



UNIVERSIDAD DE CARABOBO
FACULTAD DE INGENIERÍA
ESCUELA DE INGENIERÍA ELÉCTRICA
DEPARTAMENTO DE SISTEMAS Y AUTOMÁTICA
PROYECTO DE TRABAJO ESPECIAL DE GRADO



Diseño y elaboración de un Teleprompter portátil de bajo costo y con comunicación inalámbrica entre una pantalla LCD y un teléfono móvil inteligente mediante la utilización de Bluetooth y de la plataforma de Android.

Rosmar Castillo
C.I. 18.445.355

Tutor: Wilmer Sanz
Naguanagua; 29 de Mayo del 2013



UNIVERSIDAD DE CARABOBO
FACULTAD DE INGENIERÍA
ESCUELA DE INGENIERÍA ELÉCTRICA
DEPARTAMENTO DE SISTEMAS Y AUTOMÁTICA
PROYECTO DE TRABAJO ESPECIAL DE GRADO



Diseño y elaboración de un Teleprompter portátil de bajo costo y con comunicación inalámbrica entre una pantalla LCD y un teléfono móvil inteligente mediante la utilización de Bluetooth y de la plataforma de Android.

Trabajo Especial de Grado presentado ante la ilustre Universidad de Carabobo para optar al
Título de Ingeniero Electricista

Rosmar Castillo
C.I. 18.445.355

Tutor: Wilmer Sanz
Naguanagua; 29 de Mayo del 2013



DEDICATORIA

Con el transcurrir de nuestras vidas nos encontramos con momentos difíciles de superar; aquellos momentos en los que sentimos que ya no podemos, pero la constancia y el entusiasmo por lograr las metas propuestas, así como la motivación de nuestros seres amados, nos da la fuerza necesaria para seguir adelante. Especialmente mi dedicatoria va dirigida a:

Dios principalmente por permitirme la vida, la familia más hermosa y la salud necesaria para seguir adelante en el logro de mis metas y objetivos.

A Rosnel´s Malavé de Castillo, mi Madre, el ser más importante de mi vida a quien adoro y debo todos y cada uno de mis logros, ya que ella es el ejemplo más grande de coraje, valentía e inteligencia; pero por sobre todas las cosas es mi más grande motivación.

A Angel Omar Castillo, mi Padre, uno de los seres que más adoro, por ser mi guía en cada uno de mis pasos y aunque el ya no esté conmigo físicamente está presente cada día de mi vida en mis pensamientos y en mi corazón. Papi te amo, esta Tesis es por ti y para ti.

A Nelson Malavé, mi Abuelo, por ser mi segundo padre y haberme acompañado y guiado en cada momento de mi vida. Aunque ya no estés conmigo abuelito, esta tesis es para ti. Te quiero muchísimo...

A Rosangel`s Castillo, mi Hermana, por creer en mí en todo momento y apoyarme siempre a lo largo de mi vida. Te quiero muchísimo...

A Angélica Castillo, mi Hermana menor, por impulsarme a seguir adelante y por estar siempre a mi lado en las buenas y en las malas. Te quiero muchísimo...



AGRADECIMIENTO

A Dios por darme las herramientas más hermosas para lograr cada una de mis metas, como lo es principalmente mi familia, así como también por estar a mi lado siempre guiando cada uno de mis pasos.

A mi abuela Rosa de Malavé, por brindarme su apoyo y ayudarme a enfrentar cada una de mis metas.

A la Universidad de Carabobo por impartir los conocimientos dejando una huella de excelencia en mi vida personal y en el desarrollo de mis estudios.

A mi prima Yhannel`s Limongi, porque siempre me ha brindado su apoyo y por estar siempre que la necesito. Te quiero muchísimo...

A la Empresa Hera Audiovisual C.A. y todos los que allí laboran, gracias por su colaboración en el desarrollo de este trabajo.

A Horacio Loyola, por creer en mí y ser parte de mi vida. Te quiero mucho.

A mí Tutor Wilmer Sanz, por compartir su tiempo y sus conocimientos para permitir culminar este episodio en mi carrera.

A mis amigos Carla, Embert, Jonas y especialmente a Adolfo, que sin ustedes a mi lado no lo hubiera logrado les agradezco a todos con toda mi alma el haber llegado a mi vida y el compartir momentos agradables y tristes, por hacer más fácil este reto, por sus entusiasmos, apoyos, comprensión y alegrías. Los quiero mucho.

A TODOS MIL GRACIAS!!!



Diseño y elaboración de un Teleprompter portátil de bajo costo y con comunicación inalámbrica entre una pantalla LCD y un teléfono móvil inteligente mediante la utilización de Bluetooth y de la plataforma de Android.

RESUMEN

Esta investigación tiene como objetivo principal realizar un Teleprompter portátil para la Empresa HERA Audiovisual, de manera que no necesite una computadora para enviar el texto, sino que mediante un teléfono móvil inteligente con plataforma Android se pueda enviar la información a la pantalla por medio de comunicación Bluetooth y sea de bajo costo para facilitar la adquisición del mismo, que se ajuste a la complejidad de los procesos y permita el mejoramiento continuo. Se realizó un análisis situacional interno para conocer con exactitud los equipos que poseen la empresa y las necesidades operacionales de ésta, utilizando técnicas de recolección de datos como el cuestionario y la observación directa. Se trata de un proyecto experimental y factible dado que se busca generar un producto que puede ser comercializado y aplicado por empresas audiovisuales alrededor del mundo.

En el contenido del tomo se organiza en 5 capítulos donde se explica detalladamente la forma en la cual trabaja el equipo diseñado, selección de componentes y detalles de programación; todo con sus respectivos planos y diagramas de flujos.

Palabras Claves: Teleprompter, Bluetooth, plataforma Android.



ÍNDICE GENERAL

DEDICATORIA	i
AGRADECIMIENTO	ii
RESUMEN	iii
ÍNDICE DE FIGURAS	vii
ÍNDICE DE TABLAS	ix
ÍNDICE DE CUADROS	ix
ÍNDICE DE GRÁFICAS	ix
INTRODUCCIÓN	1
CAPÍTULO I	3
EL PROBLEMA.....	3
1.1 Planteamiento del Problema.....	3
1.2 Justificación de la Investigación.....	4
1.3 Formulación de los Objetivos	8
1.3.1 Objetivos generales.....	8
1.3.2 Objetivos Específicos.....	8
1.4 Alcance.....	9
CAPÍTULO II.....	10



MARCO TEÓRICO.....	10
2.1 Antecedentes Históricos	10
2.2 Marco Conceptual	12
CAPÍTULO III:	21
MARCO METODOLÓGICO.....	21
3.1 Clasificación de la investigación.....	21
3.2 Metodología de diseño	21
3.3 Desarrollo de las fases metodológicas.	23
CAPÍTULO IV:	27
ANÁLISIS DE RESULTADOS.....	27
4.1 UNIDAD I: Diagnóstico de los equipos o medios de comunicación implementados actualmente dentro de la empresa caso estudio, y en otras empresas de situación similar.....	27
4.2 UNIDAD II: Recopilación de toda la información posible sobre la comunicación inalámbrica por medio de Bluetooth	37
4.3 UNIDAD III: Análisis de la plataforma de Android y el lenguaje de alto nivel para la programación del microcontrolador.	54
4.3.1 Plataforma de Android	54
4.3.2 Lenguaje de Alto Nivel	59



4.4 UNIDAD IV: Creación de un software utilizando la plataforma Android	59
4.5 UNIDAD V: Elaboración de un software para el microcontrolador que reciba por medio del módulo Bluetooth la información que quiere ser visualizada en la pantalla LCD.....	65
Figura 4.29 Diagrama de subrutina de retardo.	73
4.6 UNIDAD VI: Ensamble del proyecto final mediante la conexión de la pantalla LCD, el módulo Bluetooth y el microcontrolador en un solo circuito impreso.	74
4.6.1 Circuito Integrado	74
4.6.2 Diseño de la carcasa y soporte a la cámara del Teleprompter Portátil	76
CONCLUSIONES Y RECOMENDACIONES	79
CONCLUSIONES	79
RECOMENDACIONES	80
REFERENCIAS BIBLIOGRÁFICAS	81
ANEXOS	84
ANEXO NRO. 1. Instrumento de Levantamiento de Información.	84
ANEXO NRO. 2. Data Sheet de los principales componentes del sistema.....	88
ANEXO NRO. 3. Imágenes del prototipo y sus partes	112
APÉNDICE	118
APÉNDICE A: CÓDIGO FUENTE	119



ÍNDICE DE FIGURAS

Figura 2.1 Piconet.....	16
Figura 2.2 Scatternet con dos piconets.....	17
Figura 4.1 Esquema general del Teleprompter portátil.....	36
Figura 4.2 Esquema de comunicación Maestro-Eslavo. [6].....	38
Figura 4.3 Pila de Protocolo Bluetooth.....	38
Figura 4.4 Datagrama Bluetooth [14].....	43
Figura 4.5 Estructura de la trama de Bluetooth. [15].....	44
Figura 4.6 Módulo Bluetooth RN-42 Clase 2.....	45
Figura 4.7 Pines de descripción RN-42.....	47
Figura 4.8 Circuito de conexión para el RN-42.....	48
Figura 4.9 Interfaz del BlueTerm.....	50
Figura 4.10 Configuración de los comandos del módulo bluetooth.....	51
Figura 4.11 Conexión del MAX3232.....	52
Figura 4.12 USART Terminal de Mikrobasic Pro.....	53
Figura 4.13 Estadísticas de utilización de las distintas versiones de Android. [13].....	54
Figura 4.14 Menú de ayuda.....	57
Figura 4.15 Integración de Android con Eclipse.....	58



Figura 4.16 Colocación de la ubicación del SDK.....	59
Figura 4.17 Interfaz Gráfica de la aplicación en Android.....	60
Figura 4.18 Diagrama para escribir el texto.....	62
Figura 4.19 Diagrama para cargar el texto.....	64
Figura 4.20 Pines del PIC18F2525.....	66
Figura 4.21 Diagrama de bloques del sistema en general.....	66
Figura 4.22 Circuito general del sistema.....	67
Figura 4.23 Módulo de visualización.....	68
Figura 4.24 Módulo de Control.....	69
Figura 4.25 Módulo Bluetooth RN-42.....	70
Figura 4.26 Módulo Bluetooth RN-42 en Proteus.....	70
Figura 4.27 Diagrama de flujo general del microcontrolador.....	71
Figura 4.28 Diagrama configuración de GLCD y Bluetooth.....	72
Figura 4.29 Diagrama de subrutina de retardo.....	73
Figura 4.30 Diseño de la placa del circuito integrado en tercera dimensión.....	74
Figura 4.31 Imagen Stilk Screen del circuito impreso.....	75
Figura 4.32 Diseño de la PCB.....	75
Figura 4.33 Soporte del Teleprompter Portátil.....	76
Figura 4.34 Carcasa del Teleprompter Portátil.....	77
Figura 4.35 Partes del soporte y carcasa del Teleprompter Portátil.....	78



ÍNDICE DE TABLAS

Tabla 4.1 Potencias según Clase en la tecnología Bluetooth.....	44
Tabla 4.2 Microcontroladores PIC18F2525/2620/4525/4620.....	65

ÍNDICE DE CUADROS

Cuadro 1.1 Definición conceptual de las dimensiones de impacto de este Proyecto de Grado.....	6
Cuadro 2.1 Rango de frecuencia.....	14
Cuadro 2.2 Materiales que provocan interferencia en las señales inalámbricas.....	15
Cuadro 2.3 Cuadro resume de características de Bluetooth.....	18

ÍNDICE DE GRÁFICAS

Gráfico No 4.1 Cantidad de empleados.....	29
Gráfico No 4.2 Cantidad de clientes regulares mensuales.....	29
Gráfico No 4.3 Que implica grabaciones en exteriores.....	30
Gráfico No 4.4 Cantidad de equipos en grabaciones en exteriores.....	31
Gráfico No 4.5 Utilidad del Teleprompter en exteriores.....	31
Gráfico No 4.6 Porque es útil el uso de Teleprompter en exteriores.....	32
Gráfico No 4.7 Porque la falta de Teleprompter en las empresas audiovisuales.....	33
Gráfico No 4.8 Métodos utilizados para solventar la falta de Teleprompter.....	33
Gráfico No 4.9 Distancia entre el host y la cámara.....	34
Gráfico No 4.10 Características de un Teleprompter portátil.....	35



INTRODUCCIÓN

El entorno de las productoras audiovisuales se ha vuelto significativamente más competitivo, por ello las empresas deben estar preparadas para enfrentarse a los cambios constantes del mercado; en consecuencia han surgido varios equipos tecnológicos, cuyos propósitos es actualizar los productos que ofrecen de manera que les permitan alcanzar el éxito.

En este sentido se puede mencionar como uno de estos equipos a el Teleprompter, el cual es un aparato electrónico que refleja el texto de una noticia previamente cargado en una computadora, cuya principal función es brindar una lectura adecuada de un texto sin tener que memorizarlo, pero su implementación es mayormente en estudios (como en espacios informativos de televisión) debido a la cantidad de equipos que se necesitan, por lo que su aplicación fuera del mismo (como en programas o documentales seguidos por un guion) es muy escasa, además de ser muy costoso.

Este Trabajo Especial de Grado propone las bases para la implementación de un Teleprompter portátil de bajo costo, fácil manejo y de pocos equipos, haciendo uso de un teléfono móvil inteligente con plataforma Android como dispositivo de almacenamiento y envío de la información que se desea leer en la pantalla, llevando a cabo la construcción de un prototipo del equipo en cuestión.

Para el desarrollo del Teleprompter portátil, se analizó en detalle las diferentes necesidades en cuanto a la comunicación entre el director y el o la host (presentador o presentadora) en exteriores que presenta la Empresa Audiovisual caso estudio y otras empresas de funciones similares resaltando las necesidades principales que puedan ser de interés para la creación de un mejor equipo. Permitiendo de esta forma diseñar un Teleprompter portátil que me permita enviar un texto previamente cargado en el teléfono móvil inteligente y a la vez me



permita enviar en tiempo real una idea, sugerencia o comentario que el director quiera hacerle llegar al host o la host sin tener que detener las grabaciones.

Este Trabajo Especial de Grado fue estructurado en cinco (05) capítulos.

En el capítulo I, se presenta el planteamiento del problema, los objetivos generales y específicos, delimitaciones y justificación e impacto de la investigación, con el propósito de identificar la problemática a estudiar.

En el capítulo II, se encuentra el marco teórico, donde queda de manifiesto los antecedentes de la investigación, las bases teóricas y la definición de términos básicos del presente Trabajo Especial de Grado.

En el capítulo III, se encuentra el marco metodológico de la investigación, definiendo el tipo de investigación, el diseño de las fases metodológicas, la unidad de estudio y de análisis, y las técnicas de recolección, análisis, interpretación y presentación de la información.

El capítulo IV muestra los resultados obtenidos en el Trabajo Especial de Grado, en el cual se describe el diseño y elaboración del Teleprompter portátil.

Por último, en el capítulo V se presenta las conclusiones y recomendaciones de la investigación.



CAPÍTULO I

EL PROBLEMA

1.1 Planteamiento del Problema

HERA Audiovisual es una empresa ubicada en la urbanización Palo Verde, Estado Miranda, que se especializa en las áreas de documentales, micro documentales, micros informativos, spots publicitarios, programas de televisión, cortometrajes, infomerciales, making-of de publicidad o eventos, diseño y producción; con amplia experiencia y dominio técnico del oficio. Por todo lo antes mencionado, cualquier elemento tecnológico que aporte comodidad y calidad al trabajo es siempre puesto al servicio de las realizaciones.

Uno de los equipos más utilizados en empresas audiovisuales con funciones similares a las de HERA Audiovisual, son los conocidos con el nombre de Teleprompter, los cuales tienen como función servir de apuntador y permitir al conductor leer el texto sin necesidad de memorizarlo ni desviar la mirada de la cámara. Sin embargo, los Teleprompter en su mayoría están diseñados para trabajar en estudio ya que son muy pesados y de difícil traslado.

Por otro lado, para la realización de trabajos de campo durante las grabaciones de documentales, se suelen utilizar la menor cantidad de equipos posibles, debido a que a veces son largas las horas del día y kilómetros caminando, las que pasan con sus equipamientos.

Actualmente en el mercado, existen algunos modelos de Teleprompter cuyo uso es en exteriores, pero la realidad es que los costos oscilan entre 800 y 1600 euros [1], lo que no permiten que este sea un accesorio de fácil adquisición para la empresa caso estudio. Por esta razón, la misma ha tenido que solventar la falta del aparato de maneras poco prácticas como por ejemplo, implementando una lámina de papel o pizarra con el texto escrito



obligando que el lector desvíe la mirada para leer la información, lo cual causa que sus trabajos no sean tan eficientes.

Por todo lo antes mencionado, se propone diseñar y elaborar un Teleprompter portátil que cumpla los altos estándares necesarios, que no necesite de una computadora para enviar el texto sino que mediante un teléfono inteligente con plataforma Android se pueda enviar la información a la pantalla por medio de comunicación Bluetooth y sea de bajo costo para facilitar la adquisición del mismo. Se busca de esta manera obtener una herramienta que sea accesible, autodidacta, sencilla de usar, y de fácil disponibilidad y traslado, que le permita a los usuarios una mayor aplicación sin necesidad de más equipos que los necesarios como lo son un teléfono y la pantalla LCD, y de esta forma, contribuir a su desarrollo productivo y rentable y evitar así el fracaso de estas en el sector en el que se desenvuelvan.

Es necesario entonces, que este aparato, tome las ideas de los Teleprompter de estudios, que logran reflejar los textos muy cerca del eje de mirada entre el individuo y el objetivo (lente) de la cámara, sin generar en el individuo desviar la mirada del lente, con lo que se perdería la interacción real con el telespectador. Además, de que sea de fácil traslado para evitar que los empleados de la empresa tengan que cargar más aparatos de los necesarios.

1.2 Justificación de la Investigación

La manera en que se enfoca la investigación en el presente Proyecto de Grado proviene de la necesidad inherente de cada empresario de desarrollar su empresa al punto de hacerla más competitiva en el sector al que pertenece. Y es mediante la organización y calidad de su equipo de trabajo, tanto personal como tecnológico, que las empresas audiovisuales pueden alcanzar una alta rentabilidad en sus mercados y volverse un competidor más fuerte.

En un entorno económico cada vez más competitivo, como el actual, las empresas audiovisuales necesitan disponer de equipos tecnológicos de punta y de fácil manejo que constituyan un instrumento útil para controlar su eficiencia y que proporcionen un alto



grado de viabilidad de las distintas actividades que se realizan en sus procesos para servir de apoyo en cada uno de estos.

Sin embargo, muchas empresas audiovisuales han fracasado o no han obtenido los resultados deseados por la mala calidad de sus productos audiovisuales, al no contar con los equipos básicos para la elaboración de sus productos finales o prestación de sus servicios.

A su vez, existen equipos que pueden ayudar a las empresas audiovisuales a mejorar la calidad de sus productos, muchos de estos pueden ser muy costosos, por lo que la mayoría de las empresas tienden a prescindir de los mismos. A pesar de esto, es posible implantar equipos como el Teleprompter portátil de una manera más didáctica y fácil de manejar, haciendo que las empresas consigan un mayor éxito, y el sector sea respaldo de crecimiento y desarrollo económico en el país. Esto, mediante un equipo audiovisual accesible y disponible en Venezuela, sin necesidad de adquirirlo en otros países y a muy bajo costo.

HERA Audiovisual es una de las empresas del mercado que ha manifestado la necesidad de contar con un Teleprompter, de fácil manejo, de buena resolución, y con herramientas que den soluciones y no problemas durante las grabaciones.



Cuadro 1.1

Definición conceptual de las dimensiones de impacto de este Proyecto de Grado.

Dimensión	Descripción
Innovación- Tecnología	El diseño del Teleprompter contará con una pantalla gráfica LCD de 4x5” pulgadas, la cual se colocará muy cerca del lente de la cámara y de esta manera el lector no tiene que desviar la mirada para leer; el texto será enviado desde un teléfono móvil inteligente con conexión Bluetooth para acceder de forma inalámbrica al software y darle la cadencia necesaria al ritmo de lectura del individuo, en el momento de ser necesario.
Financiero	Generará un alto impacto en el mercado ya que pondrá al alcance de las empresas audiovisuales un equipo que, sus equivalentes, actualmente, y de forma convencional, son sumamente costosos, y requieren grandes inversiones en tecnología, licencias, entre otros.
Prestigio	La elaboración de un Teleprompter portátil cuya información sea enviada mediante comunicación Bluetooth desde un teléfono inteligente con plataforma Android se convertirá en una de las pocas investigaciones que combinan uno de los sistemas operativos móviles más usado en la actualidad en los teléfonos inteligentes y la Red de Comunicación Inalámbricas de Área Personal Bluetooth, a nivel de pregrado; generando nuevos horizontes de investigación, y contribuyendo con la generación de conocimientos en áreas poco convencionales dentro de la Universidad.



Continuación. Cuadro 1.1

Definición conceptual de las dimensiones de impacto de este Proyecto de Grado.

Visión Fronteras Abiertas	La aplicación de este proyecto puede realizarse a lo largo del país, y su implementación puede darse en diversas instituciones y diversas empresas.
Utilidad y Aplicabilidad	Partiendo de la base del desarrollo de este proyecto, el mismo puede ser implementado directamente en cualquier empresa audiovisual, sin mayores inversiones, o sin mayor necesidad de recursos, permitiendo la obtención de beneficios, tangibles e intangibles, fundamentalmente económicos, ya que el software se podrá adquirir como una aplicación más para el teléfono móvil inteligente.
Ventajas	La principal ventaja radica en que ya no será necesario el uso de una computadora para enviar el texto que se desea mostrar a la pantalla, en cambio sólo con el uso de un teléfono móvil inteligente podrá realizarse todo el proceso, haciendo de este un equipo más fácil de transportar y manejar.
Desventajas	La mayor desventaja que presenta este proyecto es que el teléfono móvil inteligente debe contar con la plataforma Android, de lo contrario la aplicación no funcionará ya que el mismo será realizado sólo para teléfonos que posean dicha plataforma; además debe tener comunicación Bluetooth, ya que será la única forma de transmitir la información.



1.3 Formulación de los Objetivos

1.3.1 Objetivos generales

Elaborar un Teleprompter portátil de bajo costo y con comunicación inalámbrica entre una pantalla LCD y un teléfono móvil inteligente mediante la utilización de Bluetooth y de la plataforma de Android.

