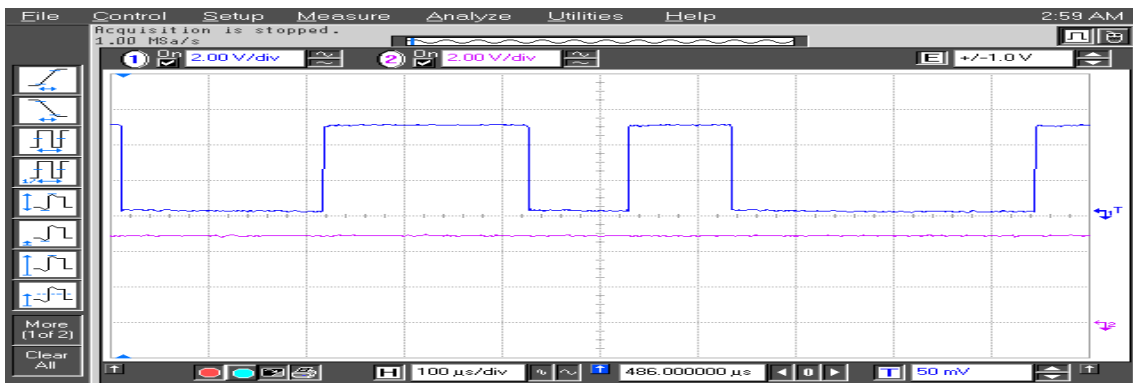


Figura 6.10. Respuesta del Equipo.

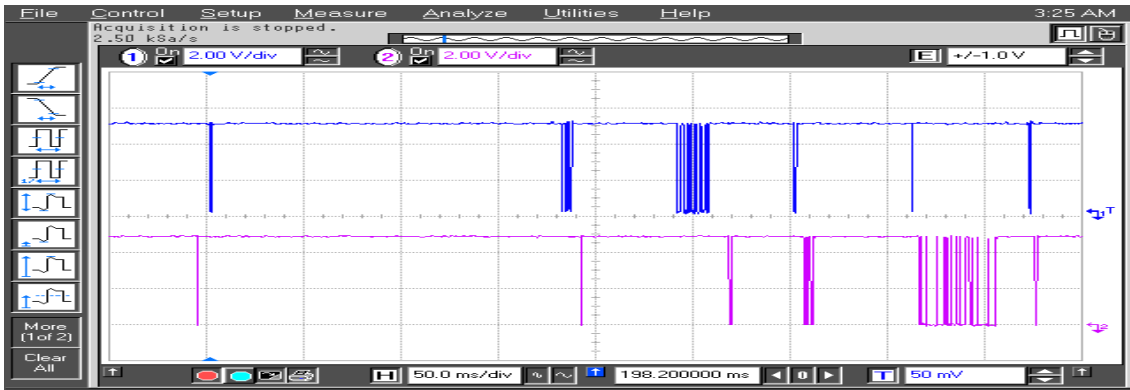


Figura 6.11. Orden de comunicación entre la PC y el Equipo.

Nota: la señal amarilla (arriba) es el dato de la PC, y la verde (abajo) el dato del equipo

6.2.c Para equipo NEC-V:



Figura 6.12. Dato enviado por la PC

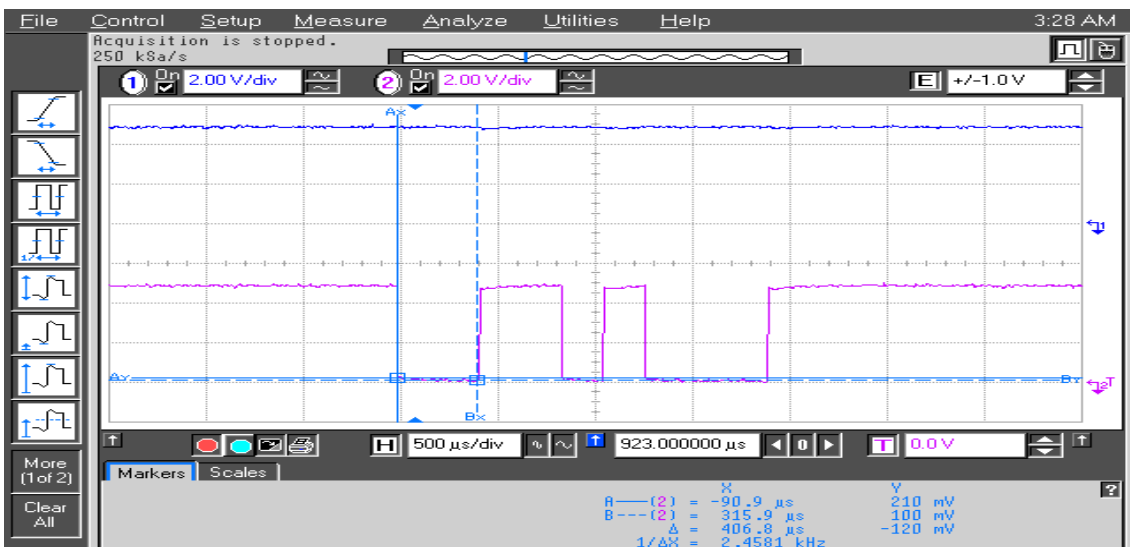


Figura 6.13. Dato enviado por el equipo NEC-V

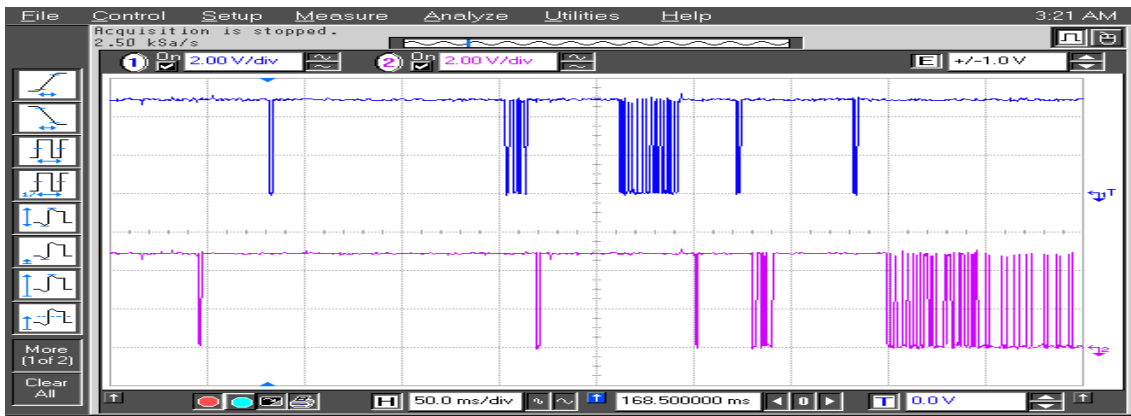


Figura 6.14. Comunicación entre el equipo NEC-V y la PC.

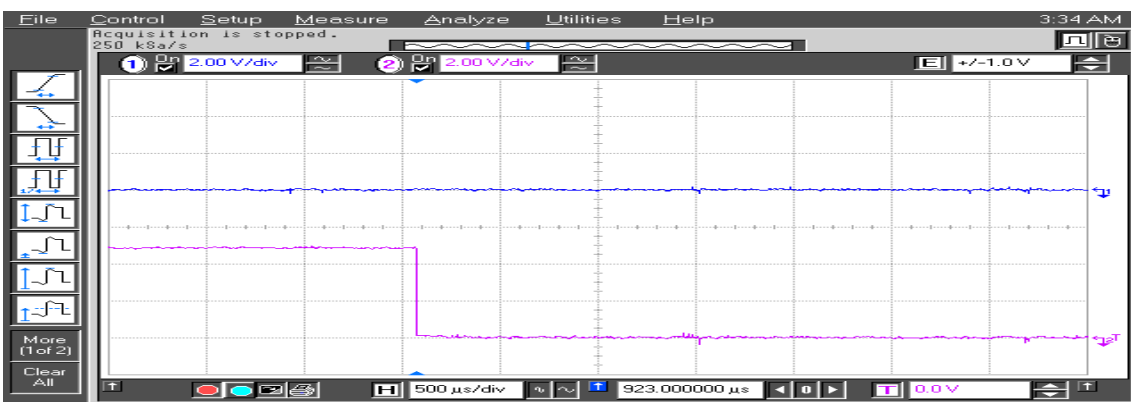


Figura 6.15. DTR vs DSR.

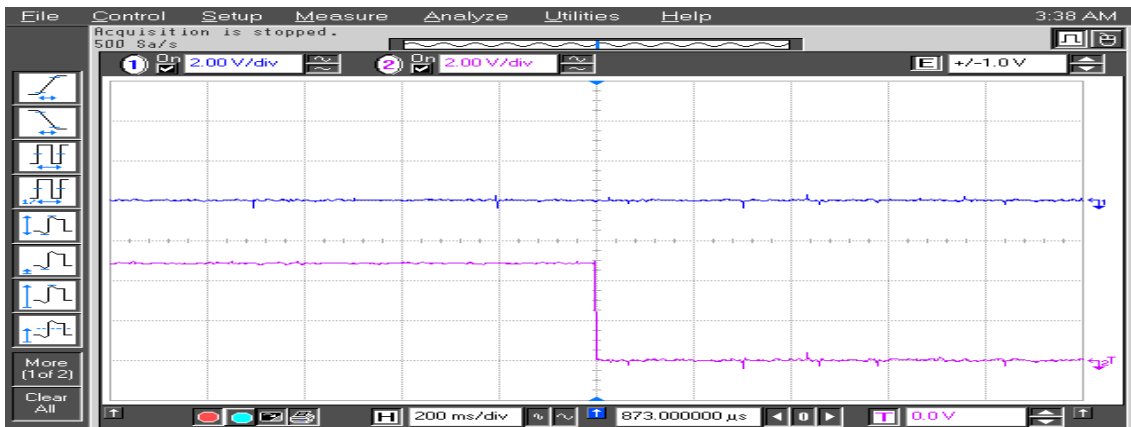


Figura 6.16. RTS vs CTS.