1.3.2 Objetivos Específicos

- Realizar un diagnóstico de los equipos o métodos de comunicación implementados actualmente dentro de la empresa caso estudio, y en otras empresas de situación similar, para la identificación los factores claves que servirán para el desarrollo de la herramienta.
- Recopilar toda la información posible sobre la comunicación inalámbrica por medio de Bluetooth, para una mejor comprensión del tema y de esta forma poder seleccionar el módulo más adecuado e implementarlo sin mayor complicación.
- Analizar la plataforma de Android, para entender con mayor claridad la forma de realizar aplicaciones en dicha plataforma y el lenguaje de alto nivel para la programación del microcontrolador que recibirá la información que se desea ver en la pantalla LCD.
- Crear un software utilizando la plataforma Android que permita enviar la información mediante la conexión Bluetooth desde el teléfono móvil inteligente hasta la pantalla LCD y que permita controlar la velocidad de aparición del mismo para que pueda ser leído con facilidad por el usuario.
- Elaborar un software para el microcontrolador que reciba por medio del módulo Bluetooth la información que quiere ser visualizada en la pantalla LCD.



- Ensamblar el proyecto final mediante la conexión de la pantalla LCD, el módulo Bluetooth y el microcontrolador en un solo circuito impreso.

1.4 Alcance

En este trabajo se plantea el diseño y elaboración de un Teleprompter, cuya utilización está reservada estrictamente para el uso en campo, esto es motivado a que sus dimensiones fueron seleccionadas de acuerdo a la necesidad de fácil traslado, debido a esto su tamaño es reducido y poco eficiente para su utilización en un estudio de grabación.

Las innovaciones presentes en el diseño es la implementación de tecnología inalámbrica a través de la comunicación por medio del sistemas Bluetooth, este es un avance importante al permitir la interconexión entre el Teleprompter y el dispositivo de control que carga la información a mostrar o apuntar.

Como medio de control, seleccionamos teléfonos móviles inteligentes con plataforma de Android, los mismos deben tener un sistema de transmisión Bluetooth para poder acceder al Teleprompter.



CAPÍTULO II

MARCO TEÓRICO

Debido a que el presente Proyecto de Grado está basado en el área de la electrónica, cuyo objetivo fundamental es la implementación de componentes como circuitos integrados, elementos electrónicos (resistencias, inductores, capacitores, entre otros) y dispositivos de entrada/salida (interruptores, LCD, entre otros), es de gran importancia la comprensión de temas específicos, que por su vinculación con el estudio resultaron de gran utilidad para la realización del Proyecto de Grado. Los puntos de interés se describen a continuación.

2.1 Antecedentes Históricos

En toda investigación para desarrollar y realizar una problemática se necesita indagar y sustentarse en trabajos investigativos de otros autores; es decir, realizados previamente y que especifiquen la labor que se desarrolla. En tal sentido, se hizo una revisión de autores, tales como:

Aranaz J [2009], Desarrollo de Aplicaciones para Dispositivos Móviles sobre la plataforma Android. Este trabajo de grado nos permite conocer el sistema Android en cuanto a su arquitectura, componentes básicos y cuál es su comportamiento al ejecutar las aplicaciones, además de sus ventajas con respecto a otros sistemas similares y el entorno en el que se desarrolla, proporcionándonos un mayor conocimiento y comprensión de las características de este sistema y a su vez facilitándonos de esta manera su utilización para la realización de nuestro proyecto. [2]

Rey Lago D [2008], Proteus VSM. Este documento es una guía rápida de uso del Proteus 6 Professional para Windows, donde se muestra la secuencia de comandos y acciones que debe realizar el usuario para la correcta implementación y simulación de circuitos básicos con el microcontrolador PIC16F877 de Microchip. Para facilitar el entendimiento de la



herramienta Proteus, se incluyen las pantallas que muestra este programa durante en las diversas etapas de diseño y simulación [3]. En este sentido, la investigación tomada en consideración fue realizada con el objetivo de conocer e implementar el simulador en el presente proyecto y de esta manera poder comprobar el buen funcionamiento del mismo sin tener que ensamblarlo cada vez que se necesite probar sino solo cuando funcione correctamente en el simulador.

Pérez G [2008], Aprendiendo Java y Programación Orientada a Objetos. Este trabajo da a conocer la codificación inicial y estructura de datos, métodos estáticos y mecanismos de programación, además de los primeros objetos como mecanismo de programación de Java de una manera detallada, en este sentido, nos ayuda conocer sobre su utilización e implementación del mismo, facilitándonos de esta forma la realización de la aplicación en Android para transmitir datos sin mayor complicación. [4]

Bellido F, Torres M [2004], Comunicación Inalámbrica con Bluetooth. En este documento se explica el origen, las topologías de conexión, características de la tecnología como tipos de enlaces e inmunidad frente a interferencias así como también las ventajas del Bluetooth frente a otras tecnologías inalámbricas, aportándonos una mayor comprensión sobre el manejo del mismo y estableciéndonos los puntos que se tienen que tener en cuenta para la utilización del mismo en el proyecto. [5]

Alvarez G, Pico J [2004], Desarrollo de un Protocolo para Comunicaciones Inalámbricas e Implementación en Dispositivos de Lógica reconfigurable. En este trabajo se presenta un sistema que tiene una estructura modular que permite una fácil adaptación a los diferentes componentes de comunicación por RF comerciales, mediante el modelo Maestro-Eslavo y la codificación Mánchester, posibilitando la utilización de diferentes bandas de comunicación libres en función de los requerimientos de velocidad de transmisión, alcance y consumo de las aplicaciones en las que se quiera utilizar, de esta forma este trabajo permite analizar y conocer la comunicación por medio del modelo



Maestro-Eslavo y la codificación Manchester que se va a utilizar para la realización del proyecto. [6]

Aguayo P [2004], Introducción al Microcontrolador. En este trabajo se describe la estructura y elementos de los microcontroladores, sus recursos especiales y la familia de los PIC de manera amplia, aportándonos de esta forma un mayor conocimiento sobre los distintos microcontroladores que existen en el mercado y de esta manera poder elegir con mayor facilidad el que se va a implementar en el proyecto de grado. [7]

2.2 Marco Conceptual

➤ **Comunicación Inalámbrica [8]:**

El término inalámbrico se refiere a la conexión de una red por medio de ondas de radio (Wireless), es decir, sin cables que comuniquen los equipos que estarán intercambiando información.

La principal ventaja que supone una red Wireless frente a una de cables, es la movilidad. En síntesis, las redes inalámbricas a diferencia de sus antecesoras son:

- Más simples de instalar.
- Escalables muy fácilmente
- Menos complejas en su administración.

Ahora, en cambio, haremos hincapié en algunas de las desventajas más notorias que acarrea la instalación de una red Wireless. La primera de ellas es la velocidad, ya que hasta el momento las redes Wi-Fi no superan la velocidad de 54 Mbps, mientras que las redes cableadas ya llegaron hace unos cuantos años a los 100 Mbps. Otro punto débil presente en las redes Wireless consiste en su propensión a interferencias, debido al rango de señal en el



cual trabajan (en su mayoría en los 2,4 GHz) suelen ser interferidas por artefactos de uso común en cualquier casa u oficina, como teléfonos inalámbricos, que utilizan ese mismo rango de comunicación.

La transmisión de datos se realiza por medio de un proceso conocido como modulación de la portadora. El aparato transmisor agrega datos a una onda de radio (onda portadora). Esta onda, al llegar al receptor, es analizada por éste, el cual separa los datos útiles de los inútiles.

Las bandas de frecuencia son el resultado de la división del espectro electromagnético, con el objeto de delimitar el acceso de usuarios a determinadas bandas.

En los Estados Unidos y otros países, las bandas de frecuencia son de 900 megahercios (MHz), 2,4 GHz y, en algunos casos, de hasta 5 GHz. Si bien estas bandas de frecuencia no requieren licencia, los equipos que las utilicen deben estar certificados por los reguladores del país donde se encuentren. Los aparatos que no poseen licencia utilizan una potencia baja y su alcance es limitado. Estos dispositivos deben ser muy resistentes a las interferencias, debido al hecho de que no se garantiza que los usuarios posean acceso exclusivo a estas frecuencias sin licencia y, por lo tanto, pueden sufrir intrusiones.

En el cuadro 2.1 se pueden observar los tipos de frecuencias y los márgenes a los que ellos trabajan.



Cuadro 2.1
Rango de frecuencia

DENOMINACIÓN	SIGLAS	MARGEN DE FRECUENCIAS
Frecuencias muy bajas	VLF	3 - 30 KHz
Frecuencias bajas	LF	30 - 300 KHz
Frecuencias medias	MF	300 - 3.000 KHz
Frecuencias altas	HF	3 - 30 MHz
Frecuencias muy altas	VHF	30 - 300 MHz
Frecuencias ultra altas	UHF	300 - 3.000 MHz
Frecuencias super altas	SHF	3 - 30 GHz
Frecuencias extra altas	EHF	30 - 300 GHz

Fuente: Redes Wireless. [8]

Las redes Wireless prevalecen en gran medida ante el problema de la línea de visión, ya que pasan a una frecuencia más alta que otros aparatos en el espectro electromagnético. Estas redes funcionan a unos 2,4 GHz y, en algunos casos, a mayor frecuencia. Aun así, se encuentran muy por debajo del espectro de luz visible. Gracias al uso de esa frecuencia, la longitud de la onda es tan imperceptible que logra traspasar objetos sólidos.

Es por esto que las redes inalámbricas funcionan perfectamente sobre distancias cortas en espacios interiores, aunque en ocasiones algunos obstáculos pueden interferir en la transmisión.

Debido a la naturaleza de la tecnología de radio, las señales de radio frecuencia pueden desvanecerse o bloquearse por la acción de materiales ambientales. La inspección en el lugar nos ayudará a identificar los elementos que afecten en forma negativa a la señal. En el cuadro 2.2, se enumeran los materiales nocivos que debemos considerar con el propósito de realizar una instalación.

Cuadro 2.2
Materiales que provocan interferencia en las señales inalámbricas.

MATERIAL	EJEMPLO	INTERFERENCIA
Madera	Tabiques	Baja
Vidrio	Ventanas	Baja
Amianto	Techos	Baja
Yeso	Paredes interiores	Baja
Ladrillo	Paredes interiores y exteriores	Media
Hojas	Árboles y plantas	Media
Agua	Lluvia / Niebla	Alta
Cerámica	Tejas	Alta
Papel	Rollos de papel	Alta
Vidrio con alto contenido en plomo	Ventanas	Alta
Metales	Vigas, armarios	Muy Alta

Fuente: Redes Wireless.[8]

Bluetooth [9]:

El término Bluetooth se refiere a una especificación abierta para una tecnología que permita comunicación inalámbrica de corto alcance para voz y datos en cualquier parte del mundo. Esta definición incluye ciertos puntos que vale la pena resaltar:

Especificación abierta: La especificación fue desarrollada por un grupo de interés especial (SIG) y está disponible a todo el público sin costo. Los fundadores del SIG fueron Ericsson, Intel, IBM, Nokia y Toshiba luego se incorporaron 3Com, Lucent, Microsoft y Motorola. Actualmente más de 2400 compañías han adoptado la especificación.

Comunicación inalámbrica de corto alcance: Bluetooth pretende reemplazar una variedad de cables de interconexión por ondas de radio con un protocolo común. La tecnología ha sido diseñada específicamente para corto alcance (nominalmente 10 metros)

lo que requiere bajos niveles de potencia y es por tanto adecuado para dispositivos pequeños y portátiles.

Voz y datos: Bluetooth tiene facilidades tanto para comunicaciones de voz como de datos y por tanto permite que todo tipo de dispositivos se comuniquen usando alguno o ambos de estos medios.

En cualquier parte del mundo: Bluetooth usa la banda de 2.4 GHz que es no regulada en casi todos los países.

En una red Bluetooth hasta 8 dispositivos se interconectan para formar una *piconet* (figura 2.1). Uno de estos dispositivos es el maestro y los otros son esclavos. Los dispositivos en una *piconet*, comparten un canal de 1 MHz y se comunican a velocidades de hasta 720 Kbps. El dispositivo maestro determina las características del canal. Los esclavos sólo pueden comunicarse con el maestro y sólo con su permiso.

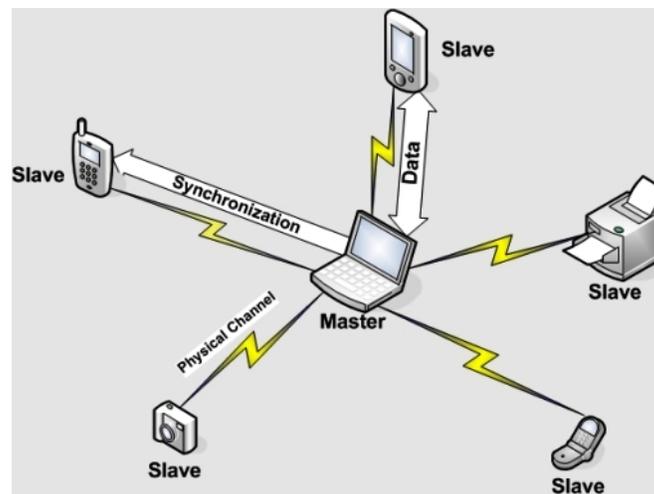


Figura 2.1 Piconet.

Un dispositivo puede pertenecer a más de una *piconet* y actuar como maestro o esclavo en cada una de ellas. Un conjunto de *piconets* forma una *scatternet* (figura 2.2). Este esquema

permite que muchos dispositivos compartan la misma área física y hagan un uso eficiente del ancho de banda. Bluetooth usa hasta 80 frecuencias con un ancho de banda total de 80 MHz. El sistema Bluetooth usa un esquema de salto de frecuencias. Cada *piconet* usa una secuencia de salto de frecuencias diferente, de modo que múltiples canales lógicos pueden compartir el mismo ancho de banda de 80 MHz. Cuando dispositivos en *piconets* distintas, en canales lógicos diferentes usan la misma frecuencia ocurre una colisión.

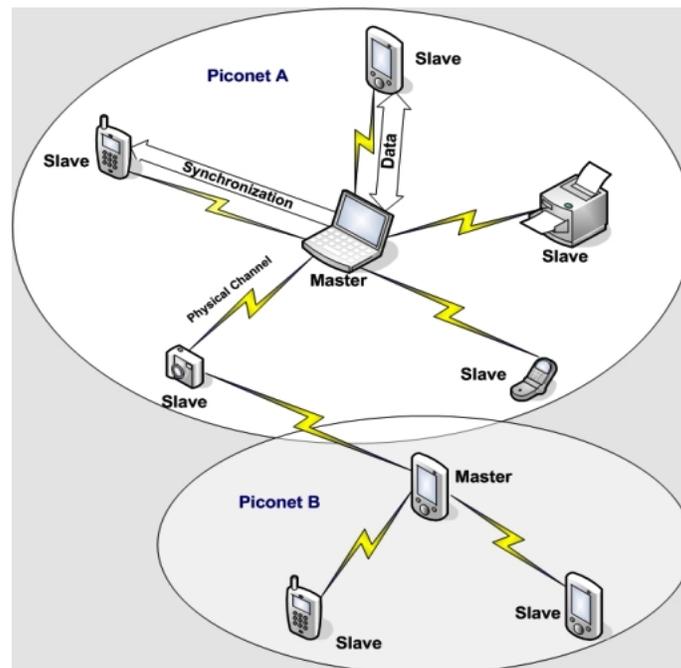


Figura 2.2 Scatternet con dos piconets.

Cuando la cantidad de *piconets* en un área aumenta, se incrementan las colisiones y se degrada el rendimiento. Las *scatternets* comparten el ancho de banda. Las *piconets* comparten el canal lógico.

En el cuadro 2.3, se muestra un resumen de las características del módulo Bluetooth que se tienen que tener en cuenta a la hora de implementarlo.

Cuadro 2.3
Cuadro resume de características de Bluetooth

CARACTERÍSTICAS/FUNCIÓN	FUNCIONAMIENTO
Tipo de conexión	Expansion de espectro (saltos de frecuencia)
Espectro	Banda ISM de 2,4 GHz.
Potencia de transmisión	1 milivatio (mw)
Velocidad de datos total	1 Mbps utilizando saltos de frecuencia
Alcance	Hasta 100 m
Estaciones soportadas	Hasta ocho dispositivos por picorred
Canales de voz	Hasta tres
Seguridad de datos	Para autenticación, una clave dde 128 bis; para cifrado, el tamaño de la clave es configurable de 8 a 128 bits
Direccionamiento	Cada dispositivo tiene una dirección (MAC) de 48 bits que se utiliza para establecer una conexión con otro dispositivo

Fuente: Bellido, F (2004). [5]

Plataforma Android [2]:

Android constituye una pila de software pensada especialmente para dispositivos móviles y que incluye tanto un sistema operativo, como *middleware* y diversas aplicaciones de usuario. Representa la primera incursión seria de Google en el mercado móvil y nace con la pretensión de extender su filosofía a dicho sector.

Todas las aplicaciones para Android se programan en lenguaje Java y son ejecutadas en una máquina virtual especialmente diseñada para esta plataforma, que ha sido bautizada con el nombre de Dalvik. El núcleo de Android está basado en Linux 2.6.

La licencia de distribución elegida para Android ha sido Apache 2.0, lo que lo convierte en software de libre distribución. A los desarrolladores se les proporciona de forma gratuita un SDK y la opción de un *plug-in* para el entorno de desarrollo Eclipse que incluyen todas las



API necesarias para la creación de aplicaciones, así como un emulador integrado para su ejecución. Existe además disponible una amplia documentación de respaldo para este SDK.

El proyecto Android está capitaneado por Google y un conglomerado de otras empresas tecnológicas agrupadas bajo el nombre de *Open Handset Alliance* (OHA). El objetivo principal de esta alianza empresarial (que incluye a fabricantes de dispositivos y operadores, con firmas tan relevantes como Samsung, LG, Telefónica, Intel o Texas Instruments, entre otras muchas) es el desarrollo de estándares abiertos para la telefonía móvil como medida para incentivar su desarrollo y para mejorar la experiencia del usuario. La plataforma Android constituye su primera contribución en este sentido.

Glosario de términos:

Cabe conocer el significado de los términos fundamentales que utilizaremos en este proyecto de grado:

- **Wireless:** en inglés, su significado es sin cables, y se denomina así a los dispositivos que no utilizan cables para realizar el envío y la recepción de datos [8].
- **Wi-Fi:** abreviatura del término inglés *Wireless Fidelity*. Es el término utilizado corrientemente para una red local sin cables (WLAN) de alta frecuencia [8].
- **WLAN** (*Wireless Local Area Network*, o red de área local inalámbrica): una WLAN es un tipo de red de área local (LAN) que utiliza ondas de radio de alta frecuencia en lugar de cables para comunicar y transmitir datos [8].
- **Onda portadora:** es una forma de onda que es modulada por una señal que se quiere transmitir (señal moduladora). Esta onda portadora es de una frecuencia mucho más alta (del espectro electromagnético), que la de la señal moduladora. De esta manera, se logra transmitir más fácilmente la señal, y el alcance que se consigue es superior [8].



- **Piconet:** es la estructura básica de comunicación donde la conexión entre dispositivos sigue un esquema maestro-esclavo, soportando hasta ocho dispositivos conectados simultáneamente [9].
- **Microcontrolador:** es un dispositivo electrónico capaz de llevar a cabo procesos lógicos. Estos procesos o acciones son programados en lenguaje ensamblador por el usuario, y son introducidos en este a través de un programador. [7].
- **Java:** es un lenguaje muy útil debido a la opción multiplataforma que provee (desde PC, Linux, Windows, hasta MAC, teléfonos, pocket PCs, etc.) [4].



CAPÍTULO III:

MARCO METODOLÓGICO

3.1 Clasificación de la investigación.

Tomando en consideración que para la realización de este Proyecto de Grado, se llevará a cabo la elaboración de un Teleprompter con comunicación Bluetooth y utilización de la plataforma de Android, esta investigación se puede clasificar como Experimental, según el criterio de la Universidad Nacional Abierta (1989). [10]

Adicionalmente, la investigación se puede clasificar como un Proyecto Factible, dado que se busca generar un servicio que puede ser comercializado, y aplicado por empresas alrededor del mundo, basado en el Manual de Trabajo de Grado de Especialización, Maestría y Tesis Doctorales de la UPEL (Universidad Pedagógica Experimental Libertador) (1998) [11] describe que un proyecto factible consiste en:

“...La elaboración y desarrollo de una propuesta de un modelo operativo viable, para solucionar problemas, requerimientos o necesidades de organizaciones o grupos sociales; puede referirse a la formulación de políticas, programas, tecnología, métodos o procesos. El proyecto debe tener apoyo en una investigación de tipo documental, de campo o un diseño que incluya ambas modalidades...” (p. 07).

3.2 Metodología de diseño

Los términos “Top-down” y “Bottom-up”, son estrategias de diseño que están embebidas en la construcción de cualquier sistema complejo. Desde una perspectiva puramente ingenieril:



Top-down (De arriba hacia abajo): al diseñar un sistema siguiendo una estrategia “Top-down” se comienza pensando en una visión global de cómo va a funcionar todo. Luego se definen cuáles van a ser los grandes componentes del sistema. Y poco a poco se va refinando y definiendo las funciones de partes más y más pequeñas hasta que se termina construyendo el sistema entero. Se entiende que la codificación no puede comenzar hasta que no se haya alcanzado un nivel de detalle suficiente, al menos en alguna parte del sistema.

Bottom-up (De abajo hacia arriba): por el contrario en un sistema diseñado con una estrategia “Bottom-up” se empieza por las partes más pequeñas, se empieza por los detalles sin tener ni idea de la visión global. Conforme se van definiendo soluciones para diversos problemas pequeños, estas soluciones se van conectando y va surgiendo una solución para varios de los sub-problemas. Conforme se van subiendo niveles el sistema general va emergiendo de forma “natural” e “inteligente”. Este enfoque tiene el riesgo de programar cosas sin saber cómo se van a conectar al resto del sistema, y esta conexión puede no ser tan fácil como se creyó al comienzo. La reutilización del código es uno de los mayores beneficios del enfoque Bottom-up.

A nivel ingenieril a veces es mejor seguir una estrategia Top-down y otras veces una Bottom-up según en lo que se esté trabajando.

Para efectos del siguiente Proyecto de Grado, un conocimiento completo del sistema se considera usualmente necesario para un buen diseño, haciendo que teóricamente sea un enfoque Top-down.

Ya que se busca crear un Teleprompter portátil el cual funciona enviando el texto que se desea ver en la pantalla desde una aplicación del teléfono móvil inteligente con plataforma Android por vía bluetooth. Dicho Teleprompter contará con dos grandes componentes



fundamentales, los cuales son el teléfono móvil con plataforma Android y el Teleprompter en cuestión.