6.2.d Para equipo Lucent:

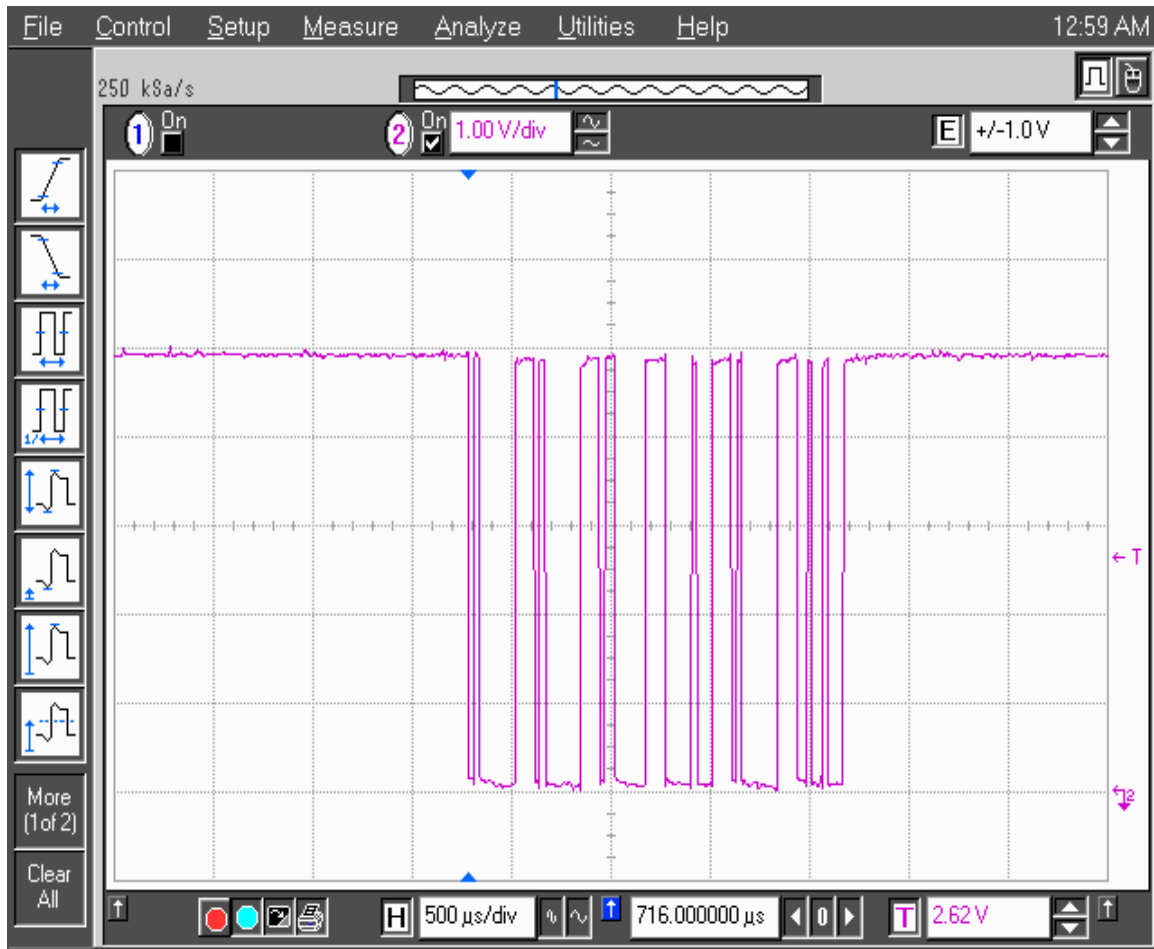


Figura 6.17. Dato enviado por la PC

6.3 Resultados obtenidos con software de monitoreo:

<i>Número de Caracter</i>	<i>Binario*</i>	<i>Hexadecimal</i>	<i>Decimal</i>
1	0111 1110	7E	126
2	11111001	9F	159
3	00000001	80	128
4	11101100	37	55
5	10011111	F9	249
6	10000000	1	1
7	00111110	7C	124
8	10101001	95	149
9	11111001	9F	159
10	11111011	DF	223

Tabla 6.2. Dato de envío de la PC al equipo Alcatel.

<i>Número de Caracter</i>	<i>Binario*</i>	<i>Hexadecimal</i>	<i>Decimal</i>
1	0111 1110	7E	126
2	11001000	13	19
3	00000101	A0	160
4	10011100	39	57
5	10000000	1	1
6	00110101	AC	172
7	01111001	9E	158
8	00111111	FC	252
9	01111111	FE	254

Tabla 6.3. Respuesta por parte del equipo Alcatel al microcontrolador

<i>Equipo</i>	<i>Velocidad (baudios)</i>	<i>Binario*</i>	<i>Hexadecimal</i>	<i>Decimal</i>
NEC V	4800	0110 1000	16	22
NEC W	9600	0110 1000	16	22

Tabla 6.4. Dato de envío y respuesta para el equipo NEC.

* Los valores en binario fueron el resultado de convertir el dato decimal a binario e invertirlo.

<i>Número de Caracter</i>	<i>Binario*</i>	<i>Hexadecimal</i>	<i>Decimal</i>
1	1000 0000	1	1
2	1111 0100	2F	47
3	0000 0111	E0	224
4	1111 1111	FF	255
5	0110 0000	6	6
6	0111 1111	FE	254
7	1111 0000	F	15
8	0100 0111	E2	226
9	1111 1111	FF	255
10	0100 0000	2	2
11	0111 1111	FE	254
12	1111 0010	4F	79
13	0100 0111	E2	226

Tabla 6.5. Caracteres enviados por la PC al equipo Lucent

<i>Caracter</i>	<i>Binario*</i>	<i>Hexadecimal</i>	<i>Decimal</i>
1	1000 0000	1	1
2	10 10 0000	5	5
3	00 10 0100	24	36
4	0000 1001	90	144
5	0110 00 10	46	70
6	0101 1001	9A	154

Tabla 6.6. Respuesta por parte del equipo Lucent al microcontrolador

* Son el resultado de convertir el de decimal a binario e invertirlo, que representa el dato de transmisión.

<i>Caracter</i>	<i>Binario*</i>	<i>Hexadecimal</i>	<i>Decimal</i>
1	00000011	C0	192
2	10 101001	95	149
3	01 10 0000	6	6
4	0000 1000	10	16
5	100000 10	41	65
6	00000000	00	0
7	100 10010	49	73
8	00 10 0000	4	4
9	0000 1000	10	16
10	00000011	C0	192

Tabla 6.7. Caracteres enviados por la PC al equipo Ericsson

<i>Caracter</i>	<i>Binario*</i>	<i>Hexadecimal</i>	<i>Decimal</i>
1	00000000	00	0
2	10 100110	65	101
3	00 10 0000	4	4
4	0011 1000	1C	28
5	000011 10	70	112
6	10100011	C5	197
7	10000000	1	1
8	11111111	FF	255

Tabla 6.8. Respuesta por parte del equipo Ericsson al microcontrolador

* Son el resultado de convertir el de decimal a binario e invertirlo, que representa el dato de transmisión.

6.4 Análisis

Las figuras, y tablas mostradas en la sección anterior del presente capítulo, representan los resultados de las mediciones realizadas, tanto de determinación de parámetros de configuración, como de comprobación de los mismos.

En la tabla 6.1, se muestran los caracteres de las tramas de envío, y de respuesta, del equipo Alcatel a distintas velocidades, que se obtuvieron mediante el software de control diseñado, cuya interfaz se observa en la figura 5.3. Estos resultados se utilizaron para retransmitirlos a través del software, pero no se tuvo éxito, ya que la configuración de la comunicación no es conocida, por lo tanto su incógnita no es únicamente la velocidad, es entonces que través de esta solución no se controlan todos los parámetros del enlace, por lo que se debe buscar una solución más efectiva.

En vista de lo anterior se procedió a realizar la medición de las tramas, a través de un osciloscopio digital, cuyos resultados se observan desde la figura 6.1 hasta la figura 6.17.

Se presentan primero los resultados para el equipo Alcatel, después para el NEC, y por último para el Lucent, del que únicamente se obtuvo el dato de transmisión de la PC, como se observa en la figura 6.17, ya que éste opera a una velocidad de [♦]38400 baudios, haciendo imposible monitorear la lectura de la comunicación en forma bidireccional, por que se buscó una solución más automatizada, como fue la implementación del microcontrolador.

De la figura 6.1 a la figura 6.4, se muestra los resultados obtenidos para el equipo Alcatel, de lo que se puede obtener, en el protocolo de comunicación:

- PC baja las siguientes señales:
 - DTR -> 0
 - RTS -> 0
- PC envía el siguiente dato (en Hexadecimal):
 - 7E9F-8037-F901- 7C95- 9FDF;
- Equipo responde con dato.
 - 7E13- A039-01AC- 9EFC-FEFF

[♦] la velocidad del equipo se obtuvo en la medición con el osciloscopio, a una mayor escala.

De la misma manera se analiza los resultados para el equipo NEC, cuyas mediciones se muestran de la figura 6.5 a la 6.16, en las que se observa el estado de todas las señales, en función de la señal de los datos de envío de la computadora; Así como el protocolo de comunicación con el equipo, mostrado en la figura 6.14. A partir de los resultados obtenidos para el equipo NEC, se puede llegar a la siguiente conclusión para lograr comunicarse con el equipo.

- PC baja las siguientes señales:
 - DTR -> 0
 - RTS -> 0
- Equipo baja las siguientes señales:
 - DSR -> 0
 - CTS -> 0
- PC envía el siguiente dato (en Hexadecimal):
 - 16;
- Equipo responde con dato (en Hexadecimal):
 - 16;

A pesar de que las lecturas con el osciloscopio, son una forma de medición muy exacta, su proceso se dificulta en función de la velocidad de comunicación; para el caso del equipo Lucent, la lectura y control del protocolo de la trama, no se pudo realizar, por lo que se optó por realizar una medición más automatizada, y más eficiente, mediante un microcontrolador. Los resultados obtenidos se muestran en las tablas 6.2 a la 6.7.

Analizando el equipo Alcatel, es importante señalar que realizando una lectura a una mayor escala del dato de la figura 6.1, se puede comprobar que es igual al de transmisión de la tabla 6.2.

De igual manera se procedió con el dato de respuesta, a una mayor escala, el valor de lectura también fue comparado, con el valor leído con el microcontrolador, mostrado en la tabla 6.3.