El teléfono móvil inteligente con plataforma Android contará con una aplicación que permita realizar la conexión bluetooth con el Teleprompter para finalmente poder enviar el texto que se desea ver en la pantalla, bien sea previamente cargado en el teléfono (en un archivo de texto) o escrito en el momento de las grabaciones, además de enviar también la velocidad en la que va aparecer cada línea en la pantalla. La aplicación será realizada en el lenguaje Java orientada a objetos ya que esta forma parte de los componentes de la estructura del sistema operativo de Android.

Por otra parte, el Teleprompter se dividirá en tres bloques principales los cuales son el de comunicación, el cual está conformado por el módulo bluetooth encargado de recibir la información y enviarla vía serial al microcontrolador; el de control, integrado por el microcontrolador y cuya función es descifrar y enviar lo recibido del módulo bluetooth a la pantalla; y finalmente el bloque de visualización, quien se encarga de mostrar la información que recibe del microcontrolador en la pantalla LCD.

3.3 Desarrollo de las fases metodológicas.

FASE I: Diagnóstico de los equipos o medios de comunicación implementados actualmente dentro de la empresa caso estudio, y en otras empresas de situación similar.

Actividades:

- A. Búsqueda y recopilación bibliográfica de información relevante para la investigación.
- B. Identificación de elementos claves de éxito a ser tomados en cuenta durante el diagnóstico.



- C. Diseño de un instrumento de levantamiento de información para ser utilizado en la empresa caso estudio, y en organizaciones similares, el cual se realizará mediante un cuestionario.
- D. Aplicación del instrumento de levantamiento de información.
- E. Identificación de elementos críticos, pertinentes a la investigación, que sean comunes a las empresas evaluadas durante el diagnóstico.

Producto Final: Situación actual de la empresa caso estudio, así como otras similares, en cuanto a los equipos que poseen para comunicarse, diagnosticada.

FASE II: Recopilación de toda la información posible sobre la comunicación inalámbrica por medio de Bluetooth.

Actividades:

- A. Búsqueda y recopilación bibliográfica, como tutoriales, manuales de los fabricantes, entre otros, que permitan mayor comprensión del funcionamiento del equipo.
- B. Identificación de los módulos Bluetooth que cumplan con las necesidades básicas del presente Proyecto de Grado, como por ejemplo su alcance e implementación.
- C. Selección del módulo Bluetooth a ser implementado.

Producto Final: Selección e información sobre el módulo Bluetooth a implementar, recopilado.



Fase III: Análisis de la plataforma de Android y el lenguaje de alto nivel para la programación del microcontrolador.

Actividades:

- A. Definición de los criterios conceptuales y operacionales de la plataforma de Android y el lenguaje de alto nivel para el microcontrolador.
- B. Identificación de las herramientas que serán desarrolladas dentro de la plataforma Android.
- C. Selección del lenguaje de alto nivel que será implementado para la comunicación.

Producto Final: Conocimientos sobre el manejo de la plataforma Android y el lenguaje de alto nivel, obtenidos.

Fase IV: Creación de un software utilizando la plataforma Android

Actividades:

- A. Diseño de la aplicación en Android para darle accesibilidad a la pantalla LCD por comunicación Bluetooth.
- B. Desarrollo de la aplicación para el teléfono móvil inteligente con plataforma Android.
- C. Validación de la herramienta diseñada y su interactividad.

Producto Final: Aplicación para el teléfono móvil inteligente lista para ser utilizada como fuente de envío del texto que se quiere visualizar en la pantalla LCD.



Fase V: Elaboración de un software para el microcontrolador que reciba por medio del módulo Bluetooth la información que quiere ser visualizada en la pantalla LCD.

Actividades:

- A. Desarrollo del lenguaje de alto nivel que me permita recibir la información mediante el módulo Bluetooth y enviar vía serial dicha información a la pantalla LCD.
- B. Validación de la herramienta diseñada.

Producto Final: Lenguaje de alto nivel listo para ser implementado en la comunicación entre el teléfono móvil inteligente y la pantalla LCD.

Fase VI: Ensamble del proyecto final mediante la conexión de la pantalla LCD, el módulo Bluetooth y el microcontrolador en un solo circuito impreso.

Actividades:

- A. Elaboración del circuito impreso donde serán acoplados el módulo Bluetooth, la pantalla LCD y el microcontrolador que conforman el Teleprompter.
- B. Ensamblaje final de todos los componentes del Teleprompter, incluyendo batería.

Producto Final: Teleprompter portátil con comunicación Bluetooth entre el teléfono móvil inteligente con plataforma Android y la pantalla LCD listo para ser implementado en la empresa caso estudio.



CAPÍTULO IV:

ANÁLISIS DE RESULTADOS

4.1 UNIDAD I: Diagnóstico de los equipos o medios de comunicación implementados actualmente dentro de la empresa caso estudio, y en otras empresas de situación similar.

En la primera actividad de esta fase se realizó una búsqueda y recopilación bibliográfica que fue clave para la realización del presente Trabajo de Grado, esta fue la base para poder estudiar y fijar los puntos más importantes a desarrollar en lo que respecta al diseño y realización del Teleprompter Portátil.

Una vez realizada la revisión bibliográfica, se pudieron identificar los elementos claves de éxito a ser tomados en cuenta durante el diagnóstico, así como para el diseño del Teleprompter Portátil, los cuales son:

- Función específica del Teleprompter Portátil: Identificar necesidades futuras de fondos, para estimar la, o las formas de obtenerlos, al menor costo posible, para mantener la operatividad de la empresa y, de esta forma, satisfacer a sus clientes.
- Entorno económico: Se refiere a todo lo que está afuera de la empresa, y que termina definiendo el escenario en el cual esta se desenvolverá.

En la tercera actividad de esta fase, una vez identificados y analizados los elementos claves de éxito a ser tomados en cuenta durante el diagnóstico, se definió que el instrumento ideal para realizar el diagnóstico de la situación actual dentro de la empresa caso estudio, y en otras empresas de situación similar, en cuanto a los equipos que poseen de comunicación dentro de la empresa, era el cuestionario, toda vez que esta:



- El entrevistado tiene mayor flexibilidad al responder las preguntas.
- Asegura la elaboración uniforme de las preguntas para todos los que van a responder.
- Arroja información más detallada y específica para cada entrevistado.
- Permite una evaluación más objetiva tanto de quienes responden como de las respuestas a las preguntas.
- Se pueden aplicar más instrumentos en menos tiempo.

En ese sentido, se procedió a diseñar un cuestionario a ser aplicado a la empresa caso estudio, y a otras de función similar, cuyo objetivo fuese el de diagnosticar los elementos de éxito que no estaban presentes en las empresas y cómo su falta podían influir en su funcionamiento. El cuestionario diseñado se encuentra disponible para su revisión en el anexo nro. 1.

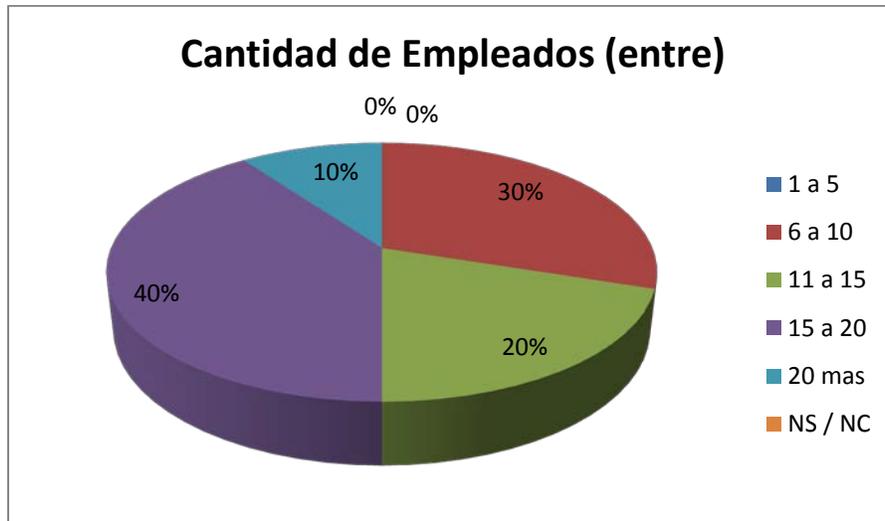
Una vez aplicados los cuestionarios para diagnosticar la empresa caso estudio, así como otras de situación similar, se pudieron obtener los siguientes resultados:

Información General de la empresa:

En los gráficos 4.1 y 4.2, se muestra las características generales de la empresa expresado en porcentajes.

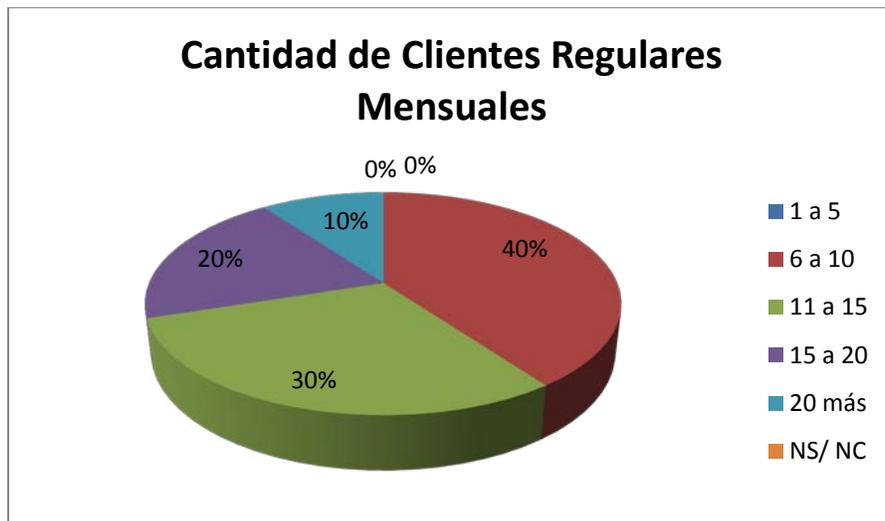
Cantidad de empleados (entre): 1 y 5: 0%, 6 y 10: 30%, 11 y 15: 20%, 15 y 20: 40%, 21 y/o más 10%, NS / NC: 0%.

Gráfico No 4.1



Cantidad de clientes regulares (entre): 1 y 5: 0%, 6 y 10: 40%, 11 y 15: 30%, 15 y 20: 20%, 20 y/o más: 10%, NS / NC: 0%.

Gráfico No 4.2

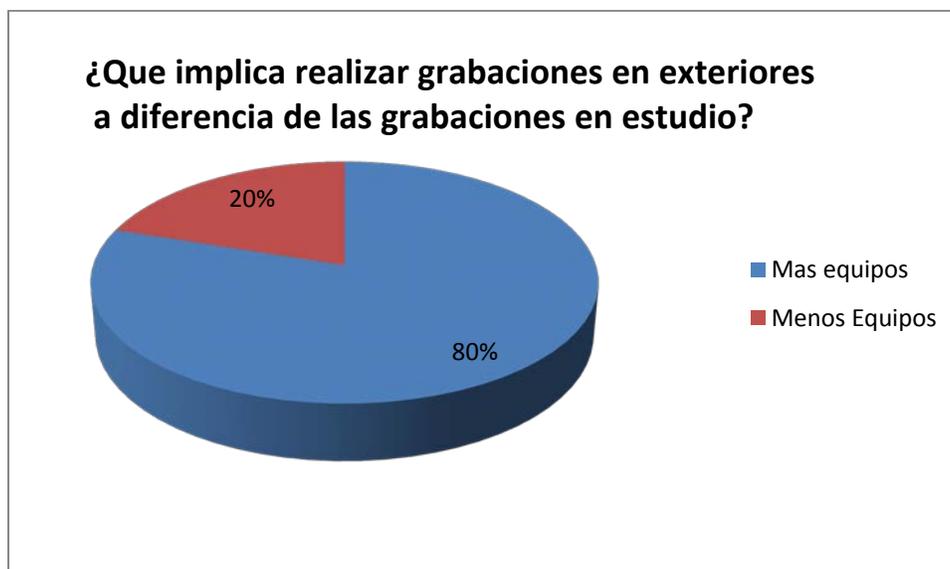


Información de los equipos de comunicación de la empresa:

En los siguientes gráficos se muestra toda la información necesaria expresada en porcentajes para determinar las características que debe tener el Teleprompter portátil.

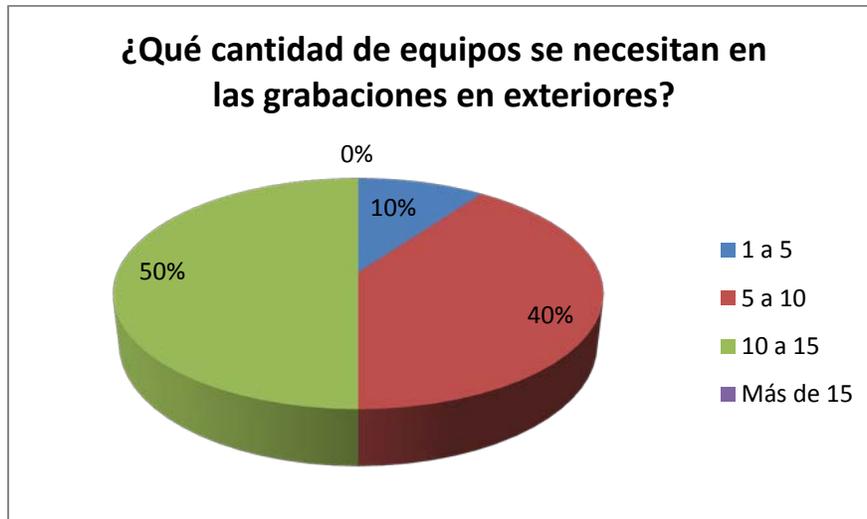
¿Que implica realizar grabaciones en exteriores a diferencia de las grabaciones en estudio?:
Más equipos de trabajo: 80%, Menos equipos de trabajo: 20%.

Gráfico No 4.3



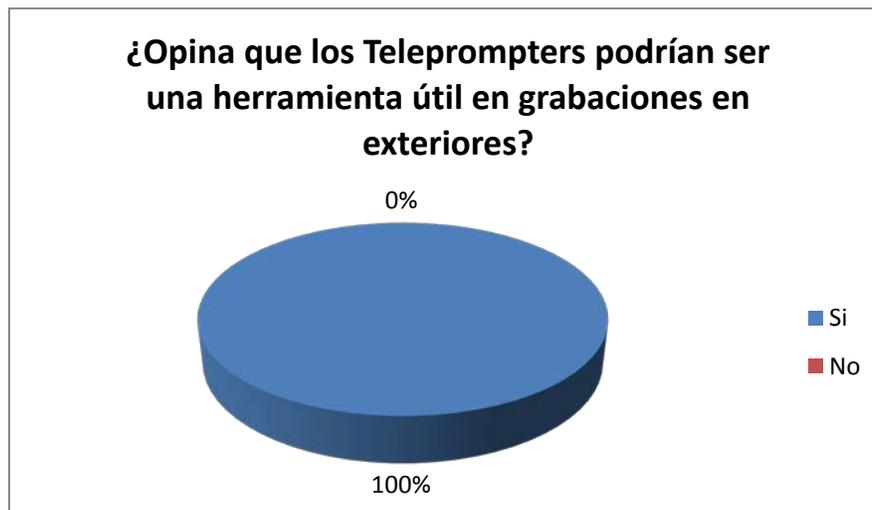
¿Qué cantidad de equipos se necesitan en las grabaciones en exteriores? (entre): 1 a 5:10%, 5 a 10: 40%, 10 a 15: 50%, Más de 15: 0%.

Gráfico No 4.4



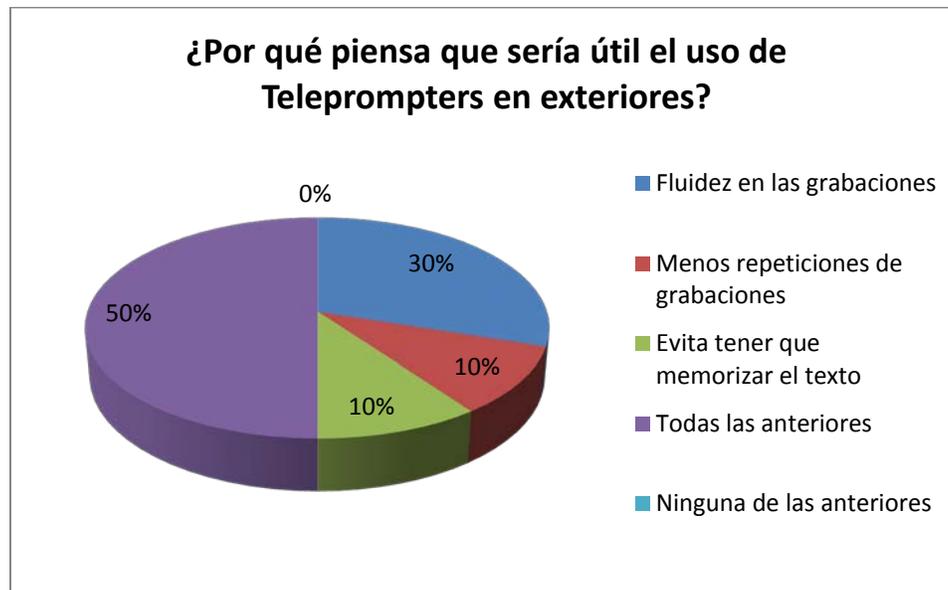
¿Opina que los Teleprompters podrían ser una herramienta útil en grabaciones en exteriores?: Si: 100%, No: 0%.

Gráfico No 4.5



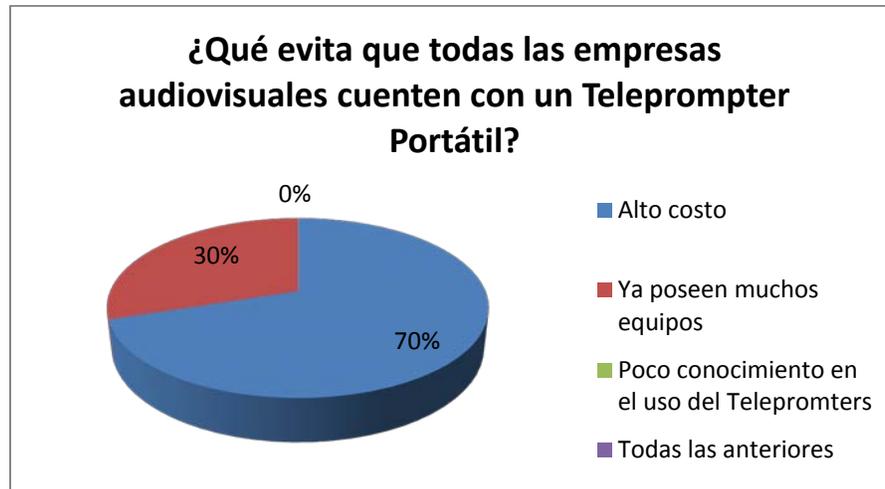
¿Por qué piensa que sería útil el uso de Teleprompters en exteriores? Fluidez en las grabaciones: 30%, Menos repeticiones de grabaciones: 10%, Evita tener que memorizar el texto: 10%, Todas las anteriores: 50%, Ninguna de las anteriores: 0%.

Gráfico No 4.6



¿Qué evita que todas las empresas audiovisuales cuenten con un Teleprompter Portátil?: Alto costo: 70%, Ya poseen muchos equipos: 30%, Poco conocimiento en el uso del Teleprompters: 0%, Todas las anteriores: 0%.

Gráfico No 4.7



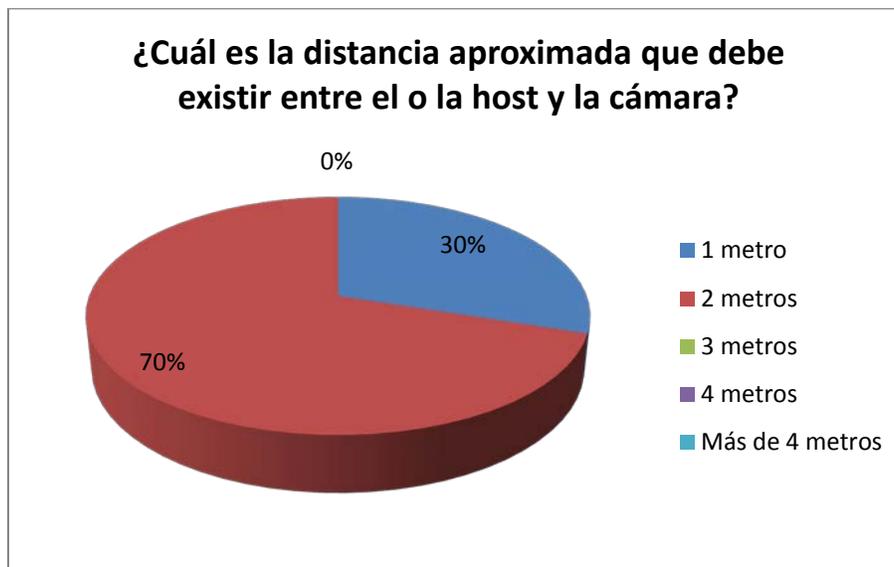
¿Cuáles son los métodos más utilizados actualmente para solventar la falta del Teleprompter en exterior?: Papel craft: 60%, Pizarra acrílica: 30%, Papel bond: 10%, Otros: 0%.

Gráfico No 4.8



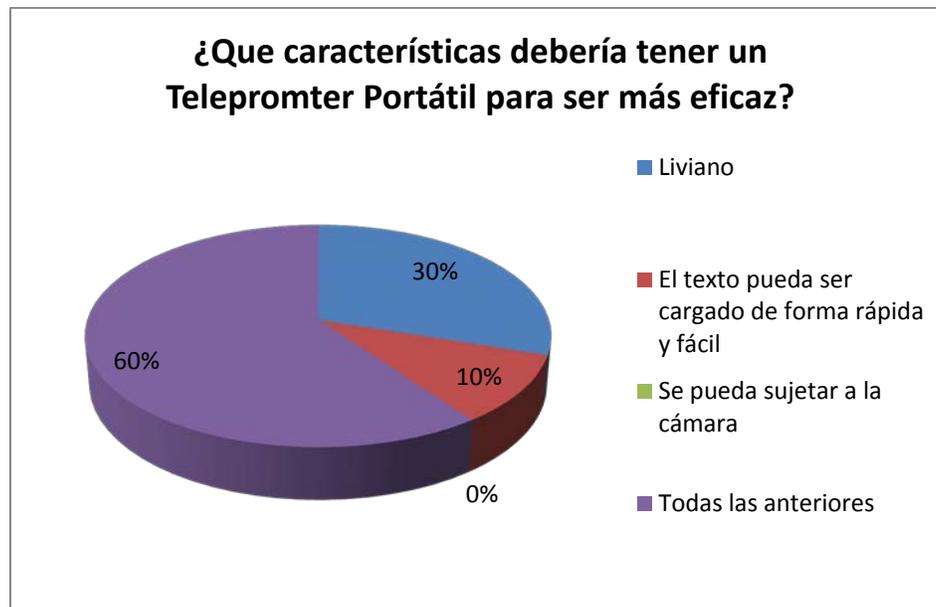
¿Cuál es la distancia que debe existir entre el host y la cámara en exteriores?: 1 metro: 30%, 2 metros: 70%, 3 metros: 0%, 4 metros: 0%, Más de 4 metros: 0%.

Gráfico No 4.9



¿Que características debería tener un Teleprompter Portátil para ser más eficaz?: Liviano: 30%, El texto pueda ser cargado de forma rápida y fácil: 10%, Se pueda sujetar a la cámara: 0%, Todas las anteriores: 60%.

Gráfico No 4.10



Habiendo tabulado los resultados de la aplicación del cuestionario, se procedió a identificar los elementos críticos, pertinentes a la investigación, que sean comunes a las empresas evaluadas durante el diagnóstico, esto con el propósito de tomarlos en consideración para el diseño y elaboración del Teleprompter portátil.

En primer lugar se estableció que para el 100% de los empleados encuestados, el Teleprompter portátil es una herramienta útil en grabaciones en exteriores, pero se detectó que para el 90% de los encuestados la cantidad de equipos necesarios, se encuentra entre 5 a 15, lo que para un alto porcentaje (80%), consideran que realizar grabaciones en exteriores requiere de una gran cantidad de equipos lo que hace el trabajo más tedioso.

Luego, se detectó que la razón por la que consideran una herramienta útil al Teleprompter portátil es el hecho de que las grabaciones serían más fluidas y sin necesidad de repetirlas ya que no es necesario memorizar el texto. Por otro lado para el 70% de los encuestados el motivo por el cual la mayoría de las empresas audiovisuales no cuentan con un

Teleprompter Portátil entre sus equipos principales para realizar las grabaciones en exteriores, es el hecho de sus altos costos.

Algunas de las características más resaltantes que para los empleados encuestados debe tener un Teleprompter Portátil, es que se pueda leer la noticia desde una distancia aproximada de 2 metros entre la cámara y el host, que sea liviano, el texto pueda cargarse de forma rápida y sencilla y que pueda fijarse a la cámara de video.

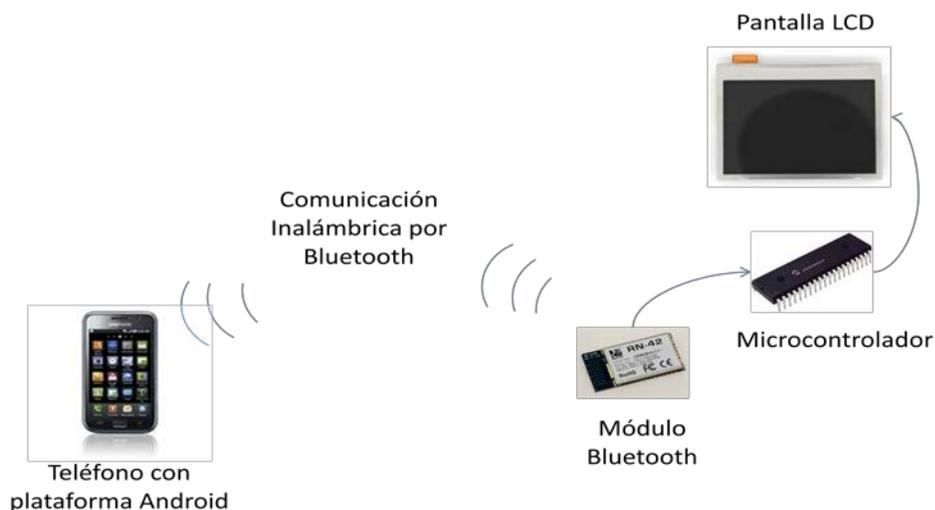


Figura 4.1 Esquema general del Teleprompter portátil.



4.2 UNIDAD II: Recopilación de toda la información posible sobre la comunicación inalámbrica por medio de Bluetooth

Habiendo completado el diagnóstico, se procedió a la definición de los dispositivos seleccionados para la elaboración del Teleprompter Portátil, entre ellos tenemos el módulo Bluetooth, el cual después de recopilar e investigar bibliografías y de estudiar las características del funcionamiento de los módulos y las variables asociadas a los mismos, a fin de seleccionar el módulo más adecuado para la comunicación entre una pantalla LCD y un teléfono móvil inteligente.

El criterio de selección se basó en el hecho de que el módulo Bluetooth utiliza como protocolo el modelo Maestro-Esclavo, lo que permite comunicar a varios elementos aislados (esclavos) con un módulo central (maestro) en el que el control de la comunicación está dirigida por el módulo central (figura. 4.2). Con este modelo se evita la posibilidad de que ocurran colisiones a la hora de realizar una transmisión, ya que en todo momento el maestro controla quien tiene acceso a la línea, lo que sugiere un método confiable y de probada capacidad para el transporte de la información. En el entorno de aplicación el maestro está conectado a un host (PC) mediante una línea de comunicación (RS232 o USB). La función del host es por un lado configurar el maestro con la lista de los esclavos a comunicar y por otro lado gestionar la comunicación bidireccional con cada esclavo.

El módulo maestro encuesta a cada uno de los esclavos (por turno circular). Para ello, envía un paquete a un esclavo proporcionándole un turno de comunicación, el esclavo le contestará con uno o varios paquetes en función de los datos que necesite transmitir. El maestro también puede enviar datos a un esclavo cuando se comunica con él.



Figura 4.2 Esquema de comunicación Maestro-Esclavo. [6]

La pila de protocolo Bluetooth [15]:

En la figura 4.3 se observa la pila de protocolos de Bluetooth.

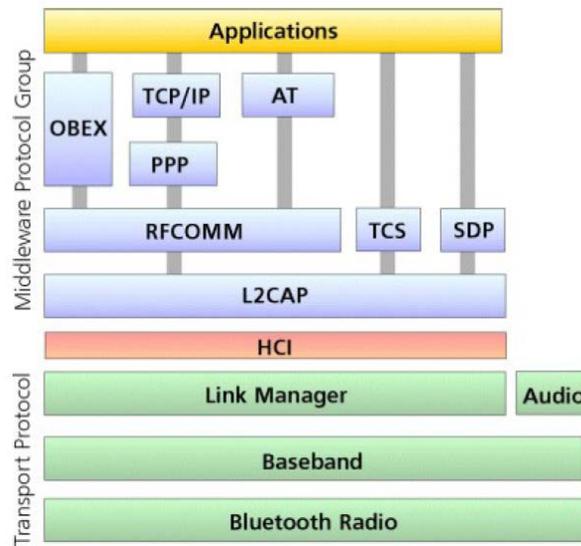


Figura 4.3 Pila de Protocolo Bluetooth.

- La capa de radio

La capa de radio define los requisitos para un transmisor-receptor de radio Bluetooth, el cual opera en la banda de los 2,4 GHz, es decir, modula y desmodula la información para



transmisión y recepción inalámbrica. Define los niveles de sensibilidad de dicho transmisor-receptor, establece los requisitos para utilizar las frecuencias del espectro expandido y clasifica a los dispositivos Bluetooth en tres clases.

Es un sistema de baja potencia con un rango de entre 1 y 100 metros que opera en la banda ISM (Industrial, Scientific and Medical) de los 2,4 GHz. La banda se divide en 79 canales de 1 MHz cada uno. La modulación es por división en frecuencia, con 1 bit por Hz, lo cual da una tasa de datos aproximada de 1 Mbps, pero gran parte de este espectro la consume la sobrecarga.

Para asignar los canales de manera equitativa, el espectro de saltos de frecuencia se utiliza a 1600 saltos por segundo y un tiempo de permanencia de 625 μ seg. Todos los nodos de una piconet saltan de manera simultánea, y el maestro establece la secuencia de salto.

Debido a que tanto el 802.11 como Bluetooth operan en la banda ISM de 2,4 GHz en los mismos 79 canales, interfieren entre sí. Puesto que Bluetooth salta mucho más rápido que 802.11, es más probable que un dispositivo Bluetooth dañe las transmisiones del 802.11 que en el caso contrario.

- La capa de banda base

La capa de banda base representa a la capa física de Bluetooth. Se usa como controladora de enlace, la cual trabaja junto con el link manager para llevar a cabo operaciones tales como la creación de conexiones de enlace con otros dispositivos.

Controla el direccionamiento de dispositivos, el control del medio (como los dispositivos se buscan unos a otros), operaciones de ahorro de energía, así como control de flujo y sincronización entre dispositivos.

La capa de banda base de Bluetooth es lo más parecido a una subcapa MAC (control de acceso al medio). Esta capa convierte el flujo de bits puros en tramas. En la forma más



sencilla, el maestro de cada piconet define una serie de ranuras de tiempo de $625\mu\text{seg}$ y las transmisiones del maestro empiezan en las ranuras pares, y las de los esclavos en las impares. Esta es la tradicional multiplexación por división de tiempo, en la cual el maestro acapara la mitad de las ranuras y los esclavos comparten la otra mitad. Las tramas pueden tener 1, 3 o 5 ranuras de longitud. Cada trama se transmite por un canal lógico llamado enlace entre el maestro y un esclavo. Hay dos tipos de enlaces:

Conexión Asíncrona (ACL), que se utiliza para datos conmutados en paquetes disponibles a intervalos irregulares. Estos datos provienen de la capa L2CAP en el nodo emisor y se entregan en la capa L2CAP en el nodo receptor. No hay garantías de entrega del tráfico ACL. Las tramas se pueden perder y tienen que retransmitirse. Un esclavo sólo puede tener un enlace ACL con su maestro.

Conexión Síncrona Orientada (SCO), para datos en tiempo real, como ocurre en las conexiones telefónicas. A este tipo de canal se le asigna una ranura fija en cada dirección. Las tramas que se envían a través de SCO nunca se retransmiten. En vez de ello se puede usar la corrección de errores hacia adelante para conferir una alta fiabilidad al sistema. Un esclavo puede establecer hasta tres enlaces SCO con su maestro. Cada enlace de este tipo puede transmitir un canal de audio PCM (modulación por impulsos codificados) de 64000 bps.

- Link Manager

El administrador de enlaces o Link Manager, se encarga de establecer canales lógicos entre dispositivos, incluyendo administración de energía, autenticación y calidad de servicio, es decir, controla y configura enlaces con otros dispositivos.



- Host Controller Interface (Interfaz del Controlador del Huésped)

El HCI permite el acceso mediante línea de comandos a la capa de banda base y al LMP para controlar y recibir información acerca del estado, en otras palabras, maneja comunicaciones entre un huésped externo y un módulo bluetooth. Se compone de tres partes:

1. El *firmware* HCI, o programa oficial del fabricante, actualmente forma parte del hardware Bluetooth.
2. El controlador HCI, o driver, que se encuentra en el software del dispositivo Bluetooth.
3. El Host Controller Transport Layer, que conecta el *firmware* con el driver.

- Protocolo de adaptación y control de enlaces lógicos L2CAP

El protocolo L2CAP aísla a las capas superiores de los detalles de transmisión. Es análogo a la subcapa LLC (Control de enlace lógico) del estándar 802, pero difiere de esta en el aspecto técnico. Proporciona servicios orientados y no orientados a la conexión a las capas superiores.

La capa L2CAP tiene tres funciones principales:

1. Acepta paquetes de hasta 64 KB provenientes de las capas superiores y los divide en tramas para transmitirlos. Las tramas se reensamblan nuevamente en paquetes en el otro extremo.
2. Maneja la multiplexación y desmultiplexación de múltiples fuentes de paquetes. Cuando se reensambla un paquete, la capa L2CAP determina que protocolo de las capas superiores lo manejará, por ejemplo, RFCOMM o el de telefonía.



3. Se encarga de la calidad de servicio, tanto al establecer los enlaces como durante la operación normal. Asimismo, durante el establecimiento de los enlaces se negocia el tamaño máximo de carga útil permitido, para evitar que un dispositivo que envíe paquetes grandes sature a uno que recibe paquetes pequeños.

- RFCOMM

Es el protocolo que emula el puerto serie estándar RS232 de los ordenadores para la conexión de teclados, ratones y módems, entre otros dispositivos. Su propósito es que dichos dispositivos no tenga por qué saber nada acerca de Bluetooth.

- TCS y SDP

El Protocolo de Control de Telefonía TCS es un protocolo de tiempo real y está destinado a los tres perfiles orientados a voz (Manos libres (HFP), Auriculares (HSP) e Intercom (ICP)). También se encarga del establecimiento y terminación de llamadas. Por su parte, el Protocolo de Descubrimiento de Servicios (SDP) se emplea para la detección automática de dispositivos dentro de la red, así como de servicios ofrecidos por dichos dispositivos.

Otro punto a destacar de los módulos Bluetooth, es que la información que se intercambia entre dos unidades Bluetooth se realiza mediante un conjunto de *slots* (sistema FH/TDD (salto de frecuencia/división de tiempo dúplex), en el que el canal queda dividido en intervalos de 625 μ s) que forman un paquete de datos. Cada paquete comienza con un código de acceso de 72 bits, que se deriva de la identidad maestra y cuyo propósito es que los esclavos que se encuentren en el rango de alcance de dos maestros sepan que tráfico está destinado a ellos. A continuación se encuentra un encabezado de 54 bits, éste contiene importante información de control, como tres bits de acceso de dirección, tipo de paquete, bits de control de flujo, bits para la retransmisión automática de la pregunta, y chequeo de errores de campos de cabecera. La dirección del dispositivo es en forma hexadecimal.

Finalmente, el paquete que contiene la información, que puede seguir al de la cabecera, tiene una longitud de 0 a 2745 bits.

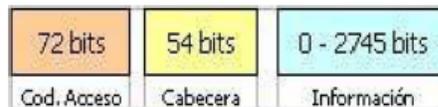


Figura 4.4 Datagrama Bluetooth [14]

En cualquiera de los casos, cada paquete que se intercambia en el canal está precedido por el código de acceso. Los receptores de la *piconet* comparan las señales que reciben con el código de acceso, si estas no coinciden, el paquete recibido no es considerado como válido en el canal y el resto de su contenido es ignorado.

Profundizando un poco en el encabezado, el campo *Dirección* identifica a cuál de los ocho dispositivos activos está destinada la trama. El campo *Tipo* indica el tipo de trama (ACL, SCO, de sondeo o nula), el tipo de corrección de errores que se utiliza en el campo de datos y cuantas ranuras de longitud tiene la trama.

Un esclavo establece el bit F (de flujo) cuando su búfer está lleno y no puede recibir más datos. Ésta es una forma primitiva de control de flujo. El bit A (de confirmación de recepción o Acknowledgement) se utiliza para incorporar un ACK en la trama. Por último, el bit S (de secuencia) sirve para numerar las tramas con el propósito de detectar retransmisiones. El protocolo es de parada y espera, por lo que 1 bit es suficiente.

A continuación se encuentra el encabezado Suma de verificación de 8 bits. Todo el encabezado de 18 bits se repite tres veces para formar el encabezado de 54 bits. En el receptor se comprueban las tres copias de cada bit y si coinciden, se acepta. De lo contrario se impone la opinión de la mayoría.

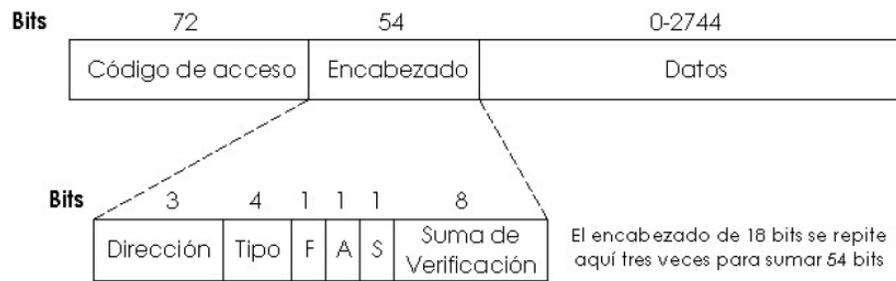


Figura 4.5 Estructura de la trama de Bluetooth. [15]

Por otro lado, el alcance de estos dispositivos Bluetooth viene marcado según la clase con el que se haya fabricado. Estas son las tres posibles clases que podemos encontrar:

- Clase 1. Alcanzando hasta los 100 m se utiliza principalmente en el sector industrial.
- Clase 2. Con un alcance de 10 m vienen siendo habituales en dispositivos portátiles.
- Clase 3. Suelen tener un alcance de 1 a 3 m.

La potencia del transmisor Bluetooth viene también determinada por la Clase que se utilice tal y como se muestra en la tabla 4.1:

Tabla 4.1 Potencias según Clase en la tecnología Bluetooth

Potencia Clase	Potencia de salida máxima	Potencia de salida nominal	Potencia de salida mínima*	Control de potencia
1	100 mW (20 dBm)	No corresponde	1 mW (0 dBm)	Pmin<+4 dBm a Pmax Opcional: Pmin** a Pmax
2	2,5 mW (4 dBm)	1 mW (0 dBm)	0,25 mW (-6 dBm)	Opcional: Pmin** a Pmax
3	1 mW (0 dBm)	No corresponde	No corresponde	Opcional: Pmin2** a Pmax

Fuente: Puig, J (2008).

Establecido todo lo anterior y basados en el hecho de que la distancia que se necesita para transmitir la información desde el teléfono móvil inteligente hasta el módulo de la pantalla LCD no requiere de grandes distancias, ya que el director o productor encargado de manejar el teleprompter debe estar cerca para supervisar las grabaciones, se decidió utilizar el módulo Bluetooth RN-42 Clase 2 de la empresa ROVINC NETWORKS, cuyo alcance es suficiente para satisfacer las necesidades del proyecto a realizar (figura 4.6).



Figura 4.6 Módulo Bluetooth RN-42 Clase 2.

Fuente: Data Sheet Módulo Bluetooth RN-42 clase 2

Características y configuración del módulo Bluetooth RN-42:

El RN-42 tiene un correspondiente reducido consumo de potencia. Es perfecto para aplicaciones de corto alcance alimentadas mediante batería. Usa solamente 26uA en modo "sleep" mientras permanece aún conectado y reconocible. Múltiples modos de configuración de potencia permiten al usuario elegir el perfil de menor consumo de potencia para una determinada aplicación.

Soporta múltiples perfiles Bluetooth como SPP (Perfil Puerto serial) y HID (Dispositivo de Interfaz Humana), así como la interface UART (transmisor-receptor asíncrono universal), lo cual facilita su integración simple con sistemas embebidos o para su conexión con dispositivos existentes.



❖ **Características [12]:**

- Módulo Bluetooth totalmente certificado.
- Certificación FCC (Comisión Federal de Comunicaciones).
- Soporta Bluetooth v2.0+EDR
- Módulo Bluetooth calificado 2.1/2.0/1.2/1.1
- Interfaces de conexión de datos UART (SPP o HCI) y USB (sólo HCI)
- Soporta ratas de datos SPP - 240Kbps (slave), 300Kbps (master)
- Soporta ratas de datos HCI - 1.5Mbps, 3.0Mbps
- Dispone de software para modo HCI ó SPP/DUN
- Antena tipo chip
- Alcance: hasta 20m con línea de vista.
- Frecuencia: 2.402 ~ 2.48 GHz
- Modulación: FHSS/GFSK (79 canales a intervalos de 1MHz)
- Comunicación segura, encriptación de 128 bits
- Corrección de errores
- Potencia de salida: 4dBm
- Sensitividad: -80dBm
- Consumo de corriente en transmisión: 25mA
- Consumo de corriente en recepción: 25mA
- Voltaje de alimentación: 3V ~ 3.6V

- Tamaño: 13.4mm x 25.8mm
- UART totalmente configurable
- Velocidades de transmisión UART hasta 3Mbps
- Velocidades de transmisión al aire de 721kbps a 2.0Mbps
- Modo de descanso de baja potencia (low power sleep)
- Compatible con todos los productos Bluetooth que soportan SPP
- Incluye soporte para protocolos BCSP, DUN, LAN, GAP SDP, RFCOMM, y L2CAP

Una alternativa al dispositivo utilizado pudiera ser el Bluetooth RN41, de similares características, cuya diferencia principal es que es de clase 1 de mayor alcance, con un poco más de consumo de corriente, pero su conexión es igual.

❖ **Pines de configuración del Bluetooth RN42:**

Los pines que utilizan para conectar el Bluetooth son el de TX (pin 14), RX (pin 13), GND (pin 12), VDD (pin 11), PIO8 (pin 31 donde se conecta un LED de estado que indica sin conexión o baja conexión), PIO2 (pin de estado conectado, aquí se conecta un LED). Como primera instancia se establece como alimentar y conectar el módulo Bluetooth, permitiendo saber sus características de funcionamiento y así poder ingresar a modo de configuración y cambiar todos sus parámetros de fábrica, como la velocidad, nombre, identificador, código de seguridad, encriptamiento etc.