Como se observa en las figuras 6.5 y 6.12, el dato de envío para el equipo NEC-W y NEC-V respectivamente, son el mismo un 22 decimal, pero con la particularidad de que la velocidad es diferente (9600 y 4800 respectivamente).

Éste valor de envío coincide con el leído por el microcontrolador, mostrado en la figura 6.4, que es el mismo que el dato de respuesta por parte del equipo como se muestra en las figuras 6.10 y 6.13.

La señal mostrada en la figura 6.17, corresponde a la trama de datos enviada por la computadora, hacia el equipo Lucent, con la que se hizo una lectura bit por bit de toda la trama en el osciloscopio, a una mayor escala, y determinando que dicho valor coincide con el recibido por el microcontrolador, mostrada en la tabla 6.5. Y a pesar de que no se hizo una lectura de los datos de respuesta del equipo Lucent, con el osciloscopio, dicha trama se resume en la tabla 6.6, cuyo dato fue leído con el microcontrolador.

Por último para el equipo Ericsson, se realizaron mediciones de las tramas de datos, las cuales se muestran en las tablas 6.7 y 6.8, cuya velocidad de comunicación es de 9600 baudios.

Con los resultados obtenidos anteriormente, e integrando todas las partes de la solución se comprobó el funcionamiento del dispositivo mediador universal, reconociendo y configurando los equipos de telecomunicaciones, planteado como meta y objetivo del proyecto. Además se logró cubrir más alcances, con la lectura de tramas para agregar equipos nuevos.

Capítulo 7: Conclusiones y Recomendaciones

7.1 Conclusiones:

1. La reducción del hardware para la multiplexación de las señales, requirió de un dispositivo integrado programable (en este caso se utilizó una GAL).
2. Se logró implementar el dispositivo para el reconocimiento de las cuatro marcas planteadas, y además se dejó provisto para un equipo adicional.
3. Se satisfizo características de diseño como capacidad de memoria, y tiempos de las señales, mediante el microcontrolador seleccionado.
4. El análisis de tiempos realizado arroja que existe una variación en las velocidades y formatos de acuerdo al equipo seleccionado.
5. Se logra determinar que la longitud máxima de la trama es de 13 bytes con lo que se dimensiona la capacidad mínima de la memoria EEPROM.
6. El dispositivo es capaz de leer tanto los datos de envío como de respuesta de los equipos multiplexores SDH.
7. El hardware de control es capaz de reconocer los multiplexores (SDH) en forma automática, o Manual.

7.2 Recomendaciones:

1. Analizar la posibilidad de utilizar un microcontrolador con mayor espacio de almacenamiento en la EEPROM, para abarcar mayor cantidad de equipos.
2. Se sugiere realizar el estudio para la incorporación de métodos para el aprendizaje de instrucciones de pruebas y mantenimiento de tal forma que se logre mayor funcionalidad del dispositivo.
3. Analizar la viabilidad de usar el integrado FT232BM 408-1, convertidor de USB a serial, para la comunicación del DMU con el programa de control.
4. Analizar la posibilidad de implementar la comunicación del dispositivo con otros sistemas operativos.

Bibliografía

- [1] Lozano, J. 2003. Comunicación Serial. Disponible en dirección <http://www.iaa.upf.es/~jlozano/interfases/serial.html>
- [2] San Martín, José. 2005. Periféricos de Computadores. Disponible en dirección <https://dac.escet.urjc.es/docencia/PC/Tema2.pdf>
- [3] Marqués, Mercedes. 2001. El Modelo Entidad-Relación. Disponible en dirección <http://www3.uji.es/~mmarques/f47/apun/node83.html>
- [4] García Chávez, Carlos. 2005. Diseño de base de datos relacionales. Disponible en dirección <http://www.mailxmail.com/curso/informatica/disenobasesdatosrelacionales/capitulo7.htm>.
- [5] 2004. Comunicación Serie. Disponible en dirección <http://tamarisco.datsi.fi.upm.es/ASIGNATURAS/TC/practicas/DescripcionUART.pdf>

Apéndice A. Medición de parámetros de la comunicación serial

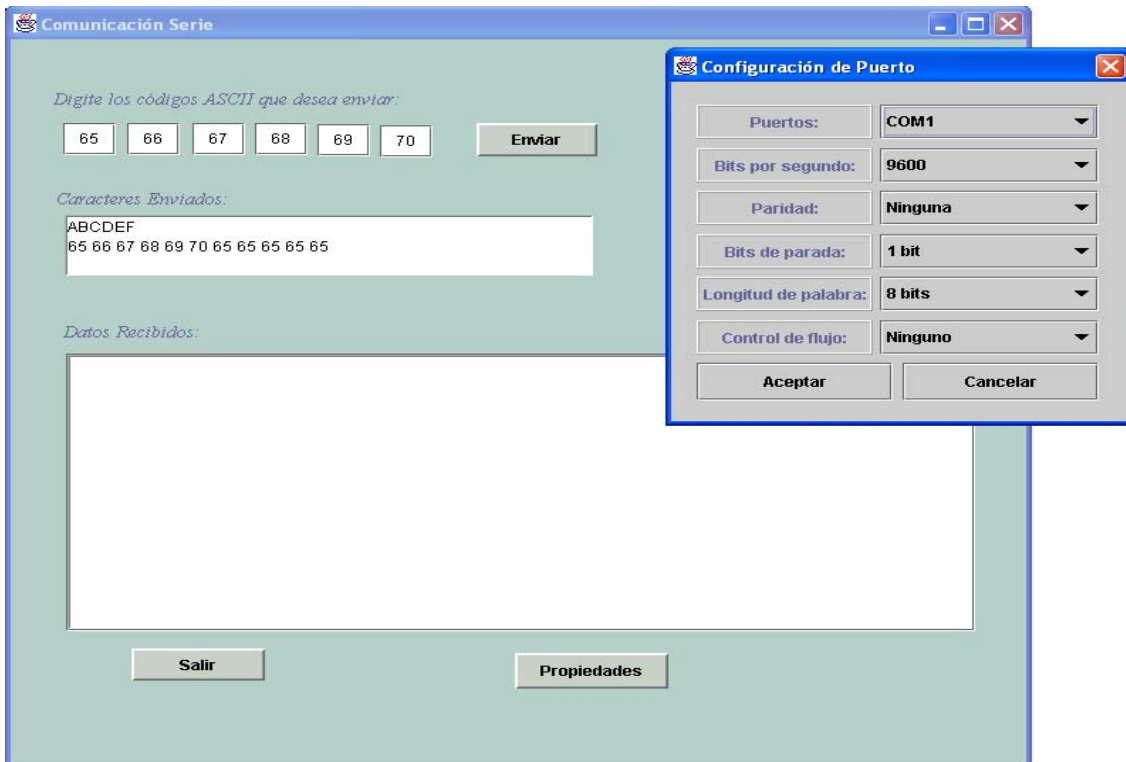


Figura A.1. Software de comunicación, transmisor de una trama de 6 bytes, con configuración variable.

Velocidad	Paridad	Bits de parada	Long. de palabra	Control de Flujo	Bits de espaciado	Cant. Bits espaciado
9600	Ninguna	1	8	Ninguno	10	2
9600	Par	1	8	Ninguno	010	3
9600	impar	1	8	Ninguno	110	3
9600	ninguna	2	8	Ninguno	110	3
9600	par	2	8	Ninguno	0110	4
9600	impar	2	8	Ninguno	1110	4
9600	Ninguna	1	8	Software	No afecta	No afecta
9600	Ninguna	1	8	Hardware	No afecta	No afecta

Tabla A.1. Configuración en comunicación serial

Esta tabla muestra tanto los bits de espaciado, como la cantidad para distintas combinaciones en la configuración asíncrona.