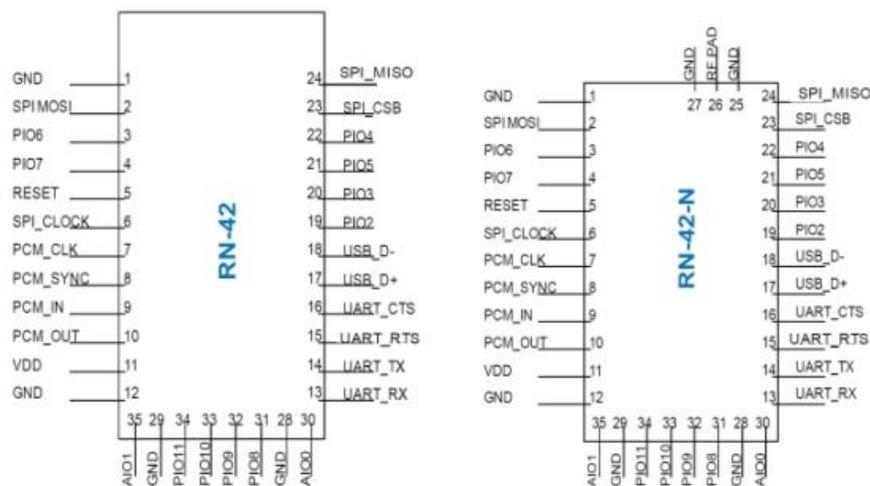
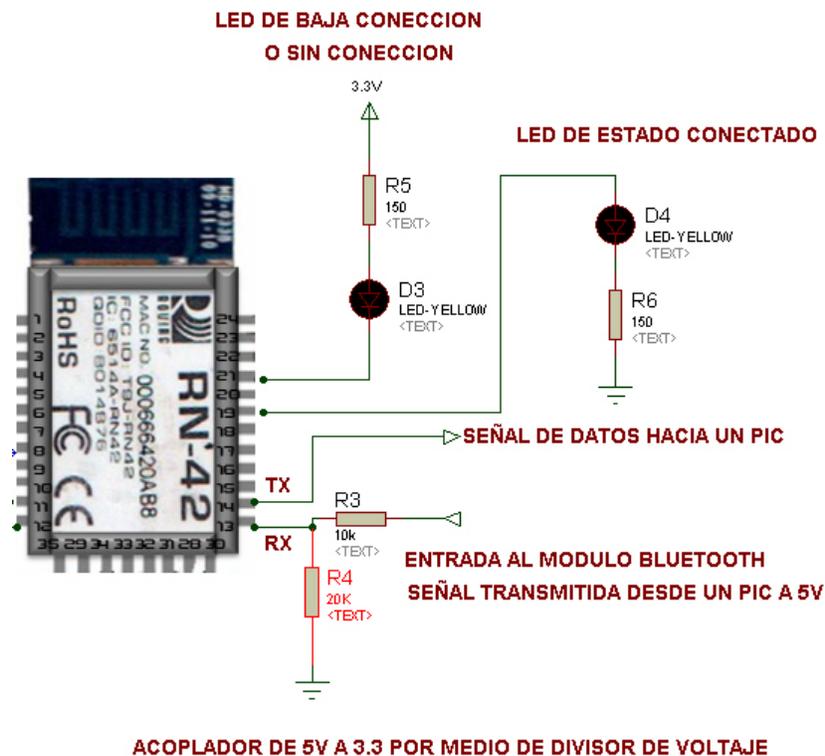


Figura 4.7 Pines de descripción RN-42

Fuente: Data Sheet Módulo Bluetooth RN-42 clase 2

Circuito recomendado para conectarlo por primera vez:

Es recomendable que el módulo Bluetooth solo trabaje a 3.3v, para su alimentación se debe usar un regulador, podemos conectar un LD33, un LD1117V33, NTE1904 o un sencillo Lm317 calibrado a 3.3v. Para este propósito usaremos el NTE1904, el cual trabaja como un regulador de voltaje de 3.3 v y 1A.

Circuito para montar el RN42:**Figura 4.8 Circuito de conexión para el RN-42**

En el circuito podemos ver una R3 y R4 que son necesarias solo cuando se conecta circuitos TTL (5v) porque si conectan directamente al módulo lo pueden quemar, para esto



se debe colocar un divisor de voltaje que garantiza una entrada de 5v y en su salida sale 3.3v, voltaje requerido para su funcionamiento, cabe aclarar que un microcontrolador por lo común se lo puede trabajar a 3.3v por lo que no es necesario las dos resistencias de protección, pero si conecta una LCD esta no funcionará y aquí es cuando debemos tener dos fuentes, una de 5v y otra de 3.3v como sugiere la figura 4.5. Estos módulos vienen de fábrica con un nombre preestablecido y a una velocidad de 115200bps, 8bits, Sin Paridad, 1bit de Stop.

Pasos para configurar el módulo RN42:

1. Alimente al módulo, active el Bluetooth del celular o computador, luego realice el enlace o emparejamiento de estos digitando el PIN que de fábrica es 1234.
2. Desconecte y vuelva a conectar el Bluetooth y tendrá 60 segundos para ingresar a su configuración.
3. El módulo se lo puede conectar por RS232 con los acoples respectivos como MAX232 y el divisor de voltaje a un puerto serie por medio de un DB9, se debe abrir el hiperterminal o cualquier programa que permita leer y enviar comandos AT. La segunda opción es prender el Bluetooth de un computador o laptop y por medio de algún programa que controle Bluetooth crear un COM virtual que le permita al hiperterminal enviar y recibir datos de forma inalámbrica (de esta manera se conecta TX y RX). Para efecto de este proyecto se utilizó una aplicación de Android llamada BlueTerm que se puede descargar gratis en el teléfono móvil inteligente, el cual es un emulador de terminal para conectarse a cualquier dispositivo con puerto serie mediante un adaptador bluetooth-serie.



Figura 4.9 Interfaz del BlueTerm.

4. Para entrar en el modo de configuración del módulo se debe enviar 3 signos pesos \$\$\$ dentro de los primeros 60 segundos tras haber alimentado a 3.3v el dispositivo, ya que si no, entra en modo datos (Data mode) y los comandos son ignorados, si los datos fueron leídos, el módulo responderá con CMD y después del envío de instrucciones nos regresa un AOK. La forma de ver si el módulo está bien, es mirar el LED de status, siempre debe estar parpadeando, después de entrar en Command mode la oscilación del LED es más rápida y cuando esta enlazado con algún dispositivo el LED deja de parpadear y también se enciende el LED de estado conectado.
5. Para mirar la ayuda y mirar que comandos necesito digite en modo Command mode la letra “h” de ayuda, para mirar la configuración de fábrica digite “d”.
6. Comandos básicos:

Para cambiar el nombre del dispositivo Bluetooth digitamos en hiperterminal o en el celular los siguientes comandos:

SN

SN, Teleprompter; Ahora nuestro dispositivo se llama Teleprompter.

Para cambiar el Baud Rate:

SU

SU, 19.2; Solo se necesitan los primeros dos números.

Se lo puede configurar en cualquiera de las siguientes velocidades

1200, 2400, 4800, 9600, 19.2, 28.8, 38.4, 57.6, 115K, 230K, 460K, 921K

Para cambiar el código de seguridad: de fábrica viene el código PIN 1234

SP

SP, 0000; ahora el código de enlace es 0000.



Figura 4.10 Configuración de los comandos del módulo bluetooth.

Un aspecto bastante curioso y útil de estos módulos es la facilidad para cambiarle el identificador al Bluetooth SC (Service Class) y SD (Device Class) sirven para decirle al Bluetooth si es una computadora, un modem, una laptop, un teléfono etc., para ello debemos buscar en internet páginas que nos generen el código CoD y mediante los comandos le cambiamos la forma de cómo lo reconocen los demás dispositivos.

Una vez realizado los pasos anteriores, el módulo Bluetooth ya está listo para ser utilizado como receptor y/o transmisor de datos. Lo cual se pudo verificar realizando el montaje del circuito MAX3232 que se muestra en la figura 4.11, el cual permite conectar el módulo bluetooth a un puerto serie del computador por medio de un DB9.

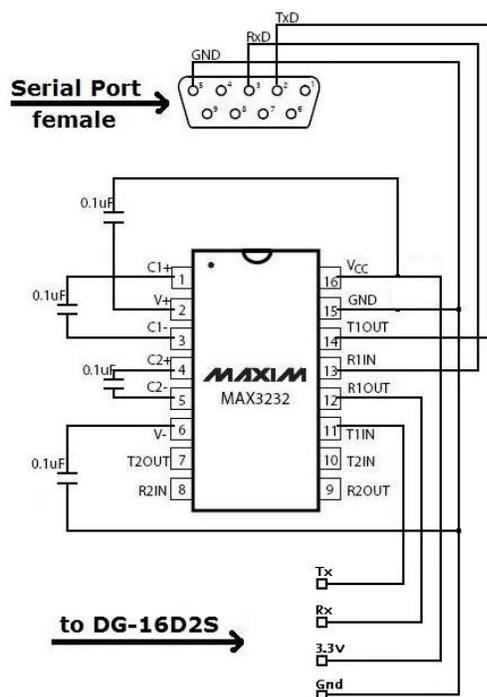


Figura 4.11 Conexión del MAX3232.

Fuente: Data Sheet MAX3222-MAX3241

Una vez realizado el montaje se utilizó la aplicación BlueTerm antes mencionada y el USART Terminal de Mikrobasic Pro (figura 4.12), los cuales sirvieron para enviar y/o recibir texto tanto del teléfono móvil como de Mikrobasic por medio del bluetooth, dejando demostrado que el módulo bluetooth está configurado y listo para ser usado.

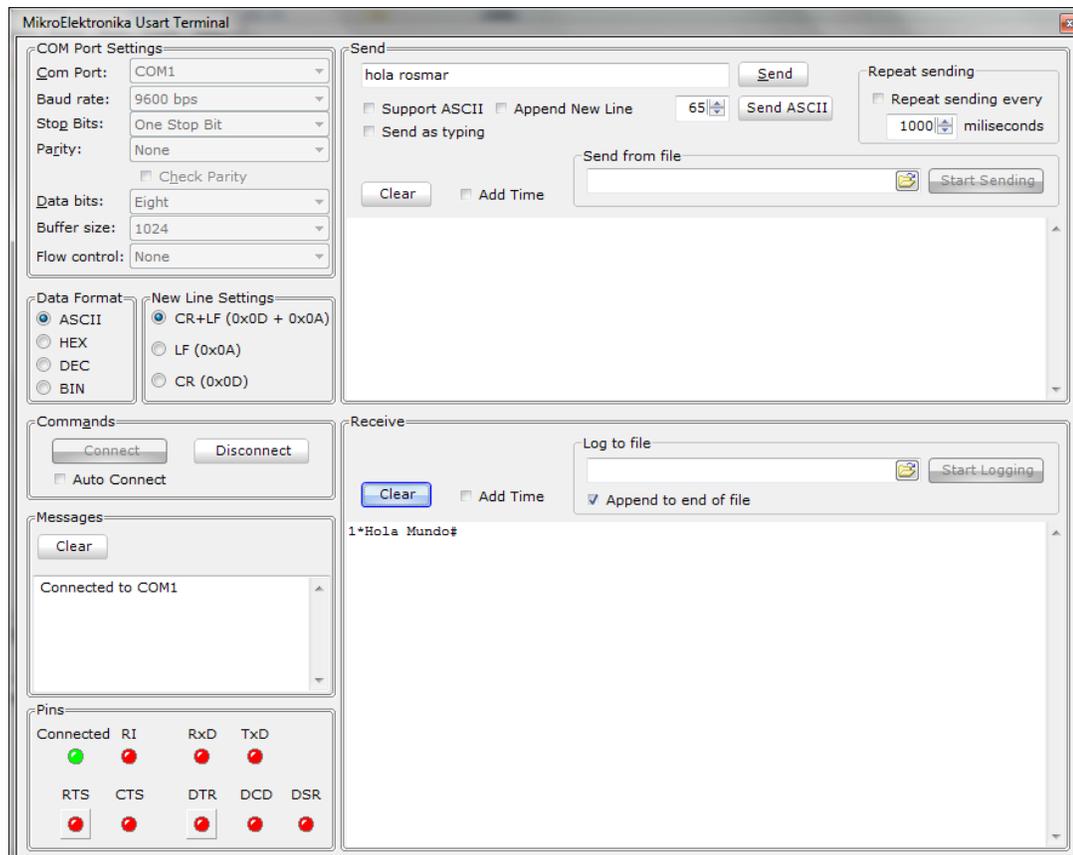


Figura 4.12 USART Terminal de Mikrobasic Pro.

4.3 UNIDAD III: Análisis de la plataforma de Android y el lenguaje de alto nivel para la programación del microcontrolador.

4.3.1 Plataforma de Android

Como se explicó en el Capítulo II, la estructura del sistema operativo Android se compone de aplicaciones que se ejecutan en un *framework* (infraestructura digital) Java de aplicaciones orientadas a objetos sobre el núcleo de las bibliotecas de Java en una máquina virtual Dalvik con compilación en tiempo de ejecución. El sistema operativo está compuesto por 12 millones de líneas de código, incluyendo 3 millones de líneas de XML, 2,8 millones de líneas de lenguaje C, 2,1 millones de líneas de Java y 1,75 millones de líneas de C++.

Google reveló sus estadísticas sobre el reparto del entorno de Android. Siendo Android 2.3 (Gingerbread), la versión más utilizada pues acapara el 55,8 % del total del mercado Android.

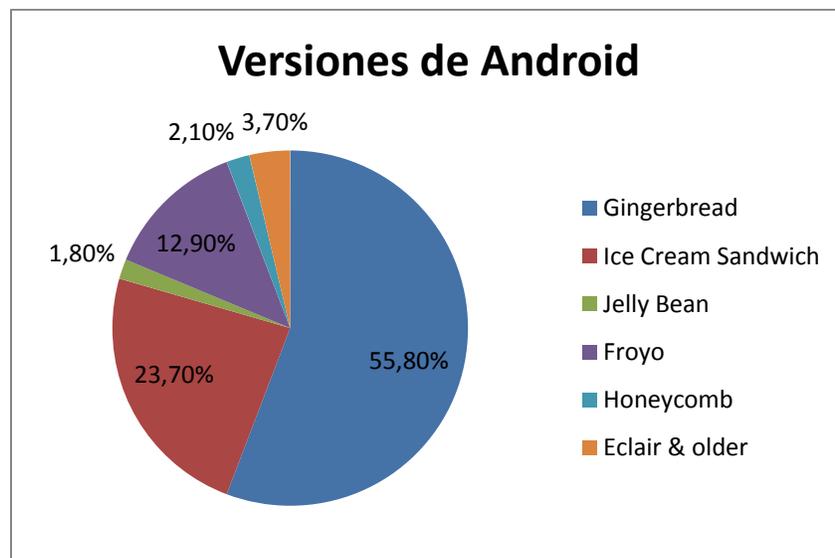


Figura 4.13 Estadísticas de utilización de las distintas versiones de Android. [13]



De las 10 versiones existentes de Android, se realizó la aplicación en la versión 2.2 (API 8), conocida con el nombre de Froyo v2.2 (Abreviatura de «Frozen Yogurt») de manera que dicha aplicación pueda ser utilizada en las versiones siguientes a esta.

El desarrollo de aplicaciones para Android no requiere aprender lenguajes complejos de programación. Todo lo que se necesita es un conocimiento aceptable de Java y estar en posesión del kit de desarrollo de software o «SDK» provisto por Google el cual se puede descargar gratuitamente.

Selección de programas para el desarrollo de la aplicación en Android:

Una vez que sabemos que es Android y la versión en la cual se desenvolverá la aplicación, se procede al desarrollo de la misma.

Antes de nada, se necesitó configurar el entorno de desarrollo, para empezar a implementar la aplicación. Por lo que fue importante tener una serie de herramientas instaladas en el sistema, antes de empezar a programar.

➤ Eclipse

Es un programa que cuenta con un conjunto de herramientas para ser usado por un programador, es de código abierto multiplataforma y permite desarrollar aplicaciones de cliente enriquecido.

La característica principal de eclipse, es que es extenso. La forma en que los *plugins* interactúan es mediante interfaces o puntos de extensión; así, los nuevos aportes se integran fácilmente.

El programa es neutro y se adapta a cualquier tipo de lenguaje, a pesar de que su mayor parte está escrita en java (excepto el núcleo). Su uso más popular es como IDE para java. Los siguientes componentes constituyen la plataforma de cliente enriquecido:



1. Plataforma principal: Inicio de Eclipse, ejecución de los *plugins*.
2. Plataforma para integrar distribuciones: OSGI (está definida con una serie de APIS básicas para el desarrollo de servicios).
3. El *Standard Widget Toolkit* (SWT): es un conjunto de componentes para construir interfaces gráficas en Java.
4. *JFace*: se encarga del manejo de archivos, manejo de texto, editores de texto.
5. El *Workbench* de Eclipse: se encarga de las vistas, los editores, las perspectivas y los asistentes.

En cuanto a las aplicaciones clientes, eclipse provee *frameworks* muy extensos para el desarrollo de aplicaciones gráficas, definición y manipulación de modelos de software, aplicaciones web.

Habiendo establecido lo anterior, se explica porque fue seleccionado este IDE, también se podía programar en Netbeans, pero el *pluging* de Netbeans para Android es mucho menos maduro que el de Eclipse.

Para instalar este IDE, se buscó en la página del proyecto Eclipse <http://www.eclipse.org/downloads/> y se descargó la versión para desarrolladores de Java.

Eclipse tiene la ventaja de que no necesita instalación. Solo se descarga el programa, lo descomprimos y tal como está descomprimido, lo podemos usar.

➤ **SDK Android**

Otro elemento muy importante para desarrollar la aplicación en Android, fue su SDK (Software Development Kit). Para adquirir el SDK, fue necesario acceder a la siguiente página:

<http://code.google.com/intl/es-ES/Android/download.html>

Una vez descargado, solo se descomprimió en cualquier carpeta en el disco duro, recordando siempre su ubicación.

En el SDK se encontrar todas las librerías y utilidades necesarias para poder trabajar con Android.

➤ Plugin Eclipse

Para integrar Android con Eclipse, se necesitó un *plugin*, que dependiendo de la versión del IDE que tengamos, se instalará de una forma concreta.

Para dicha instalación se accedió al menú “Help->Software updates”. Donde, se pudo controlar todas las actualizaciones y *plugins* de Eclipse. Posteriormente, fue necesario añadir una nueva dirección para que el sistema localizara e instalara el *plugin* de Android.

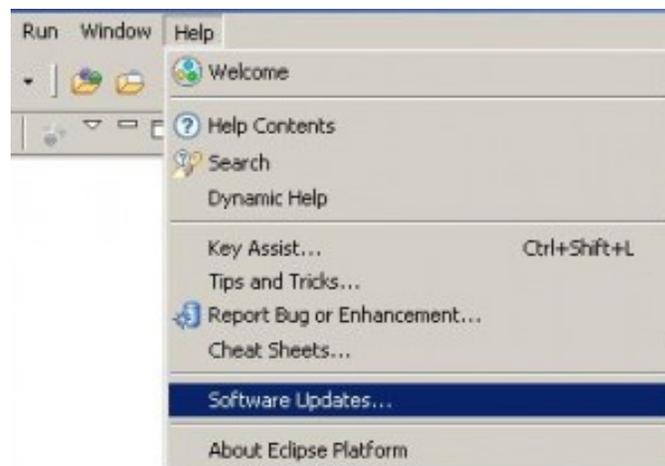


Figura 4.14 Menú de ayuda

Fuente: Software Eclipse

Para ello, se pulso “Add Site...”, y en la nueva ventana que se abre, se insertó la siguiente dirección:

<https://dl-ssl.google.com/Android/eclipse/>

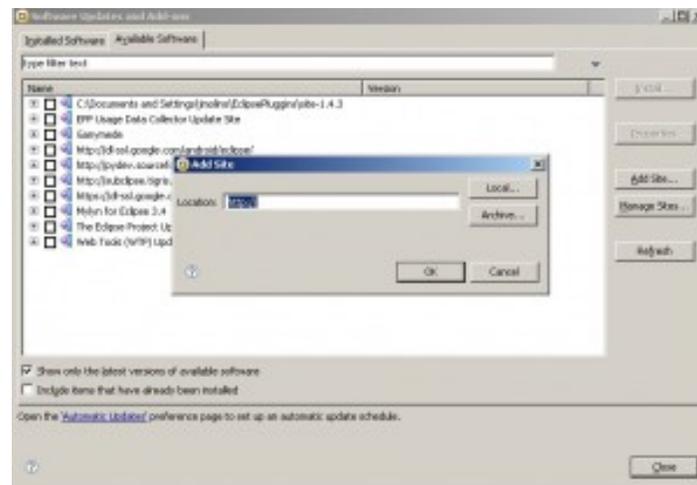


Figura 4.15 Integración de Android con Eclipse

Fuente: Software Eclipse

Nota: en el caso de no reconocer la dirección, es recomendable probar con “http” en vez de “https”.

Una vez que se añadió el sitio, se procedió a marcar e instalar el nuevo *plugin* que nos apareció en la lista. En caso que pida instalar otro *plugin*, se busca mediante el buscador de *plugins* y se instala de igual forma que empezamos a instalar el de Android. Una vez instalado todo, se reinició Eclipse.

Finalizado la instalación de todo lo antes mencionado, al abrir nuevamente Eclipse fue necesario darle la ubicación del SDK de Android. Para ello se selecciona en el menú “Windows->Preferences”. Y allí aparece una ventana como la siguiente

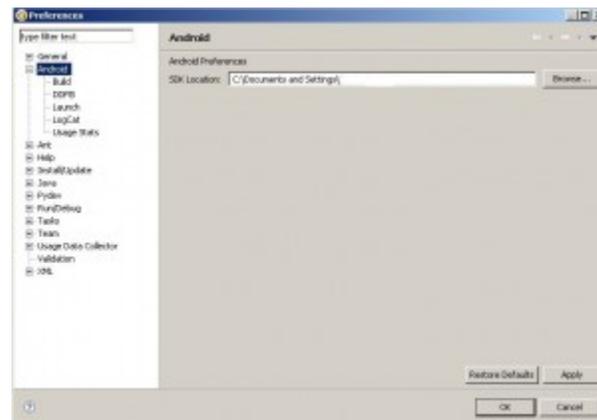


Figura 4.16 Colocación de la ubicación del SDK

Fuente: Software Eclipse

Marcamos la etiqueta de “Android” y en “SDK Location” se busca y añade la dirección donde se guardó el SDK.

Una vez hecho esto, el programa está listo para empezar a crear la aplicación en Android.

4.3.2 Lenguaje de Alto Nivel

Después de haber revisado y evaluado los diferentes lenguajes de programación del PIC se decidió utilizar Mikrobasic como software de programación. La selección se basó en el hecho de su fácil manejo y comprensión del lenguaje, así como también del hecho de que es uno de los softwares de programación más utilizados en el transcurso de la carrera.

4.4 UNIDAD IV: Creación de un software utilizando la plataforma Android

La aplicación en Android fue diseñada de forma que su utilización sea muy sencilla. Cuenta con un cuadro de texto donde se va a cargar o escribir la información que se desea enviar al Teleprompter, cuatro (4) botones que permiten seleccionar cuatro (4) tipos de velocidades distintas dependiendo de la rapidez de lectura del usuario, un (1) botón que permite cargar

un archivo de texto guardado en la memoria del teléfono móvil inteligente, otro botón cuya función es enviar el texto y la velocidad seleccionada a la pantalla del Teleprompter. Además, cuenta con un botón para conectar al dispositivo al cual se le va enviar el texto y otro botón de cerrar cuya función es finalizar la aplicación así como también terminar la conexión con el módulo bluetooth al cual se le está enviando el texto.