Apéndice B. Mapa de memoria EEPROM, del PIC

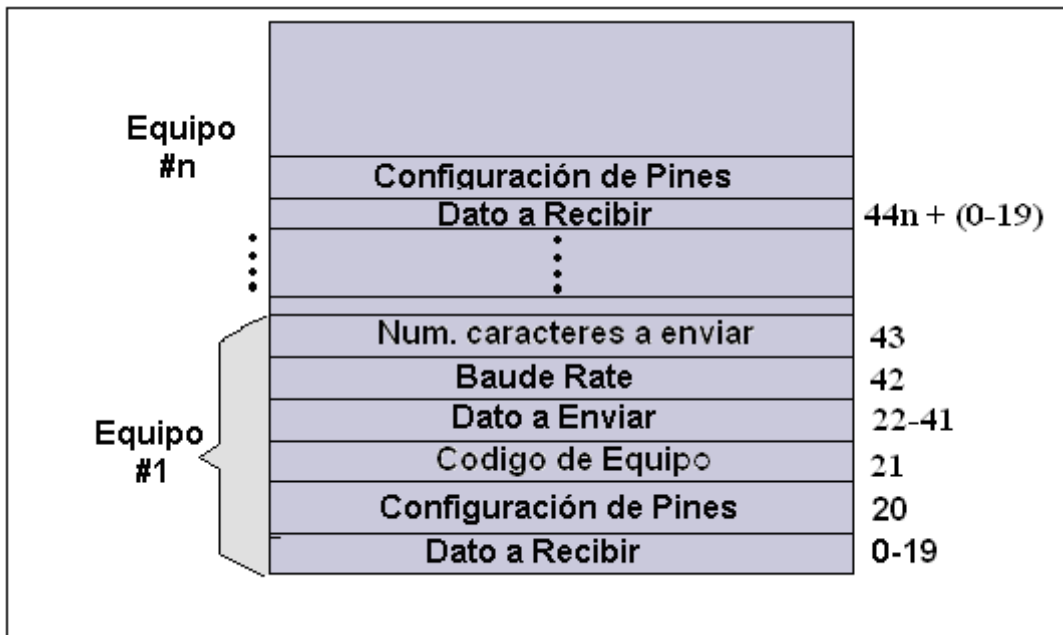
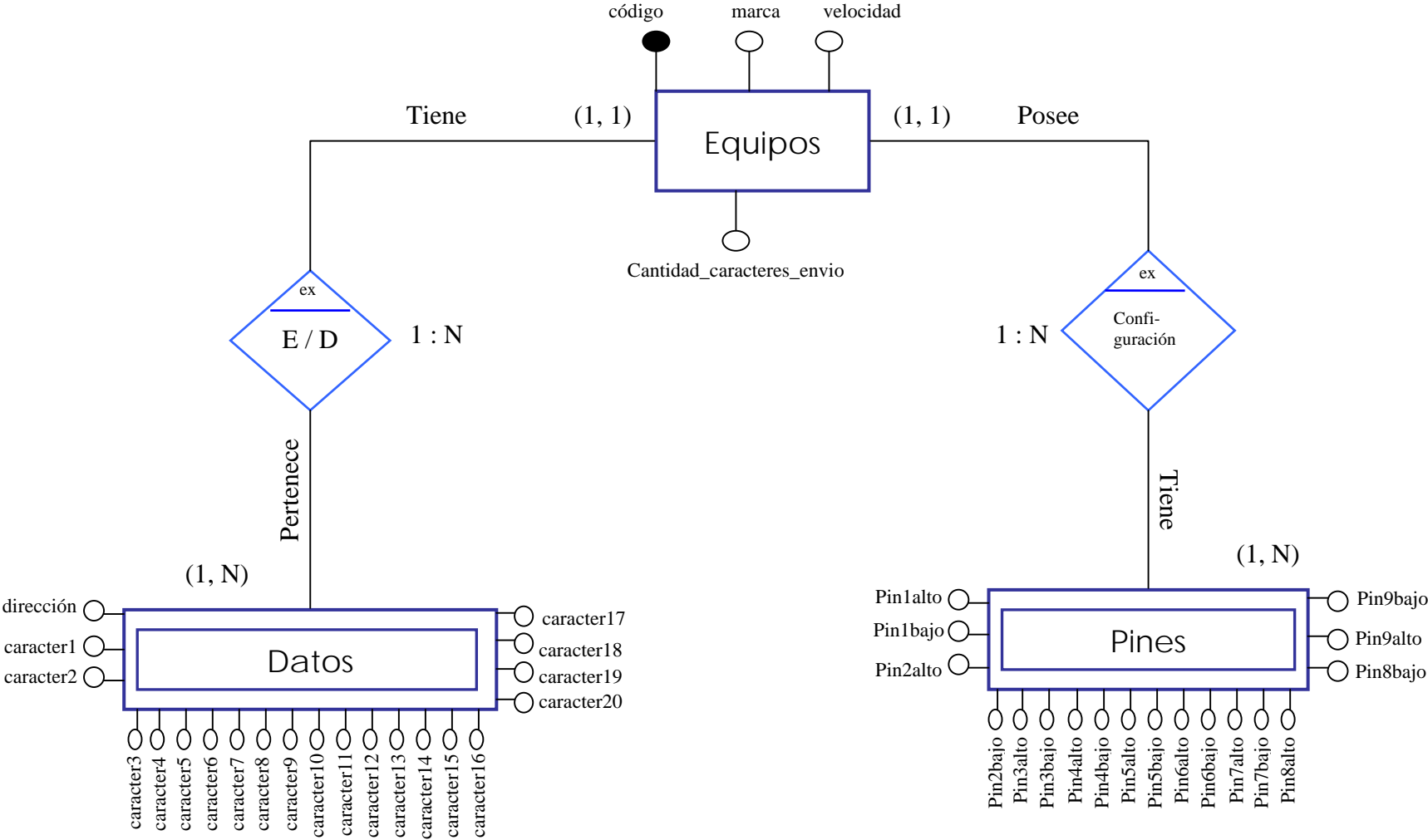


Figura B.1. Mapa de memoria de almacenamiento de la EEPROM.

Apéndice C. Modelo entidad-relación.



Apéndice D. Código de la rutina de lectura en lenguaje ensamblador

```
..... for(j=0;j<8;j++){ //leer bit y concatenar
0213: CLRF 23
0214: MOVF 23,W
0215: SUBLW 07
0216: BTFSS 03.0
0217: GOTO 242
..... delay_us(NbaudRate);
0218: MOVF 20,W
0219: MOVWF 2F
021A: GOTO 18A
..... if(input(PIN_A1))
021B: BSF 03.5
021C: BSF 05.1
021D: BCF 03.5
021E: BTFSS 05.1
021F: GOTO 230
..... bit_set(cteEnviar[i],j);
0220: MOVLW 24
0221: ADDWF 22,W
0222: MOVWF 04
0223: MOVLW 01
0224: MOVWF 77
0225: MOVF 23,W
0226: MOVWF 78
0227: BTFSC 03.2
0228: GOTO 22D
0229: BCF 03.0
022A: RLF 77,F
022B: DECFSZ 78,F
022C: GOTO 229
022D: MOVF 77,W
022E: IORWF 00,F
..... else
022F: GOTO 240
..... bit_clear(cteEnviar[i],j);
0230: MOVLW 24
0231: ADDWF 22,W
0232: MOVWF 04
0233: MOVLW 01
0234: MOVWF 77
0235: MOVF 23,W
0236: MOVWF 78
0237: BTFSC 03.2
0238: GOTO 23D
0239: BCF 03.0
023A: RLF 77,F
023B: DECFSZ 78,F
023C: GOTO 239
023D: MOVF 77,W
023E: XORLW FF
023F: ANDWF 00,F
..... }
0240: INCF 23,F
0241: GOTO 214
..... }//for limite tramas
```

|

Apéndice E. Configuración de Pines

PC	Alcatel	Lucent	NEC
1	1	N.C.	1
2	2	4	2
3	3	5	3
4	4	7	4
5	5	3	5
6	6	7	6
7	7	2	7
8	8	6	8
9	9	N.C.	9

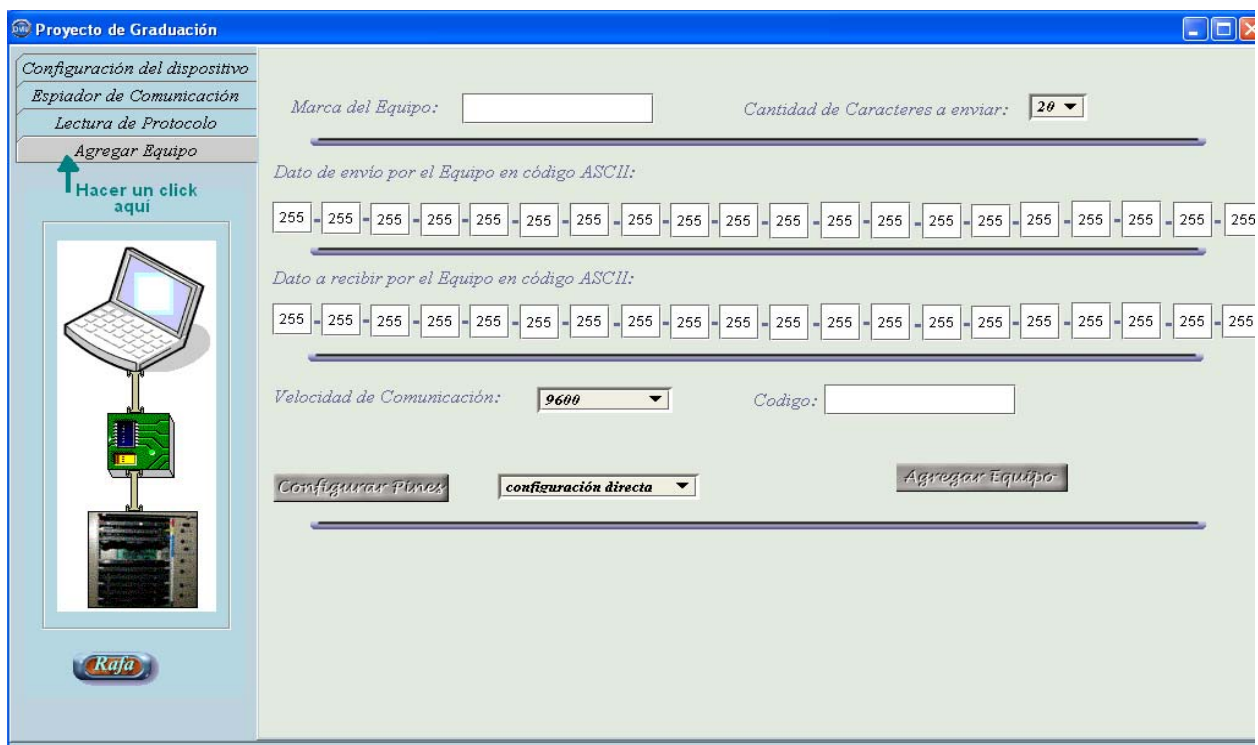
Tabla E.1. Configuración de pines para las 4 marcas de equipos evaluados

Apéndice F. Manual de Usuario

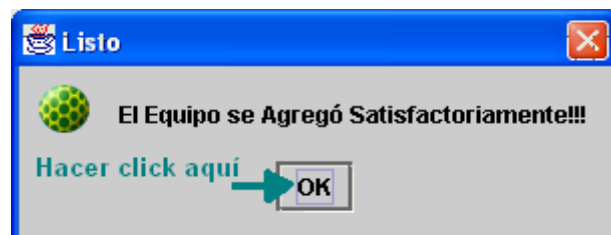
Agregar nuevos equipos

Para agregar un equipo nuevo, siga los siguientes pasos:

1. Hacer un click en Agregar Equipo



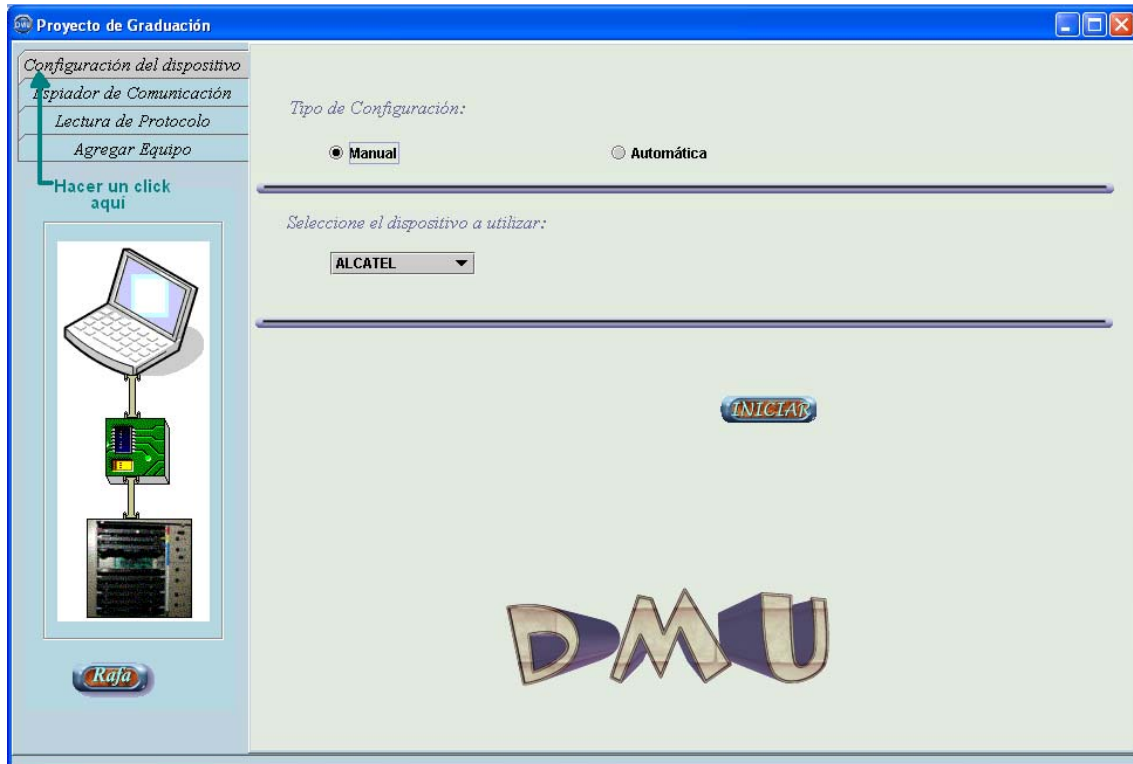
2. Colocar en los campos respectivos la información del que se desea agregar, que son la marca del equipo, la cantidad de caracteres a enviar, las tramas de envío y de respuesta, la velocidad, el código del equipo y la configuración de pines utilizada.
3. Hacer un click en el botón 'Agregar Equipo'.
4. Esperar respuesta por parte del equipo.
5. Hacer un clic en el botón 'OK'.



Modo de Configuración

Para seleccionar el modo de configuración

1. Hacer un click en Configuración del dispositivo



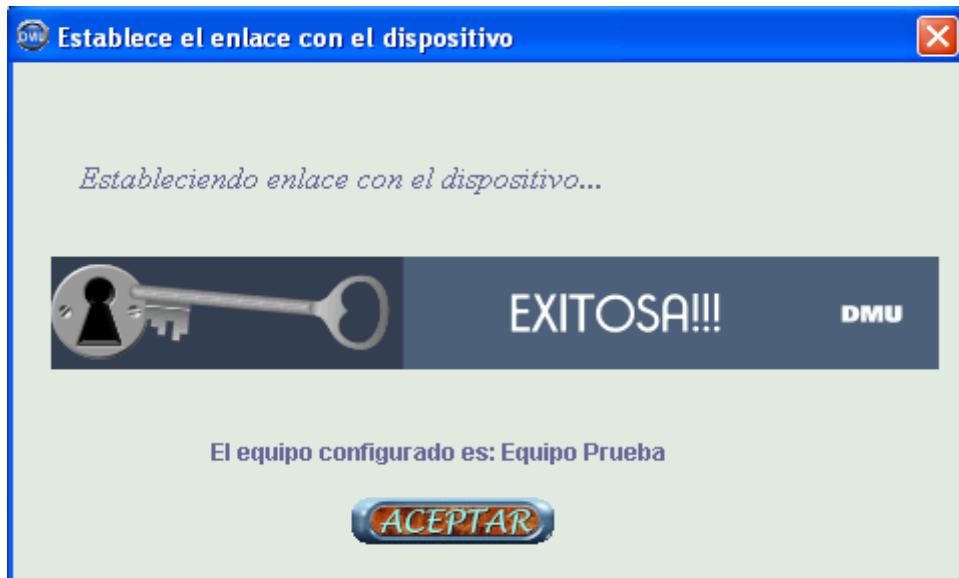
2. Seleccionar el modo de Configuración.

Modo de Configuración Manual:

1. Hacer clic a Manual en 'Tipo de Configuración'.
2. Seleccionar el dispositivo a conectar.
3. Hacer click en el botón 'Iniciar'.
4. Esperar respuesta por parte del equipo.



5. Hacer click en botón 'OK'.



Modo de Configuración Automática:

1. Hacer clic a Automática en 'Tipo de Configuración'.
2. Hacer click en el botón 'Iniciar'.
3. Esperar respuesta por parte del equipo.

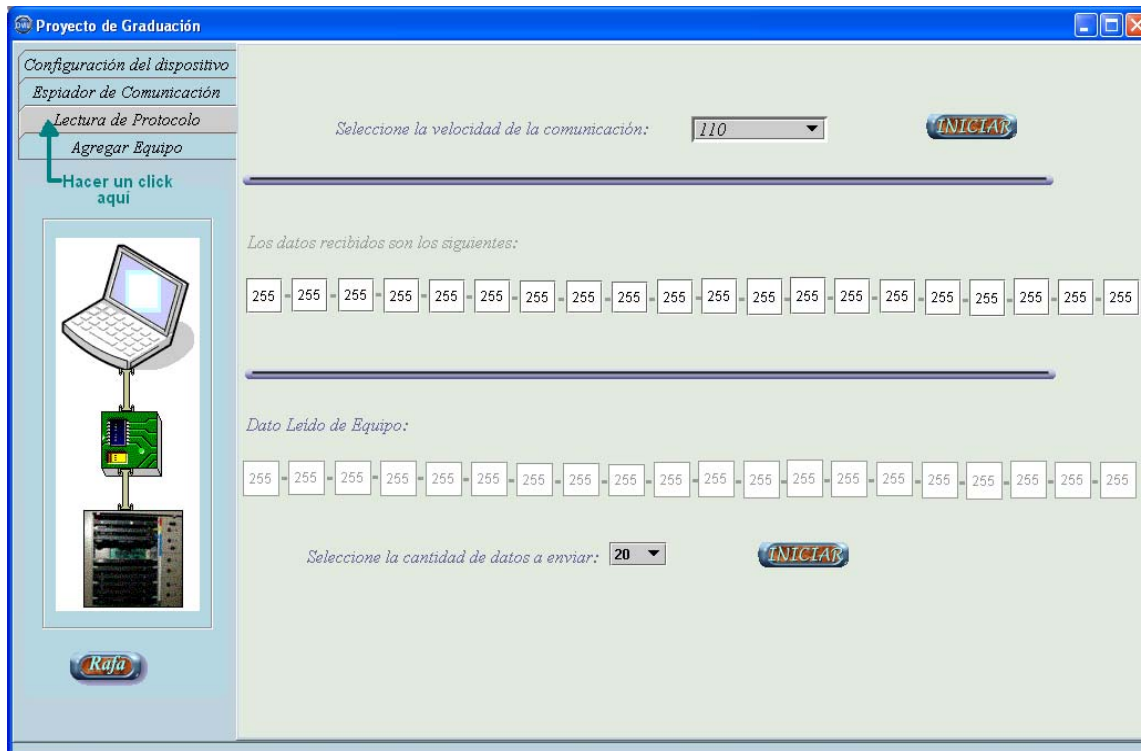


4. Hacer click en botón 'OK'.



Lectura de Tramas

1. Hacer click en 'Lectura de Protocolo'.



Trama de envío

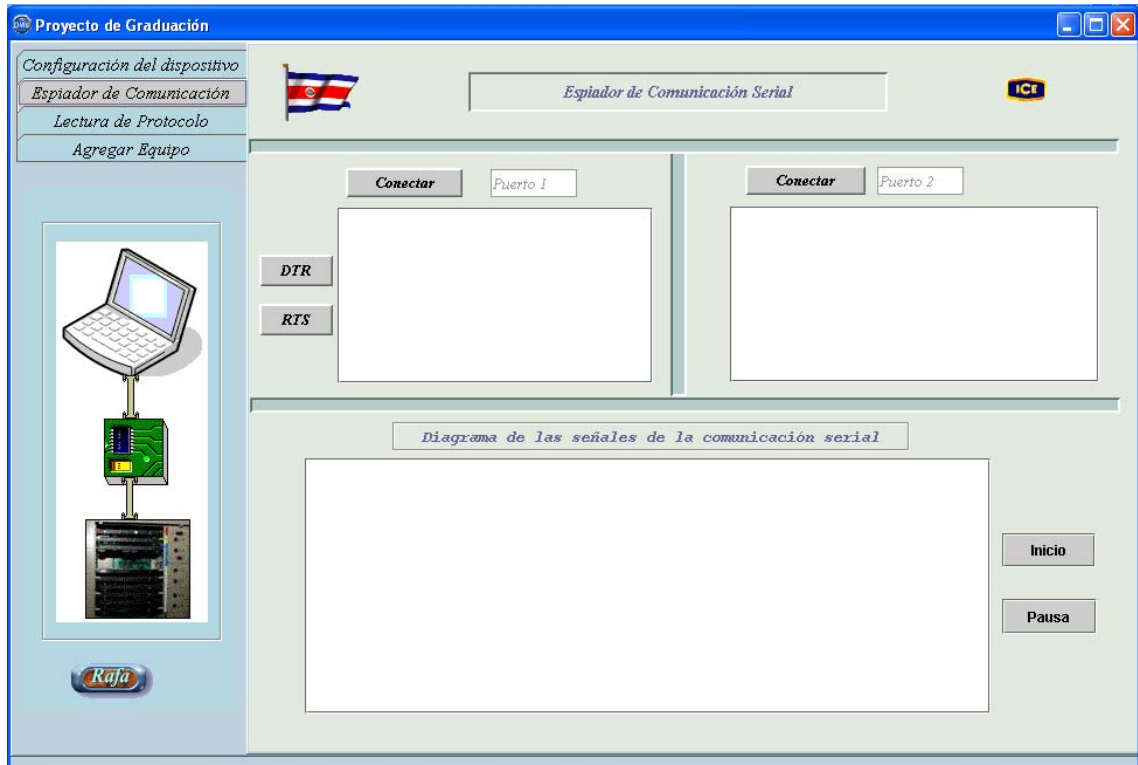
1. Seleccionar la velocidad de comunicación
2. Conectar el cable serie a la computadora.
3. Iniciar el programa del equipo en cuestión.
4. Hacer click en el botón iniciar.
5. Esperar dato de respuesta.