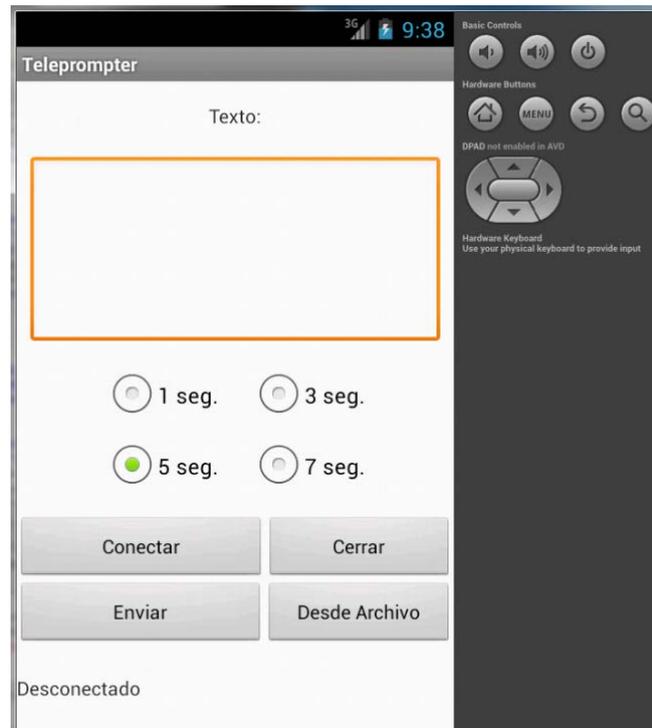


Figura 4.17 Interfaz Gráfica de la aplicación en Android.



Diagramas de Flujos del funcionamiento de la aplicación:

Al entrar a la aplicación es necesario conectar el dispositivo con el módulo bluetooth del Teleprompter. Una vez realizada la conexión se procede al envío del texto, bien sea escrito en el momento de la grabación o previamente cargado en un archivo de texto en el teléfono móvil inteligente.

En los siguientes diagramas de flujo se explica detalladamente el funcionamiento de la aplicación del Teleprompter portátil y las distintas opciones que posee:

- Diagrama de Flujos para escribir el texto a enviar:

En el siguiente diagrama se muestra la utilización de la aplicación del Teleprompter como medio para enviar una idea, comentario u opinión por parte del director o productor, con la finalidad de que dichas personas pueda comunicarse con el host (presentador) sin tener que interrumpir las grabaciones.

Su utilización radica en seleccionar el cuadro de texto que producirá el despliegue del teclado táctil para de esta manera poder escribir el texto que se desea enviar, una vez hecho esto, se selecciona la velocidad a la que se quiere que aparezca el texto en la pantalla, dicha selección se basará en la rapidez de lectura del presentador. En caso de no seleccionar una velocidad el programa por sí mismo establece como velocidad definida la de 3 segundos. La estructura del envío estará comprendida por la velocidad separada por un asterisco (*) del texto que se desee observar en la pantalla.

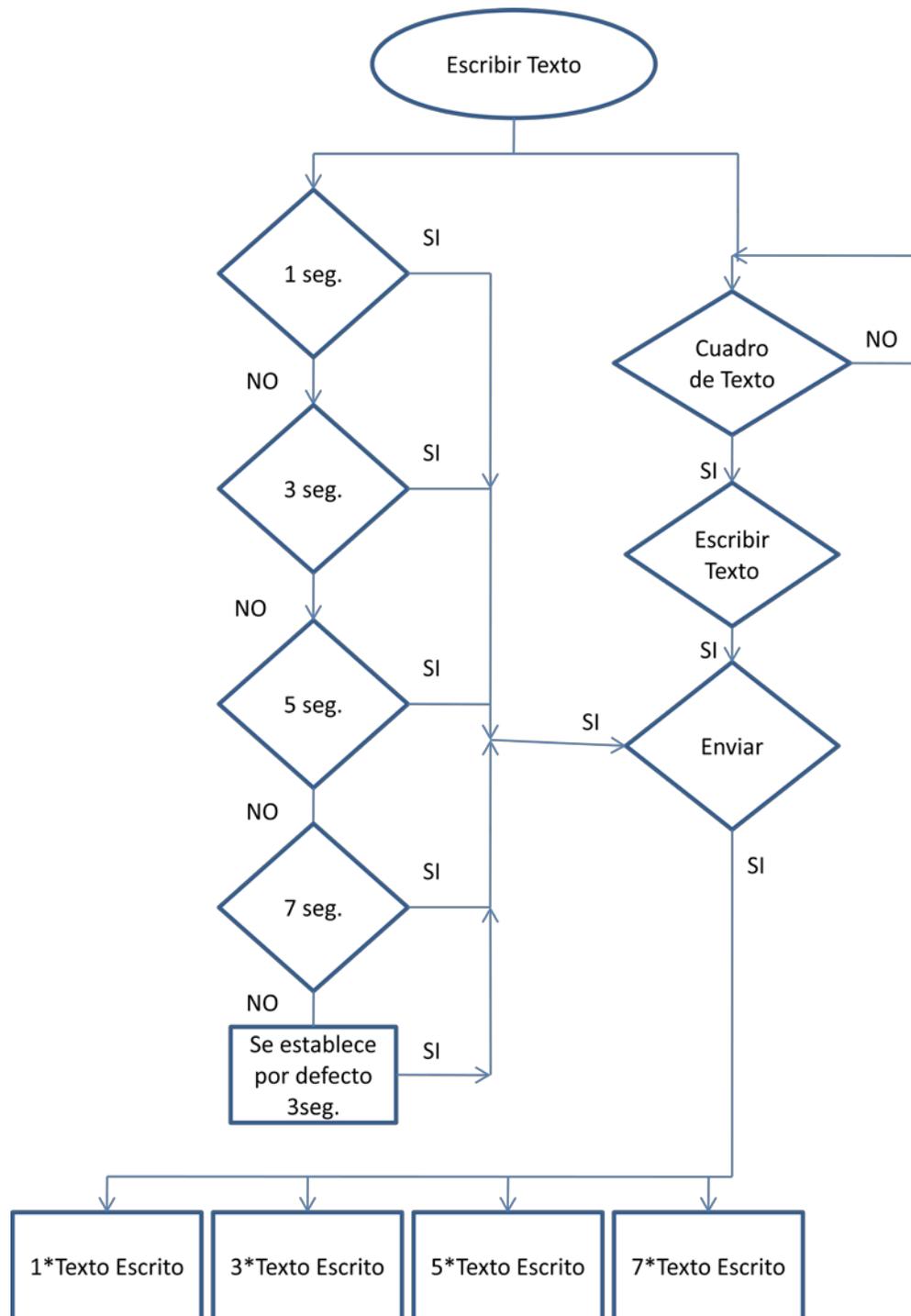


Figura 4.18 Diagrama para escribir el texto.



- Diagrama de Flujos para cargar texto de un archivo:

En el caso de que se quiera cargar un archivo ya existente en la memoria del teléfono móvil inteligente, con la finalidad de que el host no tenga que memorizar el texto y de esta forma evitar tener que parar las grabaciones por el hecho de que no recuerde algunas líneas. Existe la opción cuyo procedimiento es similar al anterior, con la diferencia de que se selecciona el botón “Desde Archivo” y el programa busca directamente en la memoria y carga el archivo en el cuadro de texto. Una vez cargado el archivo se selecciona la velocidad de aparición del texto en la pantalla del Teleprompter y se le da al botón enviar. Nuevamente, si no se le define la velocidad la aplicación automáticamente selecciona la de 3 segundos.

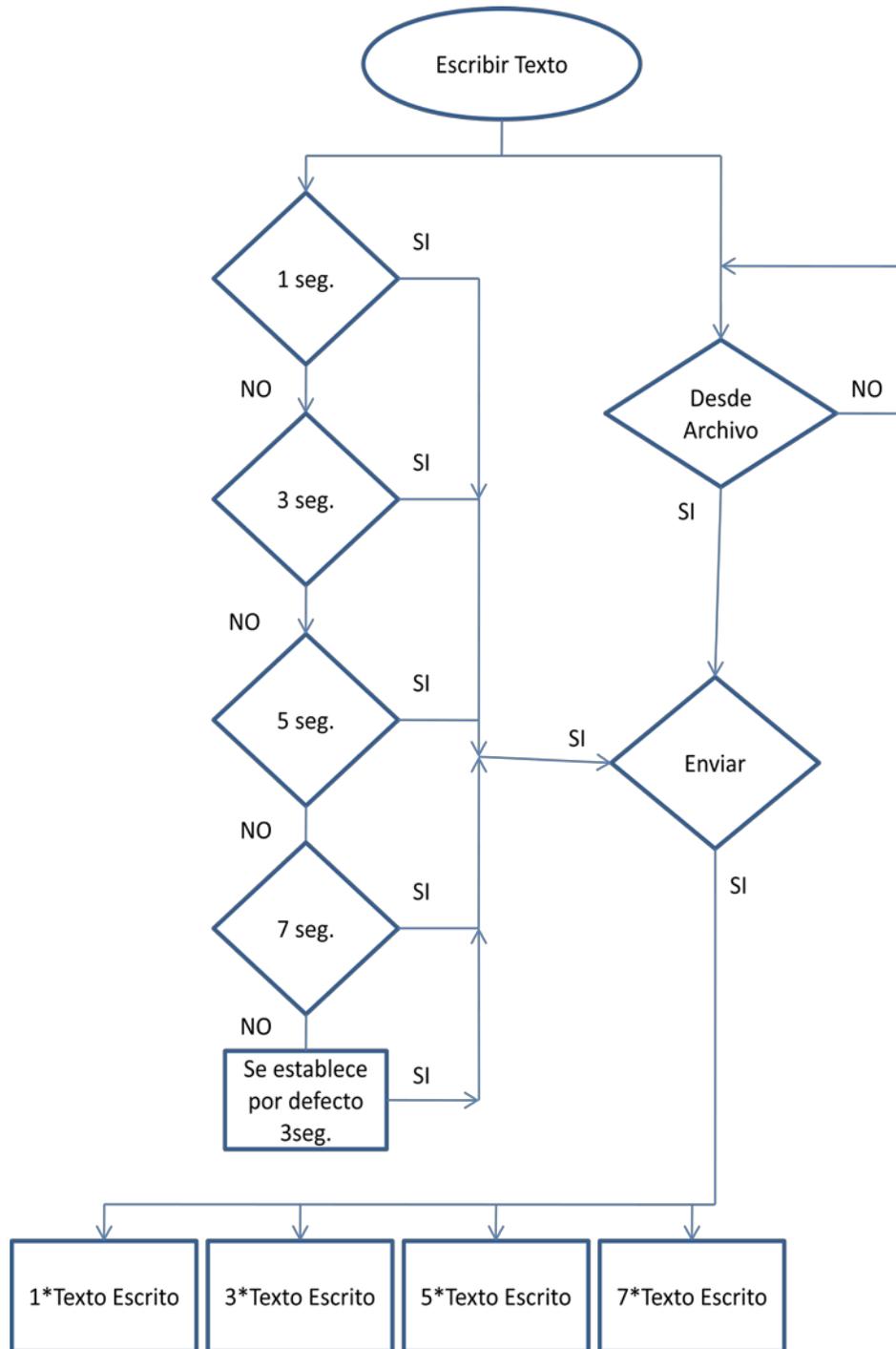


Figura 4.19 Diagrama para cargar el texto.



4.5 UNIDAD V: Elaboración de un software para el microcontrolador que reciba por medio del módulo Bluetooth la información que quiere ser visualizada en la pantalla LCD.

En esta fase se seleccionó el microcontrolador que se utilizó como medio de comunicación entre el módulo Bluetooth y la pantalla LCD gráfica para seguidamente comenzar con la programación del mismo.

La selección del dispositivo que actúa como elemento principal del sistema de recepción y envío del texto a la pantalla, se basó en lo establecido en el Capítulo I, donde se destacan como principales características que debe poseer dicho dispositivo es el de no presentar un elevado costo y adicionalmente debe tener la capacidad de manejar la cantidad de entradas y salidas requeridas en el programa. Por lo que se llegó a la conclusión de que el microcontrolador más adecuado que cumple con las características necesarias y suficientes para lo que se desea es el PIC18f2525 de la empresa Microchip, debido a su bajo costo y soporte de comunicación RS232, lo que le permite conectarse con el módulo Bluetooth de forma serial, además de que su disponibilidad en el mercado es amplia. En la tabla 4.2 se nombran algunos de los distintos microcontroladores de la gama PIC18 y sus características.

Tabla 4.2 Microcontroladores PIC18F2525/2620/4525/4620

Device	Program Memory		Data Memory		I/O	10-bit A/D (ch)	CCP/ ECCP (PWM)	MSSP		EUSART	Comp.	Timers 8/16-bit
	Flash (bytes)	# Single-Word Instructions	SRAM (bytes)	EEPROM (bytes)				SPI™	Master I ² C™			
PIC18F2525	48K	24576	3986	1024	25	10	2/0	Y	Y	1	2	1/3
PIC18F2620	64K	32768	3986	1024	25	10	2/0	Y	Y	1	2	1/3
PIC18F4525	48K	24576	3986	1024	36	13	1/1	Y	Y	1	2	1/3
PIC18F4620	64K	32768	3986	1024	36	13	1/1	Y	Y	1	2	1/3

Fuente: Data Sheet PIC18F2525

El microcontrolador PIC18F2525 posee 28 pines los cuales se encuentran distribuidos como se especifica en la figura siguiente:

28-Pin SPDIP, SOIC

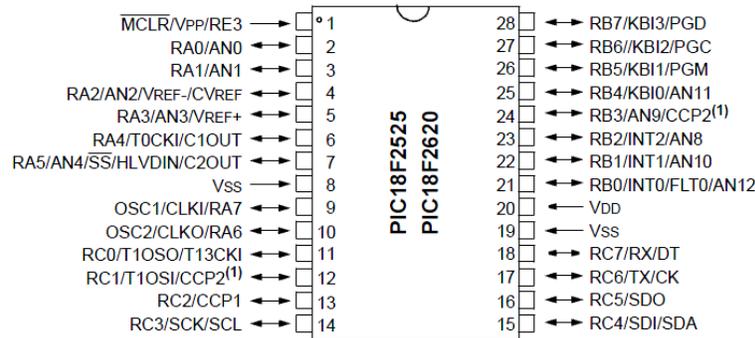


Figura 4.20 Pines del PIC18F2525.

Fuente: Data Sheet PIC18F2525

Diagrama de bloques:

En el siguiente diagrama de bloque se muestra un bosquejo general de las partes que componen el presente trabajo de grado.

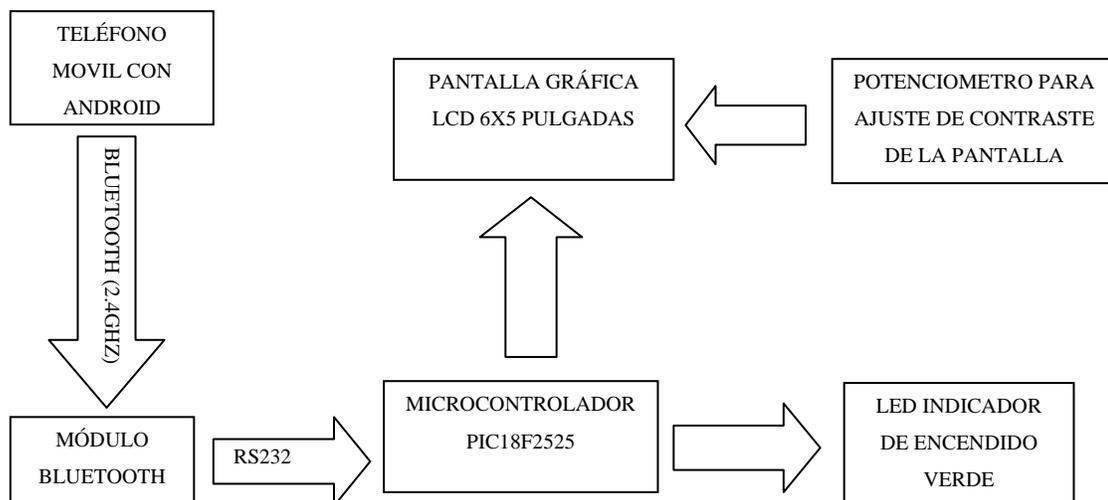


Figura 4.21 Diagrama de bloques del sistema en general.

Esquemáticos:

A continuación se presenta el esquemático del Teleprompter portátil con todos los bloques visiblemente identificados, además de la descripción y forma de conexión de dichos módulos.

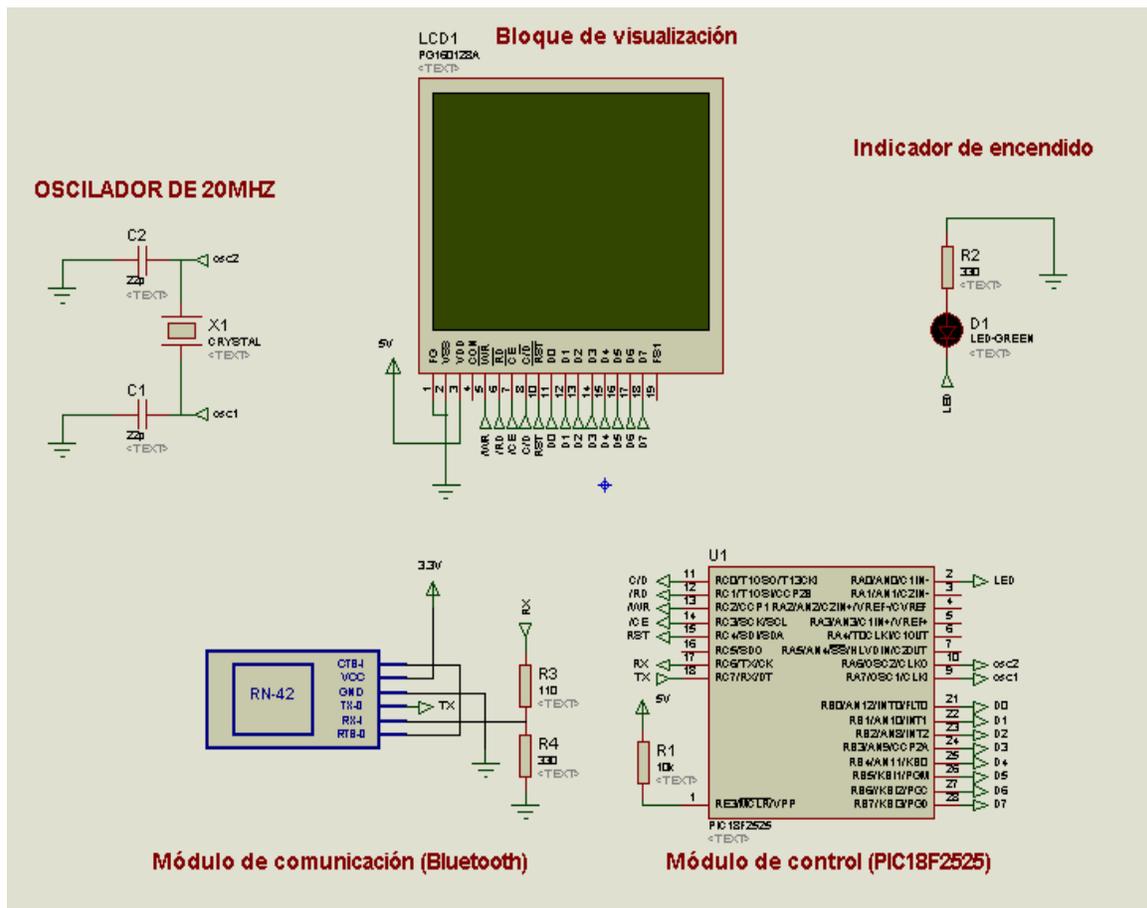


Figura 4.22 Circuito general del sistema.

Bloque de visualización: Está compuesto por la pantalla gráfica LCD de 160x128 pixel con controlador Toshiba T6963C. La pantalla LCD tiene un fondo azul y los píxeles se ven en blancos. La unidad viene con una luz de fondo LCD y regulador incorporado de la interfaz paralela y tanto la LCD como la luz de fondo son conducidos por 5V.

Para la realización de la conexión de la pantalla gráfica LCD con el PIC18F2525, se utilizaron las librerías existentes en mikrobasic las cuales facilitan la configuración de los pines y el Data Sheet (ver Anexo N.2) en donde se identifican y definen la función de cada pin que conforma la pantalla LCD.

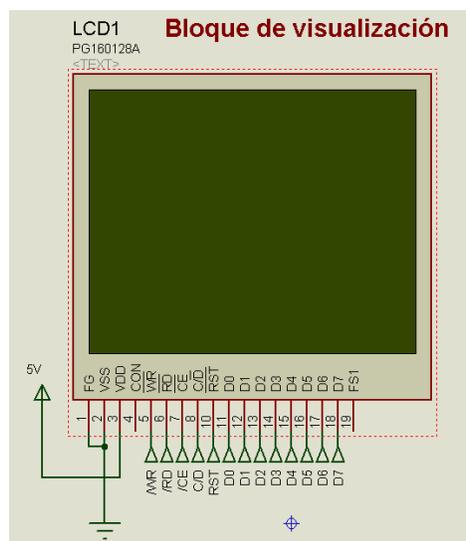


Figura 4.23 Módulo de visualización.

Bloque de control: conformado por el microcontrolador PIC18F2525 el cual es el componente principal del sistema ya que es el responsable de recibir el texto enviado por el módulo bluetooth y enviarlo al módulo de visualización. Su rango de tensión de funcionamiento está entre 2.0V a 5.5V.

Para la conexión con el bloque de visualización se utilizaron los puertos C y B, donde en el puerto C se realizó la inicialización de los pines de configuración mientras que en el puerto B la conexión del bus de dato que envía el texto a la pantalla.

En los pines RC6 y RC7 se conectaron los pines RX y TX respectivamente del módulo bluetooth, en donde el pin RC7 es el responsable de recibir los datos enviados al microcontrolador. Cuenta con un LED conectado al pin RA0 cuya función es indicar si el Teleprompter está encendido y con un cristal de 20MHZ (oscilador) conectado en los pines RA6 y RA7.

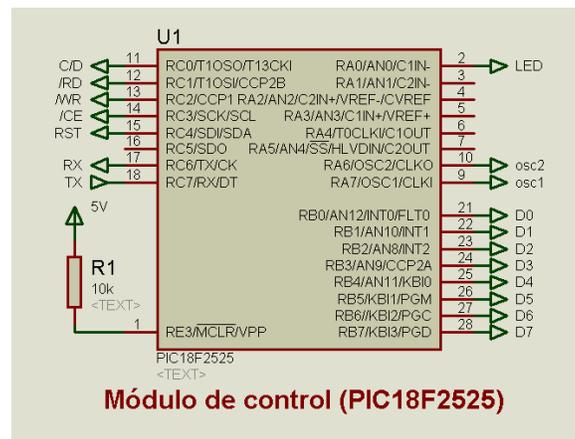


Figura 4.24 Módulo de Control.