Trama de respuesta

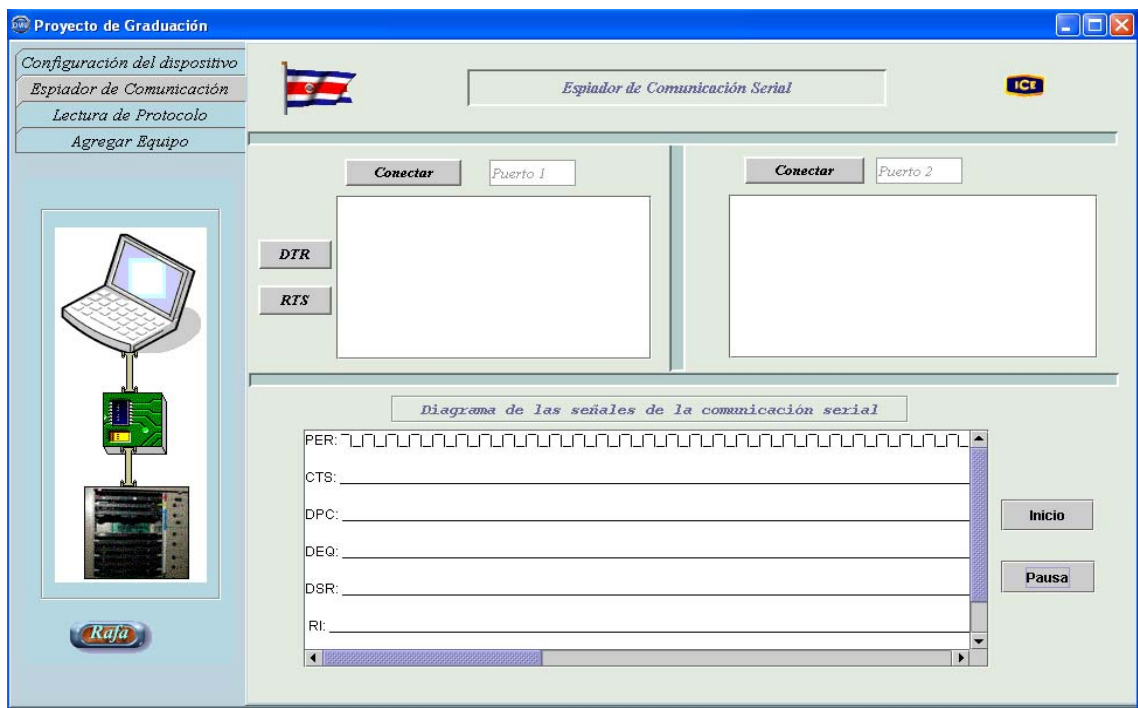
1. Seleccionar la velocidad de comunicación
2. Conservar el dato recibido(o escribirlo nuevamente) de la computadora.
3. Seleccionar la cantidad de caracteres a enviar(los últimos cuadros que tienen un valor de 255, no se toman como datos).
4. Hacer clic en el botón 'Iniciar'.
5. Esperar dato de respuestta.

Espiador de Puerto Serie

1. Hacer clic en 'Espiador de Comunicación'.



2. Hacer click en el botón 'Conectar', para recibir datos de los puertos.
3. Hacer click en el botón de 'Inicio', para observar un diagrama temporal de las señales.



Apéndice G. Glosario, abreviaturas y simbología

- BER: Bit Error Rate
- UTP: Par trenzado no blindado.
- RS232: enviar datos en serie
- USB: Universal Serial Bus
- Bauds: baudios (símbolos) por segundo.
- Bps: Convención utilizada para definir las velocidades de transmisión. Bits por segundo.
- Equipo de medición: equipo utilizado para el estudio de los parámetros
- Equipo: equipo utilizado para las telecomunicaciones que brinda el ICE
FDMA: Frequency Division Multiple Access
- ICE: Instituto Costarricense de Electricidad
- IEEE: Institute of Electrical and Electronics engineers
- Infocomunicaciones: voz y datos.
- LAN: Local Area Network

Apéndice H. Información sobre la empresa

El Instituto Costarricense de Electricidad (ICE) fue creado por el Decreto - Ley No.449 del 8 de abril de 1949 como una institución autónoma, con personalidad jurídica y patrimonio propio. Está dotado de plena autonomía e independencia administrativa, técnica y financiera. Al ICE le corresponde, por medio de sus empresas, desarrollar, ejecutar, producir y comercializar todo tipo de servicios públicos de electricidad y telecomunicaciones, así como actividades o servicios complementarios a estos.

Como objetivos primarios el ICE debe desarrollar, de manera sostenible, las fuentes productoras de energía existentes en el país y prestar el servicio de electricidad. A su vez, se encarga de desarrollar y prestar los servicios de telecomunicaciones, con el fin de promover el mayor bienestar de los habitantes del país y fortalecer la economía nacional.

Posteriormente, en 1963 y por medio de la Ley No. 3226, la Asamblea Legislativa le confirió al ICE un nuevo objetivo: el establecimiento, mejoramiento, extensión y operación de los servicios de comunicaciones telefónicas, radiotelegráficas y radiotelefónicas en el territorio nacional. Tres años más tarde, instaló las primeras centrales telefónicas automáticas y, a partir de entonces, las telecomunicaciones iniciaron su desarrollo.

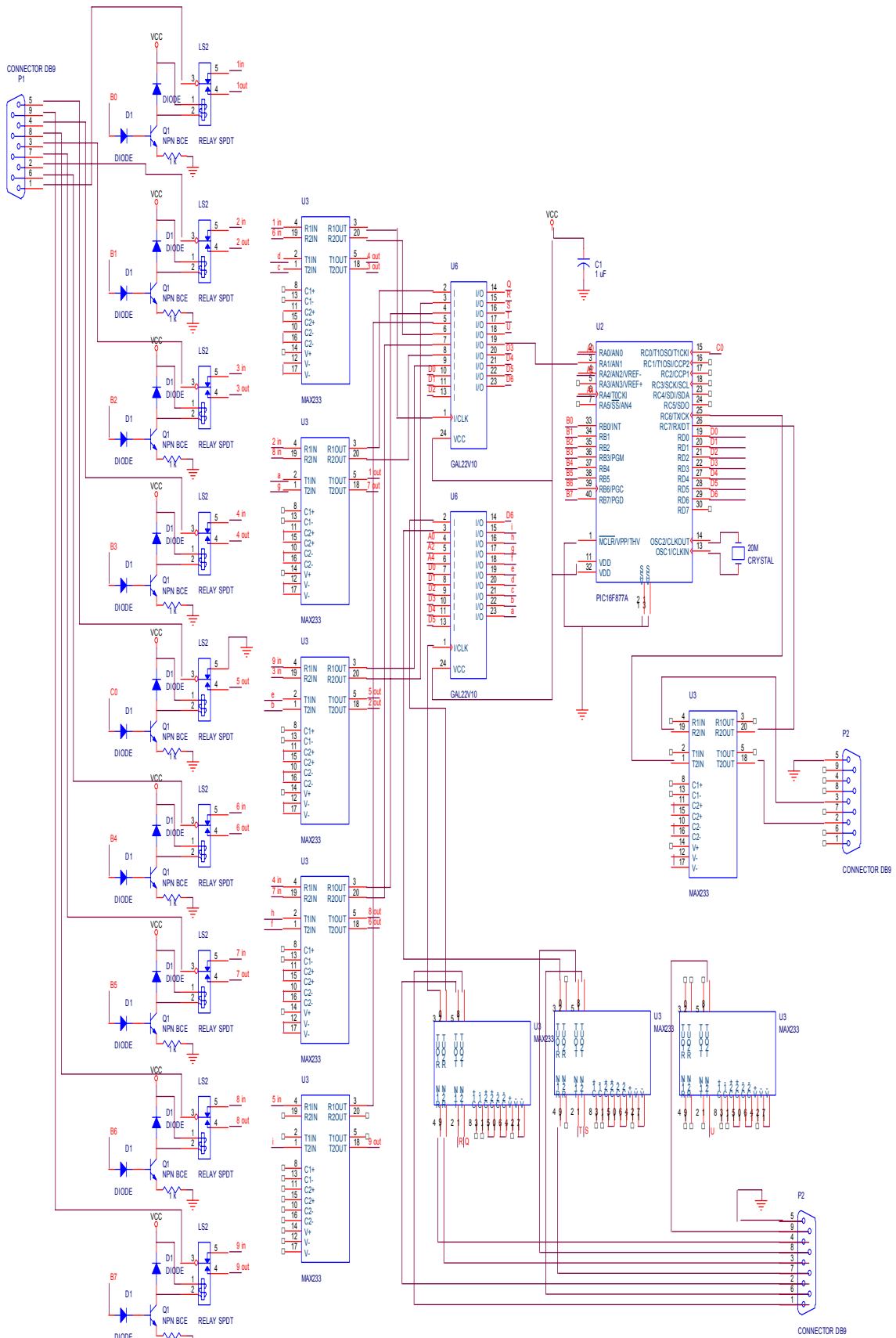
Con el devenir del tiempo, ha evolucionado como un grupo corporativo de empresas estatales, integrado por el propio ICE (Sectores Electricidad y Telecomunicaciones) y sus empresas: Radiográfica Costarricense S.A. (RACSA) y la Compañía Nacional de Fuerza y Luz S.A. (CNFL), las cuales han trazado su trayectoria, mediante diversos proyectos de modernización desarrollados en las últimas décadas.

Apéndice I. Descripción del área donde se realizó el proyecto

El proyecto se realizará en el área de mantenimiento preventivo y correctivo en las telecomunicaciones, que el ICE brinda a nivel nacional.

El lugar donde se encuentran los equipos se compone de una sala en forma rectangular, la cual internamente se divide en filas de equipos.

Apéndice J. Diagrama de conexión con el uso de Relay's.