Bloque de módulo de comunicación: Como ya se dijo previamente, la comunicación entre el microcontrolador y el celular se hace vía serial pero usando como medio de transmisión la conexión Bluetooth. Por lo tanto, es necesaria una interfaz entre estos dos dispositivos que pueda convertir la señal recibida de bluetooth a una señal serial que el microcontrolador pueda entender. En este proyecto se usó un módulo conversor Bluetooth RN-42 Class 2 Bluetooth Module. Este se muestra en la siguiente imagen.

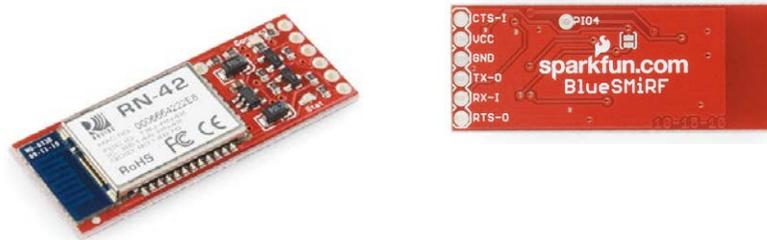


Figura 4.25 Módulo Bluetooth RN-42.

La conexión es realmente sencilla. Para explicar esta observe la distribución de pines en la figura 4.24. Se polariza el dispositivo con 3.3V entre los pines GND y VCC. Conecte entre sí los pines CTS-I y RTS-O pues no se va a hacer control de flujo en esta aplicación usando estos pines. Finalmente quedan dos pines disponibles RX-I y TX-O por los cuales se transmite la información serial. Los cuales se conectan RX-I del RN-42 al pin TX del micro, y TX-O al pin RX del micro.

Tenga en cuenta que este módulo viene programado para hacer comunicación serial a una tasa de 115200 baudios, por lo tanto se tiene que configurar dependiendo de la velocidad a la que se va a trabajar.

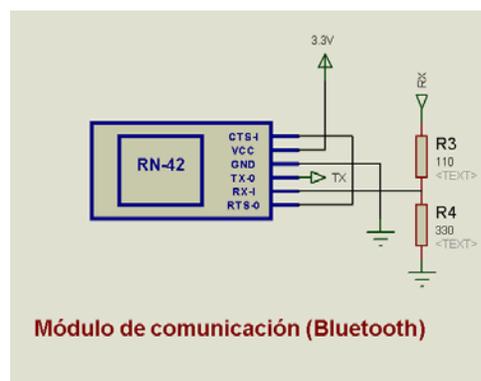


Figura 4.26 Módulo Bluetooth RN-42 en Proteus.

Lógica de programación en dispositivos programables:

A continuación se muestra un diagrama general donde se puede observar cuales son las funciones principales para la cual el microcontrolador está programado para realizar.

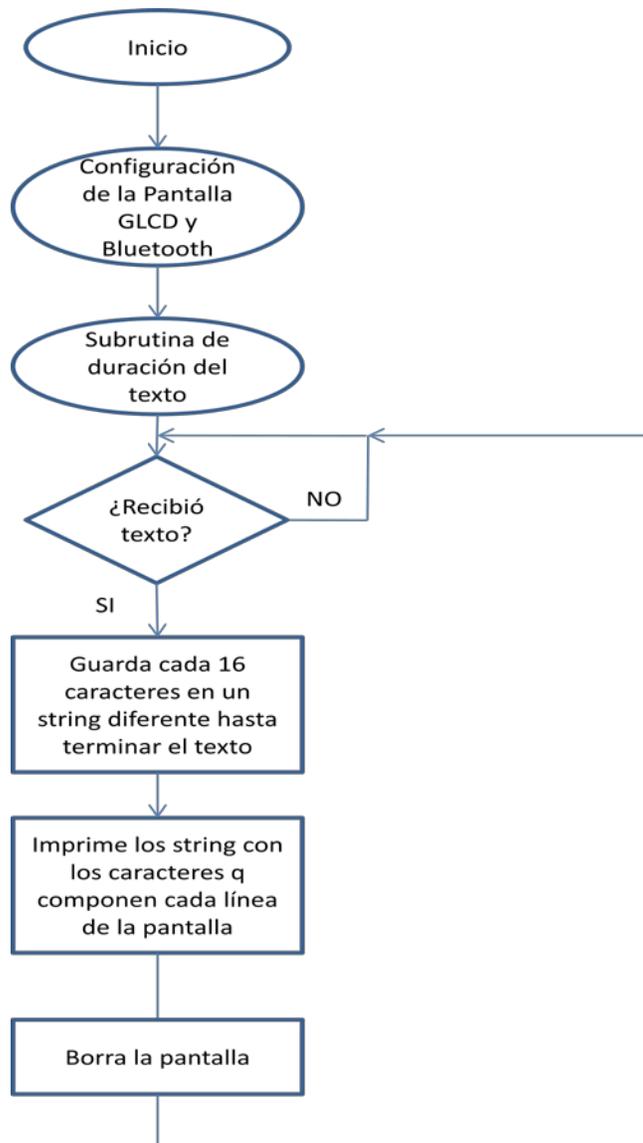


Figura 4.27 Diagrama de flujo general del microcontrolador.

- **Diagrama de flujo de la configuración de la pantalla LCD y el Bluetooth:**

Estas son las configuraciones iniciales que el PIC le indica a la pantalla y el módulo Bluetooth para poder cumplir con lo establecido en el diagrama general. Ajuste de comunicación en 19200 baudios, selección del fondo de la pantalla, inicialización de la LCD gráfica, ubicación del cursor.

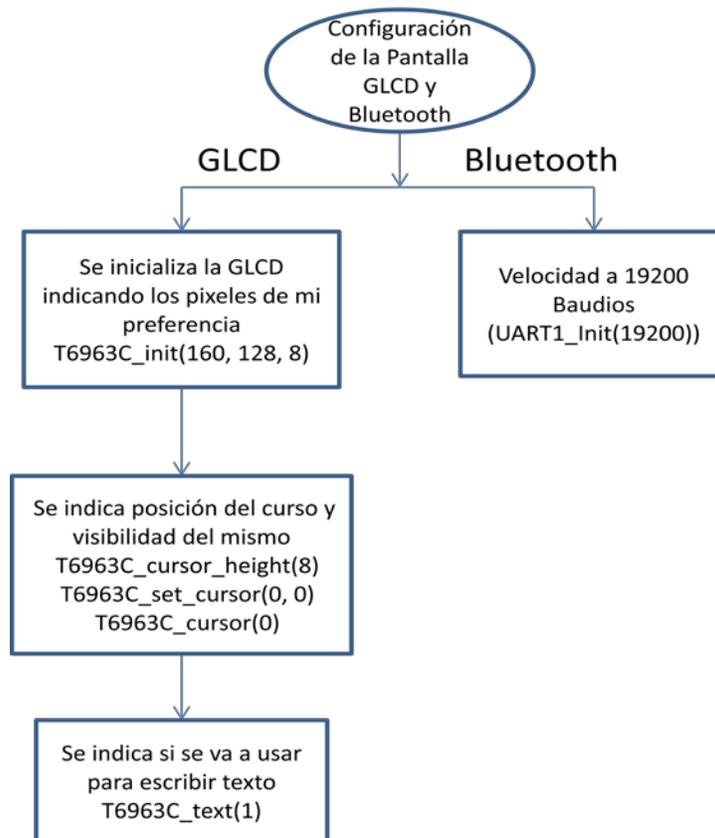


Figura 4.28 Diagrama configuración de GLCD y Bluetooth.

- **Diagrama de flujo de la subrutina de duración del texto en la pantalla:**

En esta etapa la subrutina se encarga de identificar la opción enviada por el teléfono móvil inteligente y aplicar un retardo en la aparición de cada línea de texto en la pantalla GLCD, dependiendo de dicha opción.

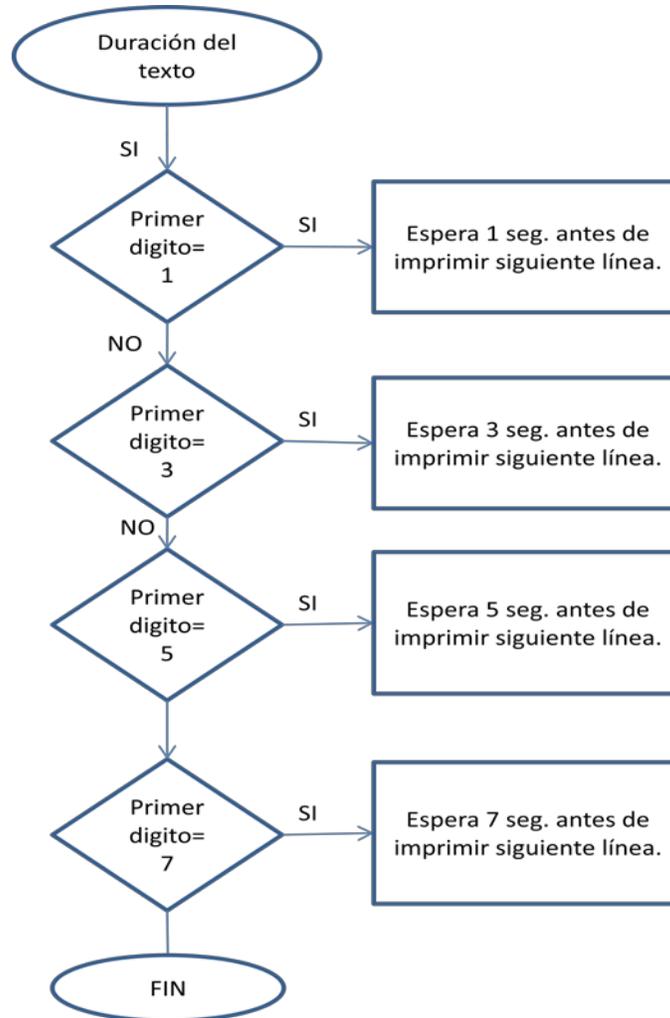


Figura 4.29 Diagrama de subrutina de retardo.

4.6 UNIDAD VI: Ensamble del proyecto final mediante la conexión de la pantalla LCD, el módulo Bluetooth y el microcontrolador en un solo circuito impreso.

Habiendo finiquitado el desarrollo de las herramientas y realizado la prueba finales del funcionamiento del equipo en la Empresa Caso Estudio, se procedió a realizar los circuitos integrados necesarios para la integración de todos los componentes, el diseño y elaboración del soporte a la cámara y la carcasa de protección del Teleprompter.

4.6.1 Circuito Integrado

Para el diseño de la placa del sistema impreso se realizó un primer esquema en tercera dimensión en el programa Eagle 3D para tener una visión de cómo quedara el diseño del circuito impreso final, además del silk screen y la PCB.

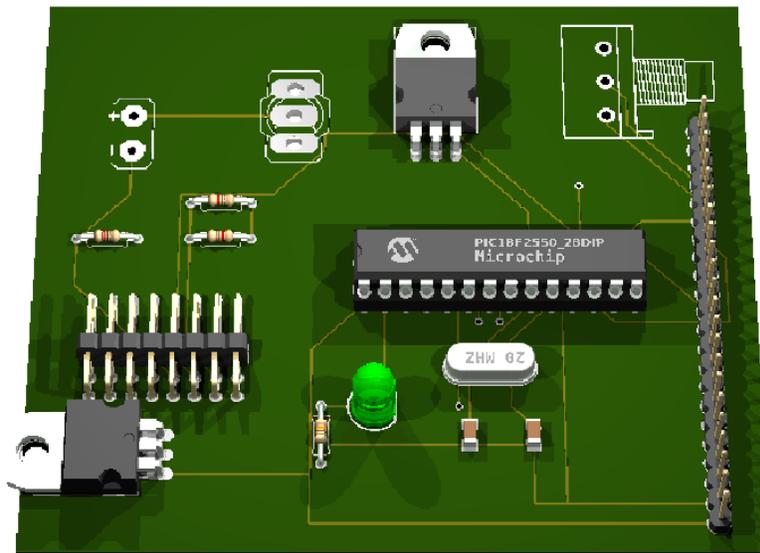


Figura 4.30 Diseño de la placa del circuito integrado en tercera dimensión

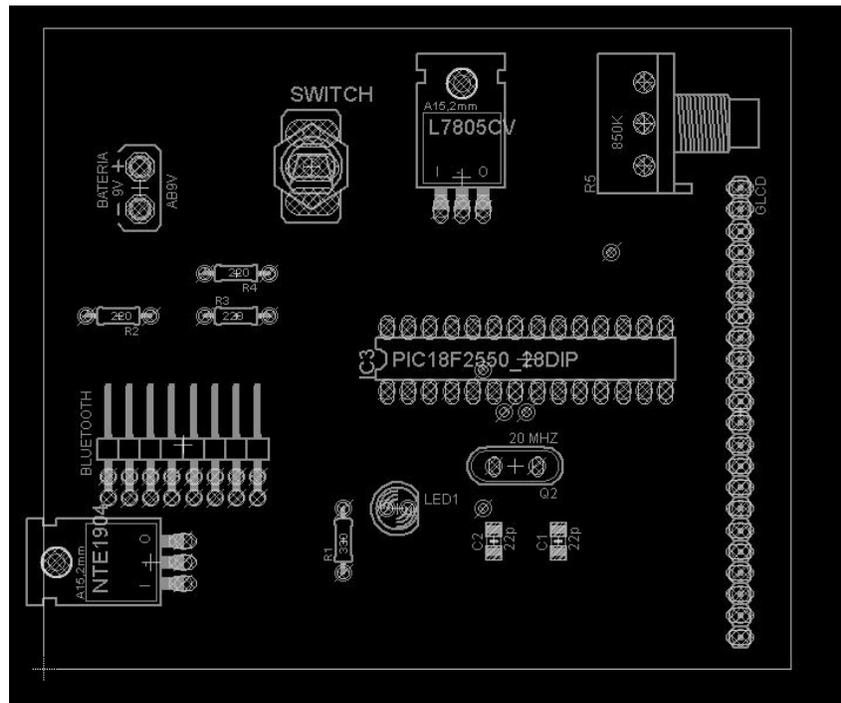


Figura 4.31 Imagen Stilk Screen del circuito impreso.

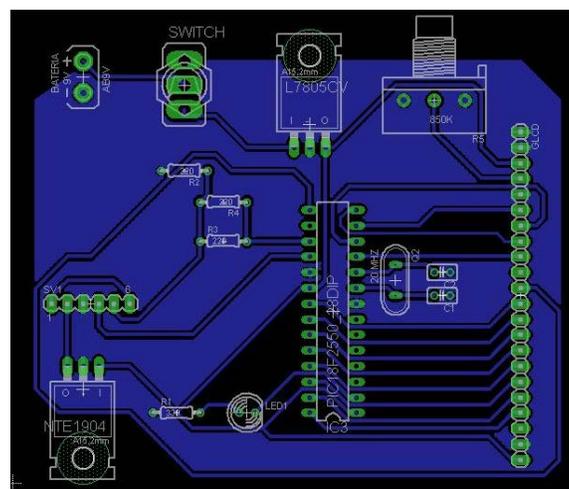


Figura 4.32 Diseño de la PCB.

4.6.2 Diseño de la carcasa y soporte a la cámara del Teleprompter Portátil

Soporte del Teleprompter:

El diseño del soporte del Teleprompter, cuya función es el de fijarlo a la cámara de video sin causar incomodidades al camarógrafo, se basó en una especie de brazo sujetador el cual se abre y cierra alrededor del cuello de la cámara aplicándole presión en uno de sus extremos por medio de un tornillo, para que de esta forma quede fijo a la cámara sin moverse. También posee un riel por el cual se desliza el Teleprompter que permite colocarlo en distintas posiciones a gusto del usuario.

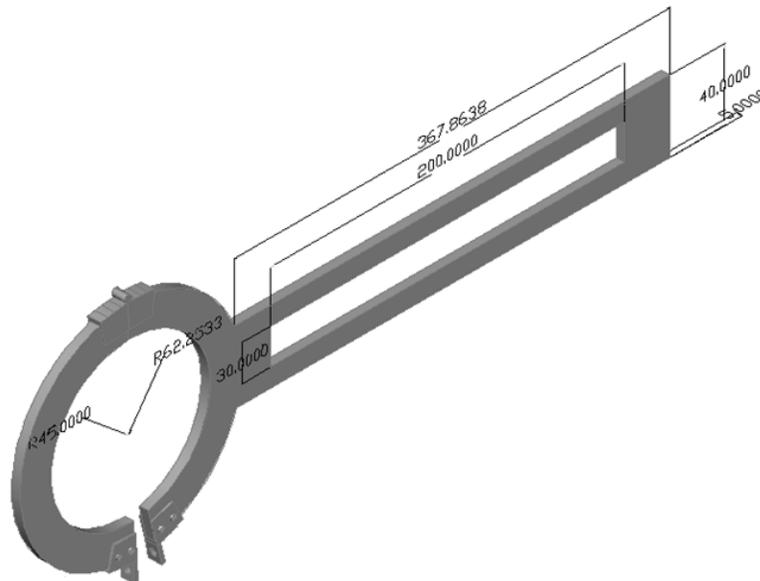


Figura 4.33 Soporte del Teleprompter Portátil.

Carcasa del Teleprompter:

El diseño de la carcasa se basó en las dimensiones de la pantalla LCD gráfica Toshiba, de los circuitos impresos y batería a utilizar, para que de esta forma no hubiese inconvenientes en la colocación de estos equipos en el interior de la misma.

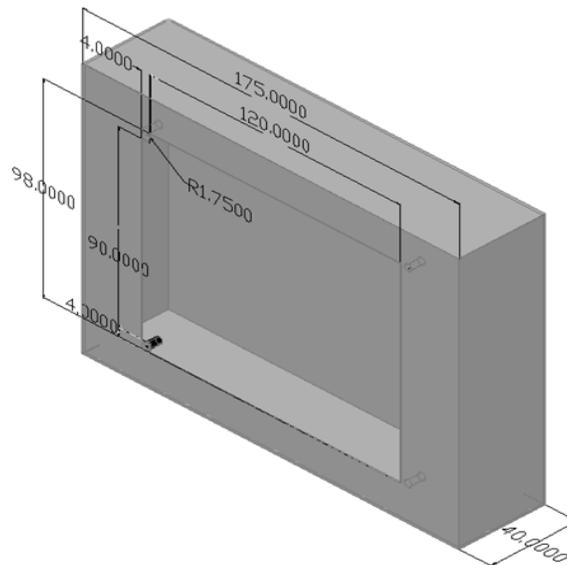


Figura 4.34 Carcasa del Teleprompter Portátil.

Isometría general de la carcasa y soporte del Teleprompter Pórtatil:

En las siguientes figuras se observa el resultado final de la carcasa y soporte del Teleprompter portátil, donde se puede ver que no implica muchos aparatos lo que lo convierte en un equipo fácil de trasladar y se puede instalar fácilmente a la cámara de video permitiendo colocar el brazo del soporte tanto vertical como horizontal dependiendo de la comodidad del usuario.

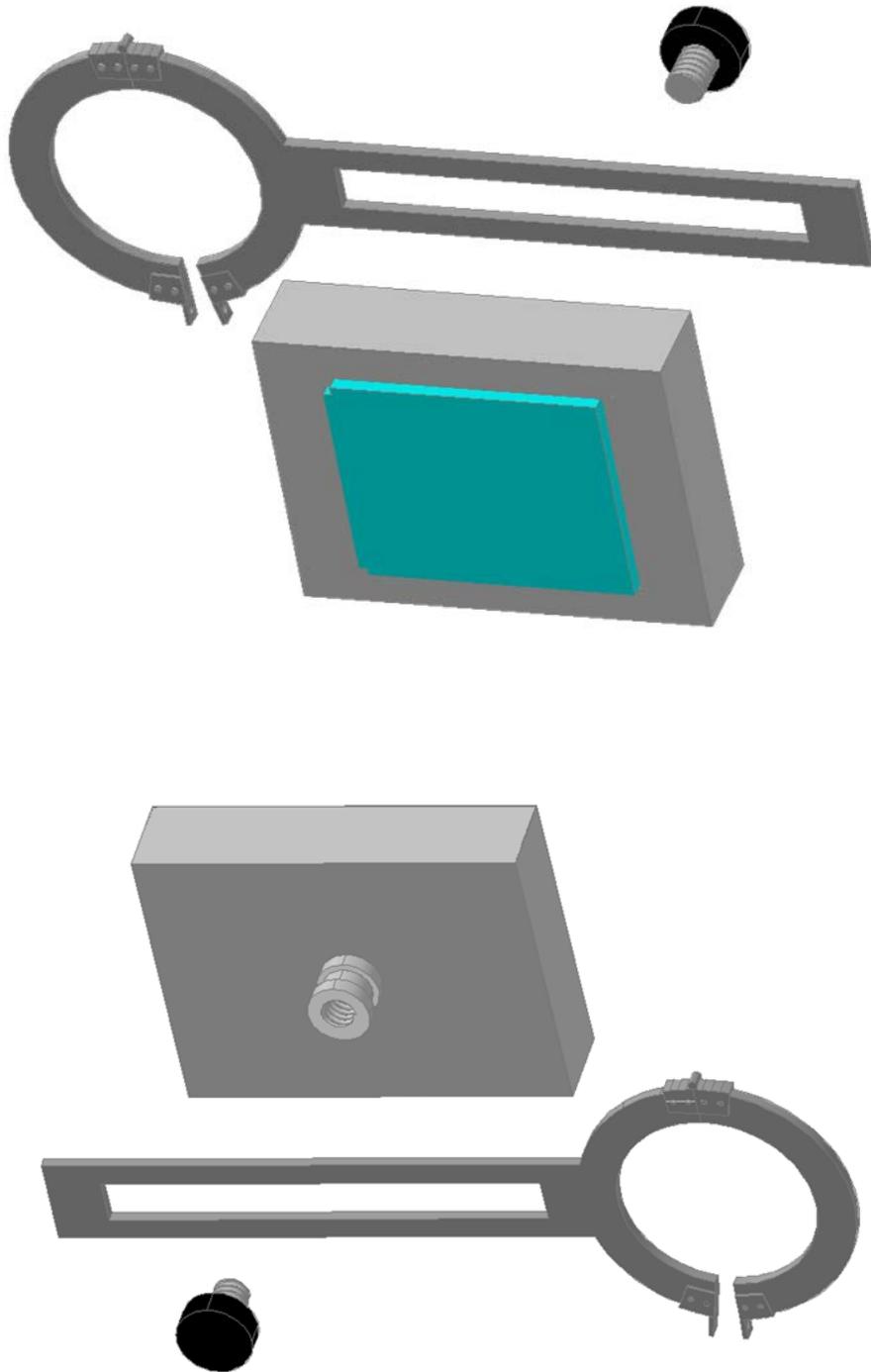


Figura 4.35 Partes del soporte y carcasa del Teleprompter Portátil.