Anexo A. Hojas de datos del multiplexor/demultiplexor CD74HC4067



Data sheet acquired from Harris Semiconductor
SCHS209C

February 1998 - Revised July 2003

CD74HC4067, CD74HCT4067

High-Speed CMOS Logic 16-Channel Analog Multiplexer/Demultiplexer

Features

- Wide Analog Input Voltage Range
- Low "ON" Resistance
 - $V_{CC} = 4.5V$ 70Ω (Typ)
 - $V_{CC} = 6V$ 60Ω (Typ)
- Fast Switching and Propagation Speeds
- "Break-Before-Make" Switching. 6ns (Typ) at 4.5V
- Available in Both Narrow and Wide-Body Plastic Packages
- Fanout (Over Temperature Range)
 - Standard Outputs 10 LSTTL Loads
 - Bus Driver Outputs 15 LSTTL Loads
- Wide Operating Temperature Range ... $-55^{\circ}C$ to $125^{\circ}C$
- Balanced Propagation Delay and Transition Times
- Significant Power Reduction Compared to LSTTL Logic ICs
- HC Types
 - 2V to 6V Operation
 - High Noise Immunity: $N_{IL} = 30\%$, $N_{IH} = 30\%$ of V_{CC} at $V_{CC} = 5V$
- HCT Types
 - 4.5V to 5.5V Operation
 - Direct LSTTL Input Logic Compatibility, $V_{IL} = 0.8V$ (Max), $V_{IH} = 2V$ (Min)
 - CMOS Input Compatibility, $I_I \leq 1\mu A$ at V_{OL} , V_{OH}

Description

The CD74HC4067 and CD74HCT4067 devices are digitally controlled analog switches that utilize silicon-gate CMOS technology to achieve operating speeds similar to LSTTL, with the low power consumption of standard CMOS integrated circuits.

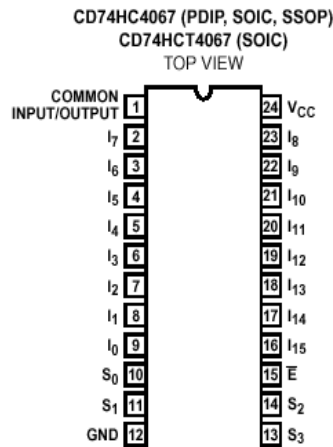
These analog multiplexers/demultiplexers control analog voltages that may vary across the voltage supply range. They are bidirectional switches thus allowing any analog input to be used as an output and vice-versa. The switches have low "on" resistance and low "off" leakages. In addition, these devices have an enable control which when high will disable all switches to their "off" state.

Ordering Information

PART NUMBER	TEMP. RANGE (°C)	PACKAGE
CD74HC4067E	-55 to 125	24 Ld PDIP
CD74HC4067M	-55 to 125	24 Ld SOIC
CD74HC4067M96	-55 to 125	24 Ld SOIC
CD74HC4067SM96	-55 to 125	24 Ld SSOP
CD74HCT4067M	-55 to 125	24 Ld SOIC

NOTE: When ordering, use the entire part number. The suffix 96 denotes tape and reel.

Pinout



CAUTION: These devices are sensitive to electrostatic discharge. Users should follow proper IC Handling Procedures.

Copyright © 2003, Texas Instruments Incorporated

Anexo B. Hojas de datos de la GAL 22V10



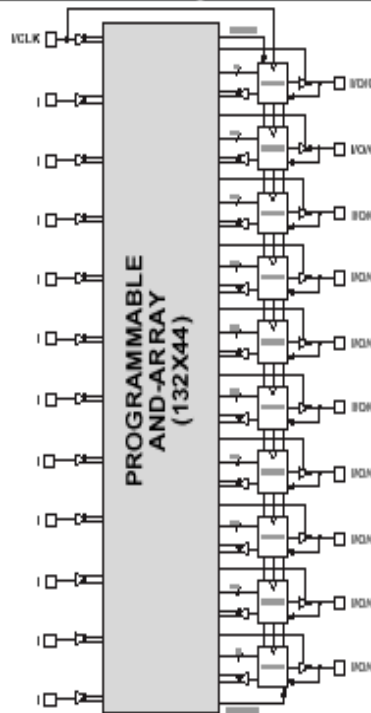
GAL22V10

High Performance E²CMOS PLD
Generic Array Logic™

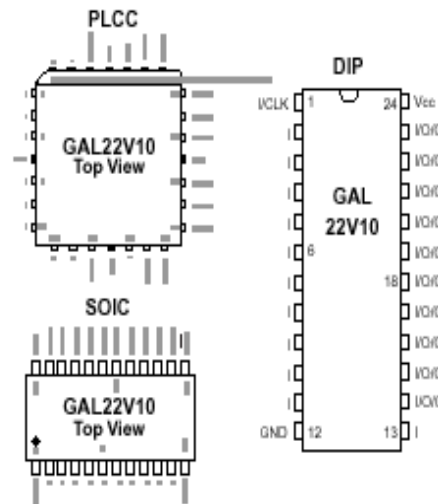
Features

- HIGH PERFORMANCE E²CMOS™ TECHNOLOGY
 - 4 ns Maximum Propagation Delay
 - Fmax = 250 MHz
 - 3.5 ns Maximum from Clock Input to Data Output
 - UltraMOS™ Advanced CMOS Technology
- ACTIVE PULL-UPS ON ALL PINS
- COMPATIBLE WITH STANDARD 22V10 DEVICES
 - Fully Function/Fuse-Map/Parametric Compatible with Bipolar and UVCMS 22V10 Devices
- 50% to 75% REDUCTION IN POWER VERSUS BIPOLAR
 - 90mA Typical Icc on Low Power Device
 - 45mA Typical Icc on Quarter Power Device
- E² CELL TECHNOLOGY
 - Reconfigurable Logic
 - Reprogrammable Cells
 - 100% Tested/100% Yields
 - High Speed Electrical Erasure (<100ms)
 - 20 Year Data Retention
- TEN OUTPUT LOGIC MACROCELLS
 - Maximum Flexibility for Complex Logic Designs
- PRELOAD AND POWER-ON RESET OF REGISTERS
 - 100% Functional Testability
- APPLICATIONS INCLUDE:
 - DMA Control
 - State Machine Control
 - High Speed Graphics Processing
 - Standard Logic Speed Upgrade
- ELECTRONIC SIGNATURE FOR IDENTIFICATION
- LEAD-FREE PACKAGE OPTIONS

Functional Block Diagram



Pin Configuration



Description

The GAL22V10, at 4ns maximum propagation delay time, combines a high performance CMOS process with Electrically Erasable (E²) floating gate technology to provide the highest performance available of any 22V10 device on the market. CMOS circuitry allows the GAL22V10 to consume much less power when compared to bipolar 22V10 devices. E² technology offers high speed (<100ms) erase times, providing the ability to reprogram or reconfigure the device quickly and efficiently.

The generic architecture provides maximum design flexibility by allowing the Output Logic Macrocell (OLMC) to be configured by the user. The GAL22V10 is fully function/fuse map/parametric compatible with standard bipolar and CMOS 22V10 devices.

Unique test circuitry and reprogrammable cells allow complete AC, DC, and functional testing during manufacture. As a result, Lattice Semiconductor delivers 100% field programmability and functionality of all GAL products. In addition, 100 erase/write cycles and data retention in excess of 20 years are specified.

Copyright © 2004 Lattice Semiconductor Corp. All brand or product names are trademarks or registered trademarks of their respective holders. The specifications and information herein are subject to change without notice.

LATTICE SEMICONDUCTOR CORP., 5555 Northeast Moore Ct., Hillsboro, Oregon 97124, U.S.A.
Tel. (503) 268-8000; 1-800-LATTICE; FAX (503) 268-8566; <http://www.latticesemi.com>

August 2004

Anexo C. Hojas de datos del MAX233

19-4323, Rev 14, 6/04

MAXIM

+5V-Powered, Multichannel RS-232 Drivers/Receivers

MAX220-MAX249

General Description

The MAX220-MAX249 family of line drivers/receivers is intended for all EIA/TIA-232E and V.28/V.24 communications interfaces, particularly applications where $\pm 12V$ is not available.

These parts are especially useful in battery-powered systems, since their low-power shutdown mode reduces power dissipation to less than $5\mu W$. The MAX225, MAX233, MAX235, and MAX245/MAX246/MAX247 use no external components and are recommended for applications where printed circuit board space is critical.

Applications

Portable Computers
Low-Power Modems
Interface Translation
Battery-Powered RS-232 Systems
Multidrop RS-232 Networks

AutoShutdown and UCSP are trademarks of Maxim Integrated Products, Inc.

Next-Generation Device Features

- ♦ For Low-Voltage, Integrated ESD Applications
MAX3222E/MAX3232E/MAX3237E/MAX3241E/
MAX3246E: +3.0V to +5.5V, Low-Power, Up to
1Mbps, True RS-232 Transceivers Using Four
0.1 μF External Capacitors (MAX3246E Available
in a UCSP™ Package)
- ♦ For Low-Cost Applications
MAX221E: $\pm 15kV$ ESD-Protected, +5V, 1 μA ,
Single RS-232 Transceiver with AutoShutdown™

Ordering Information

PART	TEMP RANGE	PIN-PACKAGE
MAX220CPE	0°C to +70°C	16 Plastic DIP
MAX220CSE	0°C to +70°C	16 Narrow SO
MAX220CWE	0°C to +70°C	16 Wide SO
MAX220CD	0°C to +70°C	Dice*
MAX220EPE	-40°C to +85°C	16 Plastic DIP
MAX220ESE	-40°C to +85°C	16 Narrow SO
MAX220EWE	-40°C to +85°C	16 Wide SO
MAX220EJE	-40°C to +85°C	16 CERDIP
MAX220MJE	-55°C to +125°C	16 CERDIP

Ordering information continued at end of data sheet.

*Contact factory for dice specifications.

Selection Table

Part Number	Power Supply (V)	No. of RS-232 Drivers/Rx	No. of Ext. Caps	Nominal Cap. Value (μF)	SHDN & Thres- State	Rx Active in SHDN	Data Rate (kbps)	Features
MAX220	+5	2/2	4	0.047/0.33	No	—	120	Ultra-low-power, industry-standard pinout
MAX222	+5	2/2	4	0.1	Yes	—	200	Low-power shutdown
MAX223 (MAX213)	+5	4/5	4	1.0 (0.1)	Yes	✓	120	MAX241 and receivers active in shutdown
MAX225	+5	5/5	0	—	Yes	✓	120	Available in SO
MAX230 (MAX200)	+5	5/0	4	1.0 (0.1)	Yes	—	120	5 drivers with shutdown
MAX231 (MAX201)	+5 and +7.5 to +13.2	2/2	2	1.0 (0.1)	No	—	120	Standard +5/+12V or battery supplies; same functions as MAX232
MAX232 (MAX202)	+5	2/2	4	1.0 (0.1)	No	—	120 (64)	Industry standard
MAX232A	+5	2/2	4	0.1	No	—	200	Higher slew rate, small caps
MAX233 (MAX203)	+5	2/2	0	—	No	—	120	No external caps
MAX233A	+5	2/2	0	—	No	—	200	No external caps, high slew rate
MAX234 (MAX204)	+5	4/0	4	1.0 (0.1)	No	—	120	Replaces 1488
MAX235 (MAX205)	+5	5/5	0	—	Yes	—	120	No external caps
MAX236 (MAX206)	+5	4/3	4	1.0 (0.1)	Yes	—	120	Shutdown, three state
MAX237 (MAX207)	+5	5/3	4	1.0 (0.1)	No	—	120	Complements IBM PC serial port
MAX238 (MAX208)	+5	4/4	4	1.0 (0.1)	No	—	120	Replaces 1488 and 1489
MAX239 (MAX209)	+5 and +7.5 to +13.2	3/5	2	1.0 (0.1)	No	—	120	Standard +5/+12V or battery supplies; single-package solution for IBM PC serial port
MAX240	+5	5/5	4	1.0	Yes	—	120	DIP or flatpack package
MAX241 (MAX211)	+5	4/5	4	1.0 (0.1)	Yes	—	120	Complete IBM PC serial port
MAX242	+5	2/2	4	0.1	Yes	✓	200	Separate shutdown and enable
MAX243	+5	2/2	4	0.1	No	—	200	Open-line detection simplifies cabling
MAX244	+5	8/10	4	1.0	No	—	120	High slew rate
MAX245	+5	8/10	0	—	Yes	✓	120	High slew rate, int. caps, two shutdown modes
MAX246	+5	8/10	0	—	Yes	✓	120	High slew rate, int. caps, three shutdown modes
MAX247	+5	8/9	0	—	Yes	✓	120	High slew rate, int. caps, nine operating modes
MAX248	+5	8/8	4	1.0	Yes	✓	120	High slew rate, selective half-chip enables
MAX249	+5	6/10	4	1.0	Yes	✓	120	Available in quad flatpack package

MAXIM

Maxim Integrated Products 1

For pricing, delivery, and ordering information, please contact Maxim/Dallas Direct! at 1-888-629-4642, or visit Maxim's website at www.maxim-ic.com.

+5V-Powered, Multichannel RS-232 Drivers/Receivers

MAX220-MAX249

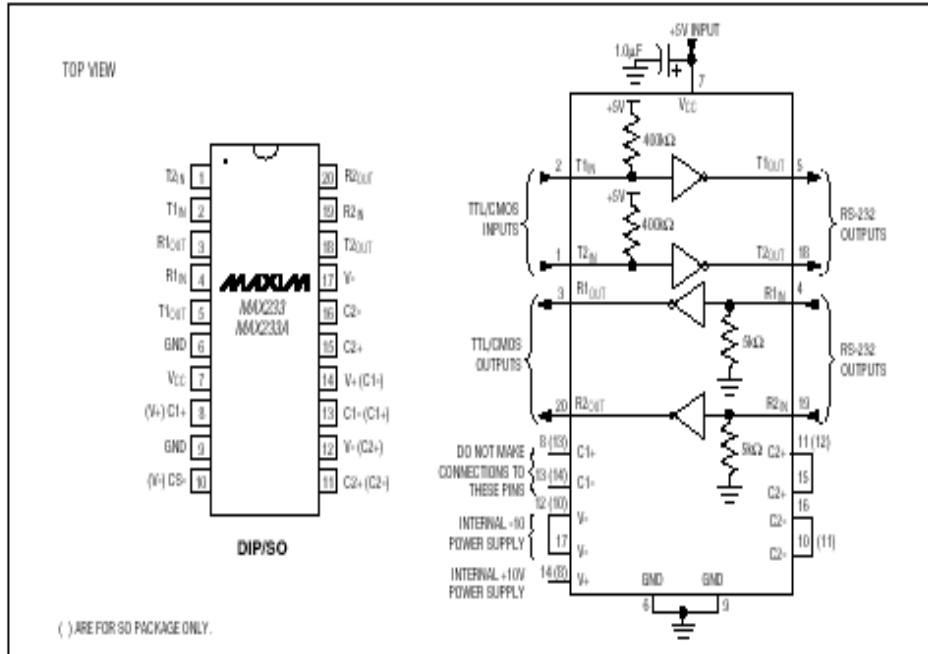


Figure 11. MAX233/MAX233A Pin Configuration and Typical Operating Circuit

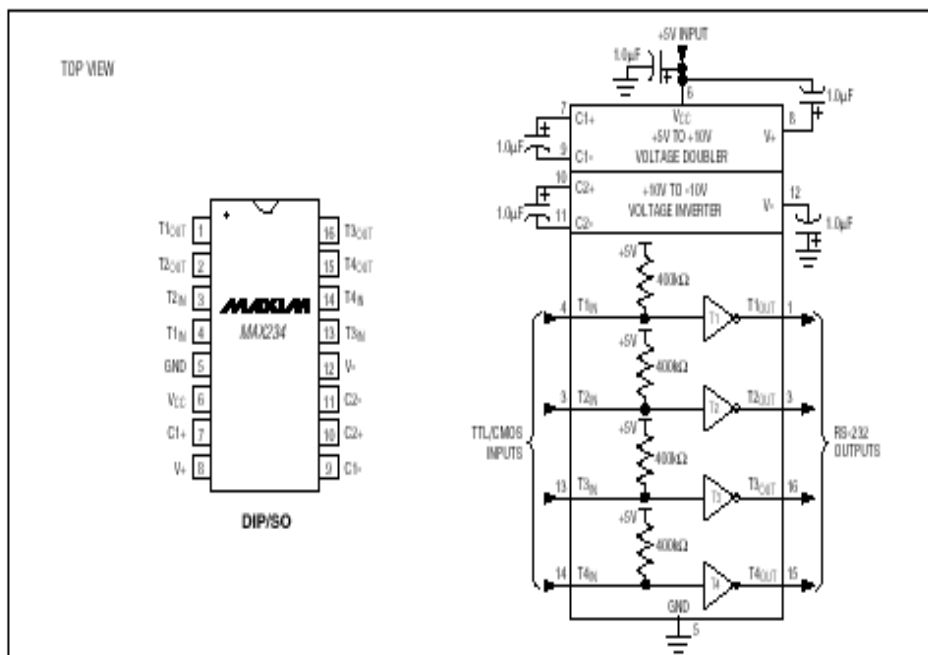


Figure 12. MAX234 Pin Configuration and Typical Operating Circuit

Anexo D. Hojas de datos del microcontrolador PIC16F877A



PIC16F87X

28/40-Pin 8-Bit CMOS FLASH Microcontrollers

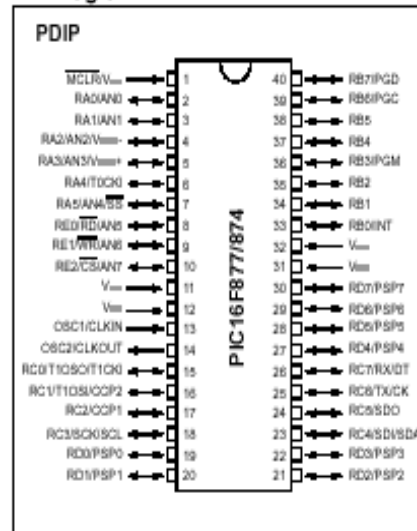
Devices Included in this Data Sheet:

- PIC16F873
- PIC16F876
- PIC16F874
- PIC16F877

Microcontroller Core Features:

- High performance RISC CPU
- Only 35 single word instructions to learn
- All single cycle instructions except for program branches which are two cycle
- Operating speed: DC - 20 MHz clock input
DC - 200 ns instruction cycle
- Up to 8K x 14 words of FLASH Program Memory
Up to 368 x 8 bytes of Data Memory (RAM)
Up to 256 x 8 bytes of EEPROM Data Memory
- Pinout compatible to the PIC16C73B/74B/76/77
- Interrupt capability (up to 14 sources)
- Eight level deep hardware stack
- Direct, indirect and relative addressing modes
- Power-on Reset (POR)
- Power-up Timer (PWRT) and Oscillator Start-up Timer (OST)
- Watchdog Timer (WDT) with its own on-chip RC oscillator for reliable operation
- Programmable code protection
- Power saving SLEEP mode
- Selectable oscillator options
- Low power, high speed CMOS FLASH/EEPROM technology
- Fully static design
- In-Circuit Serial Programming™ (ICSP) via two pins
- Single 5V In-Circuit Serial Programming capability
- In-Circuit Debugging via two pins
- Processor read/write access to program memory
- Wide operating voltage range: 2.0V to 5.5V
- High Sink/Source Current: 25 mA
- Commercial, Industrial and Extended temperature ranges
- Low-power consumption:
 - < 0.6 mA typical @ 3V, 4 MHz
 - 20 µA typical @ 3V, 32 kHz
 - < 1 µA typical standby current

Pin Diagram



Peripheral Features:

- Timer0: 8-bit timer/counter with 8-bit prescaler
- Timer1: 16-bit timer/counter with prescaler, can be incremented during SLEEP via external crystal/clock
- Timer2: 8-bit timer/counter with 8-bit period register, prescaler and postscaler
- Two Capture, Compare, PWM modules
 - Capture is 16-bit, max. resolution is 12.5 ns
 - Compare is 16-bit, max. resolution is 200 ns
 - PWM max. resolution is 10-bit
- 10-bit multi-channel Analog-to-Digital converter
- Synchronous Serial Port (SSP) with SPI™ (Master mode) and I²C™ (Master/Slave)
- Universal Synchronous Asynchronous Receiver Transmitter (USART/SCI) with 9-bit address detection
- Parallel Slave Port (PSP) 8-bits wide, with external \overline{RD} , \overline{WR} and \overline{CS} controls (40/44-pin only)
- Brown-out detection circuitry for Brown-out Reset (BOR)