CONCLUSIONES Y RECOMENDACIONES

CONCLUSIONES

Una empresa audiovisual bien equipada tecnológicamente posee una de las principales fortalezas que permite abrir las posibilidades de su consolidación en el mercado.

Los resultados obtenidos en el Proyecto de Grado describen la situación actual de algunas de las empresas audiovisuales con respecto a sus equipos de comunicación implementados en exteriores, la cual en muchos casos es precaria debido a que no poseen el capital necesario para adquirir dichos equipos y la disponibilidad en el mercado de los mismos es muy limitada, como es el caso del Teleprompter portátil.

El trabajo revela que la adquisición de un Teleprompter portátil puede ayudar exponencialmente a mejorar su posición con respecto a otras afines, ya que mejoran sus posibilidades de perfeccionar sus productos audiovisuales, ayudando a su crecimiento, desarrollo y rentabilidad.

El Teleprompter portátil presentado en este trabajo, ofrece a los directores y dueños de las empresas audiovisuales una herramienta de fácil adquisición y acertada, tanto para la comunicación entre el host (presentador) y el director en exteriores como para facilitarle el trabajo al host proporcionándole una herramienta que evita que tenga que memorizar el texto que se quiere decir.

El Teleprompter portátil resultó para la empresa caso estudio, muy sencillo de implantar, ya que es de fácil entendimiento por las herramientas donde está presentado. La implantación del mismo, logró en los directivos una visión más clara de hacia dónde se dirige la empresa y los efectos que producen la falta de estos equipos.



En ese sentido, más allá de las acciones que el Teleprompter pueda emprender en la comunicación en exteriores, deben emplear equipos nuevos que les permitan mejorar su desempeño como empresa.

RECOMENDACIONES

- Desarrollar un programa en lenguaje de alto nivel (Mikrobasic, MikroC o similar) que me permita voltear los caracteres de forma que se vean como en un espejo, para que de esta manera al colocar un vidrio polarizado en el lente de la cámara con un cierto ángulo se refleje el texto de la pantalla de forma correcta. Esto es con la finalidad de que el presentador o presentadora no tenga que desviar la mirada del lente de la cámara para leer la información.
- Elaborar una aplicación que permita ampliar o reducir el tamaño de los caracteres de forma que el lector no tenga que forzar la vista para leer el texto.
- Colocar una memoria externa que me permita almacenar textos de grandes tamaños.



REFERENCIAS BIBLIOGRÁFICAS

[1] Apodax. [Datos en línea]. Disponible: <http://www.apodax.com/teleprompter-CT58.html>
[Consulta: Noviembre de 2011].

[2] Aranaz, J. (2009). *Desarrollo de Aplicaciones para Dispositivos Móviles sobre la Plataforma Android de Google*. Trabajo de grado de licenciatura no publicado, Universidad Carlos III de Madrid, Madrid.

[3] Lago, R. (2008). *Proteus VSM*. [Documento en línea]. Disponible: http://w3.ing.uc.edu.ve/Members/81ersi/Lab_Micro/Guia-Proteus.pdf/view. [Consulta: Octubre de 2011].

[4] Pérez G (2008). *Aprendiendo Java y Programación Orientada a Objetos*. [Documento en línea]. Disponible: <http://twileshare.com/uploads/AprendiendoJava-y-POO.pdf>. [Consulta: Abril de 2012].

[5] Bellido F, Torres M (2004). *Comunicación Inalámbrica con Bluetooth*. [Documento en línea]. Disponible: <http://www.tecnicaindustrial.es/tiadmin/81ersión/15/06/a06.pdf>.

[6] Alvarez G, Pico J (2004), *Desarrollo de un Protocolo para Comunicaciones Inalámbricas e Implementación en Dispositivos de Lógica reconfigurable*. Trabajo de grado de licenciatura no publicado, Facultad de Informática, San Sebastián, España.

[7] Aguayo, P. (2004). *Introducción al Microcontrolador*. [Documento en línea]. Disponible: <http://www.olimex.cl/tutorial/tutorial1.pdf>. [Consulta: Noviembre de 2011].



[8] *Redes Wireless*. (s.f.). [Documento en línea]. Disponible: http://www.redsinfronteras.org/pdf/redes_wireless.pdf. [Consulta: Enero de 2012].

[9] *Comunicaciones Inalámbricas*. (2005). [Documento en línea]. Disponible: <http://www.ica.luz.ve/cstufano/CursoDeRedes/Wirelessdoc.pdf>. [Consulta: Enero de 2012].

[10] Bisquerra, R. (1989). *Métodos de investigación educativa: Guía práctica*. Trabajo de grado de licenciatura no publicado, Universidad Nacional Abierta, Caracas, Venezuela.

[11] Universidad Pedagógica Experimental Libertador (1998). *Manual de Trabajo de Grado de Especialización, Maestría y Tesis Doctorales*. Universidad Pedagógica Experimental Libertador (UPEL).

[12] *I+D Electrónica*. [Datos en línea]. Disponible: http://www.didacticaselectronicas.com/index.php?page=shop.product_details&flypage=flypage.tpl&product_id=402&category_id=152&option=com_virtuemart&Itemid=73&vmchk=1&Itemid=73. [Consulta: Octubre de 2012].

[13] *Android* [Documento en línea]. Disponible: <http://developer.Android.com/index.html>. [Consulta: Febrero de 2013].

[14] Borches, P. (2004). *Java 2 Micro Edition Soporte Bluetooth 82ersión 1.0*. [Documento en línea]. Disponible: http://www.it.uc3m.es/celestec/docencia/j2me/tutoriales/bluetooth/EstudioTecnologico1_0.pdf. [Consulta: Febrero de 2013].



-
- [15] *Protocolo Bluetooth*. [Documento en línea]. Disponible:
<http://ants.dif.um.es/~felixgm/docencia/android/resources/ApendicesBibliografia.pdf>.
[Consulta: Marzo de 2013].



ANEXO NRO. 1.
Instrumento de Levantamiento de
Información.



Buenos días / Buenas tardes.

Quisiera, en primer lugar, agradecerle su colaboración con este Trabajo Especial de Grado.

A continuación procederemos a realizarle una serie de preguntas relacionadas con los equipos de comunicación implementados en su empresa, las cuales tienen por finalidad identificar el estatus de los avances tecnológicos en esta organización.

Tenga plena confianza en que la información levantada mediante esta encuesta será tratada con la mayor discreción y confidencialidad.

INSTRUMENTO DE LEVANTAMIENTO DE INFORMACION:

1. Información General de la empresa:

- 1.1. Nombre:
- 1.2. Ubicación: Urb. _____, Ciudad _____.
- 1.3. Cantidad de empleados (entre):
1 y 5 _____, 6 y 10 _____, 11 y 15 _____, 15 y 20 _____, 20 y más _____, NS / NC _____.
- 1.4. Cantidad de clientes regulares (entre):
1 y 5 _____, 6 y 10 _____, 11 y 15 _____, 15 y 20 _____, 20 y más _____, NS / NC _____.

2. Información de los equipos de comunicación de la empresa:

- 2.1. ¿Qué implica realizar grabaciones en exteriores a diferencia de las grabaciones en estudio?

Más equipos de trabajo_____, Menos equipos de trabajo_____.

Explique:



2.2. ¿Qué cantidad de equipos se necesitan en las grabaciones en exteriores? Entre:

1 a 5_____, 5 a 10_____, 10 a 15_____, Mas de 15_____.

2.3. ¿Opina que los Teleprompters podrían ser una herramienta útil en grabaciones en exteriores?

Si _____, No _____.

2.4. Si su respuesta anterior fue positiva, ¿por qué piensa que sería útil el uso de Teleprompters en exteriores?

Fluidez en las grabaciones_____,

Menos repeticiones de grabaciones_____,

Evita tener que memorizar el texto_____,

Todas las anteriores_____,

Ninguna de las anteriores_____.

2.5. ¿Qué evita que todas las empresas audiovisuales cuenten con un Teleprompter Portátil?

Alto costo_____,

Ya poseen muchos equipos_____,

Poco conocimiento en el uso del Teleprompters _____,

Todas las anteriores_____.



2.6. ¿Cuáles son los métodos más utilizados actualmente para solventar la falta del Teleprompter en exterior?

Papel craft_____, Pizarra acrílica_____, Papel bond_____, Otros_____.

2.7. ¿Cuál es la distancia que debe existir entre el host y la cámara en exteriores?

1 metro_____,

2 metros_____,

3 metros_____,

4 metros_____,

Más de 4 metros_____.

2.8. ¿Qué características debería tener un Teleprompter Portátil para ser más eficaz?

Liviano_____,

El texto pueda ser cargado de forma rápida y fácil_____,

Se pueda sujetar a la cámara_____,

Todas las anteriores_____.

Una vez más, muchas gracias por su colaboración.



ANEXO NRO. 2.

**Data Sheet de los principales
componentes del sistema.**



Data Sheet PIC18F2525

MICROCHIP **PIC18F2525/2620/4525/4620**

28/40/44-Pin Enhanced Flash Microcontrollers with 10-Bit A/D and nanoWatt Technology

Power Managed Modes:

- Run: CPU on, peripherals on
- Idle: CPU off, peripherals on
- Sleep: CPU off, peripherals off
- Idle mode currents down to 2.5 μ A typical
- Sleep mode current down to 100 nA typical
- Timer1 Oscillator: 1.8 μ A, 32 kHz, 2V
- Watchdog Timer: 1.4 μ A, 2V typical
- Two-Speed Oscillator Start-up

Flexible Oscillator Structure:

- Four Crystal modes, up to 40 MHz
- 4x Phase Lock Loop (PLL) – available for crystal and internal oscillators
- Two External RC modes, up to 4 MHz
- Two External Clock modes, up to 40 MHz
- Internal oscillator block:
 - 8 user selectable frequencies, from 31 kHz to 8 MHz
 - Provides a complete range of clock speeds from 31 kHz to 32 MHz when used with PLL
 - User tunable to compensate for frequency drift
- Secondary oscillator using Timer1 @ 32 kHz
- Fail-Safe Clock Monitor
 - Allows for safe shutdown if peripheral clock stops

Peripheral Highlights:

- High-current sink/source 25 mA/25 mA
- Three programmable external interrupts
- Four input change interrupts
- Up to 2 Capture/Compare/PWM (CCP) modules, one with Auto-Shutdown (28-pin devices)
- Enhanced Capture/Compare/PWM (ECCP) module (40/44-pin devices only):
 - One, two or four PWM outputs
 - Selectable polarity
 - Programmable dead time
 - Auto-Shutdown and Auto-Restart

Peripheral Highlights (Continued):

- Master Synchronous Serial Port (MSSP) module supporting 3-wire SPI™ (all 4 modes) and I²C™ Master and Slave modes
- Enhanced Addressable USART module:
 - Supports RS-485, RS-232 and LIN 1.2
 - RS-232 operation using internal oscillator block (no external crystal required)
 - Auto-Wake-up on Start bit
 - Auto-Baud Detect
- 10-bit, up to 13-channel Analog-to-Digital Converter module (A/D):
 - Auto-acquisition capability
 - Conversion available during Sleep
- Dual analog comparators with input multiplexing
- Programmable 16-level High/Low-Voltage Detection (HLVD) module:
 - Supports interrupt on High/Low-Voltage Detection

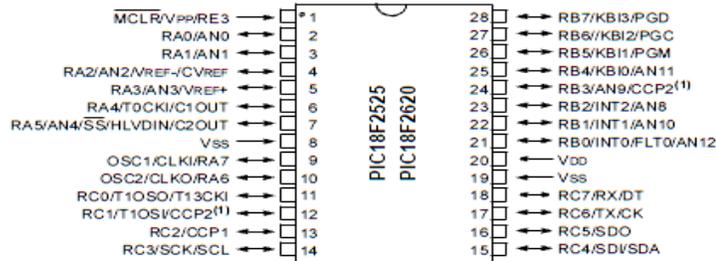
Special Microcontroller Features:

- C compiler optimized architecture:
 - Optional extended instruction set designed to optimize re-entrant code
- 100,000 erase/write cycle Enhanced Flash program memory typical
- 1,000,000 erase/write cycle Data EEPROM memory typical
- Flash/Data EEPROM Retention: 100 years typical
- Self-programmable under software control
- Priority levels for interrupts
- 8 x 8 Single Cycle Hardware Multiplier
- Extended Watchdog Timer (WDT):
 - Programmable period from 4 ms to 131s
- Single-supply 5V In-Circuit Serial Programming™ (ICSP™) via two pins
- In-Circuit Debug (ICD) via two pins
- Wide operating voltage range: 2.0V to 5.5V
- Programmable Brown-out Reset (BOR) with software enable option

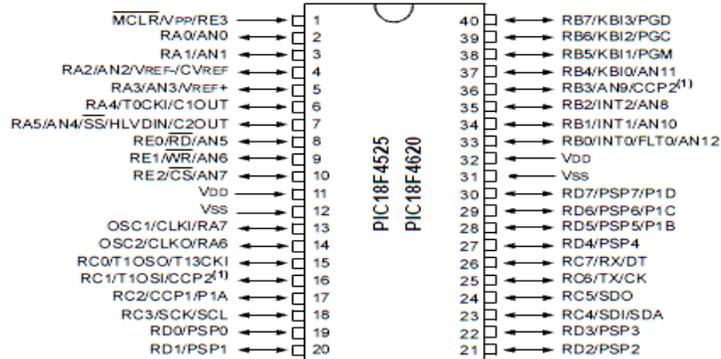
Device	Program Memory		Data Memory		I/O	10-bit A/D (ch)	CCP/ ECCP (PWM)	MSSP		USART	Comp.	Timers 8/16-bit
	Flash (bytes)	# Single-Word Instructions	SRAM (bytes)	EEPROM (bytes)				SPI™	Master I ² C™			
PIC18F2525	48K	24576	3986	1024	25	10	2/0	Y	Y	1	2	1/3
PIC18F2620	64K	32768	3986	1024	25	10	2/0	Y	Y	1	2	1/3
PIC18F4525	48K	24576	3986	1024	36	13	1/1	Y	Y	1	2	1/3
PIC18F4620	64K	32768	3986	1024	36	13	1/1	Y	Y	1	2	1/3

Pin Diagrams

28-Pin SPDIP, SOIC



40-Pin PDIP



Note 1: RB3 is the alternate pin for CCP2 multiplexing.



TABLE 1-1: DEVICE FEATURES

Features	PIC18F2525	PIC18F2620	PIC18F4525	PIC18F4620
Operating Frequency	DC – 40 MHz			
Program Memory (Bytes)	49152	65536	49152	65536
Program Memory (Instructions)	24576	32768	24576	32768
Data Memory (Bytes)	3968	3968	3968	3968
Data EEPROM Memory (Bytes)	1024	1024	1024	1024
Interrupt Sources	19	19	20	20
I/O Ports	Ports A, B, C, (E)	Ports A, B, C, (E)	Ports A, B, C, D, E	Ports A, B, C, D, E
Timers	4	4	4	4
Capture/Compare/PWM Modules	2	2	1	1
Enhanced Capture/Compare/ PWM Modules	0	0	1	1
Serial Communications	MSSP, Enhanced USART	MSSP, Enhanced USART	MSSP, Enhanced USART	MSSP, Enhanced USART
Parallel Communications (PSP)	No	No	Yes	Yes
10-bit Analog-to-Digital Module	10 Input Channels	10 Input Channels	13 Input Channels	13 Input Channels
Resets (and Delays)	POR, BOR, RESET Instruction, Stack Full, Stack Underflow (PWRT, OST), MCLR (optional), WDT			
Programmable Low-Voltage Detect	Yes	Yes	Yes	Yes
Programmable Brown-out Reset	Yes	Yes	Yes	Yes
Instruction Set	75 Instructions; 83 with Extended Instruction Set enabled			
Packages	28-pin SPDIP 28-pin SOIC	28-pin SPDIP 28-pin SOIC	40-pin PDIP 44-pin QFN 44-pin TQFP	40-pin PDIP 44-pin QFN 44-pin TQFP

FIGURE 1-1: PIC18F2525/2620 (28-PIN) BLOCK DIAGRAM

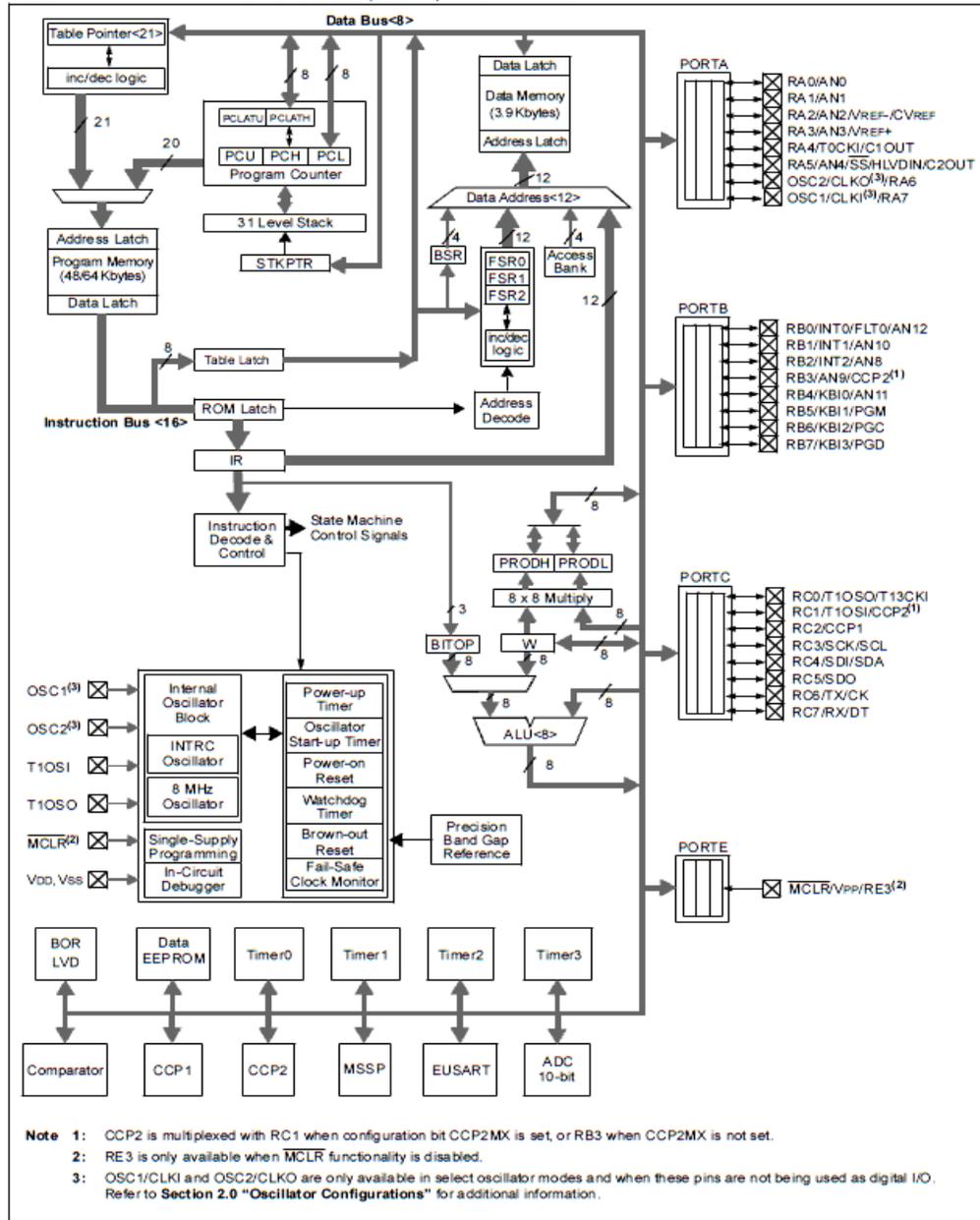




TABLE 1-3: PIC18F4525/4620 PINOUT I/O DESCRIPTIONS (CONTINUED)

Pin Name	Pin Number			Pin Type	Buffer Type	Description	
	PDIP	QFN	TQFP				
RA0/AN0	2	19	19	I/O	TTL	PORTA is a bidirectional I/O port. Digital I/O.	
RA0				I	Analog		Analog input 0.
AN0							
RA1/AN1	3	20	20	I/O	TTL	Digital I/O.	
RA1				I	Analog		Analog input 1.
AN1							
RA2/AN2/VREF-/CVREF	4	21	21	I/O	TTL	Digital I/O.	
RA2				I	Analog		Analog input 2.
VREF-				I	Analog		A/D reference voltage (low) input.
CVREF				O	Analog		Comparator reference voltage output.
RA3/AN3/VREF+	5	22	22	I/O	TTL	Digital I/O.	
RA3				I	Analog		Analog input 3.
AN3				I	Analog		A/D reference voltage (high) input.
VREF+							
RA4/T0CKI/C1OUT	6	23	23	I/O	ST	Digital I/O.	
RA4				I	ST		Timer0 external clock input.
T0CKI							
C1OUT				O	—		Comparator 1 output.
RA5/AN4/SS/HLVDIN/C2OUT	7	24	24	I/O	TTL	Digital I/O.	
RA5				I	Analog		Analog input 4.
AN4				I	TTL		SPI™ slave select input.
SS				I	Analog		High/Low-Voltage Detect input.
HLVDIN							
C2OUT				O	—		Comparator 2 output.
RA6						See the OSC2/CLKO/RA6 pin.	
RA7						See the OSC1/CLKI/RA7 pin.	

Legend: TTL = TTL compatible input CMOS = CMOS compatible input or output
 ST = Schmitt Trigger input with CMOS levels I = Input
 O = Output P = Power

Note 1: Default assignment for CCP2 when configuration bit CCP2MX is set.
2: Alternate assignment for CCP2 when configuration bit CCP2MX is cleared.



TABLE 1-3: PIC18F4525/4620 PINOUT I/O DESCRIPTIONS (CONTINUED)

Pin Name	Pin Number			Pin Type	Buffer Type	Description
	PDIP	QFN	TQFP			
RB0/INT0/FLT0/AN12	33	9	8	I/O	TTL	PORTB is a bidirectional I/O port. PORTB can be software programmed for internal weak pull-ups on all inputs. Digital I/O. External interrupt 0. PWM Fault input for Enhanced CCP1. Analog input 12.
RB0				I	ST	
INT0				I	ST	
FLT0				I	Analog	
AN12						
RB1/INT1/AN10	34	10	9	I/O	TTL	Digital I/O. External interrupt 1. Analog input 10.
RB1				I	ST	
INT1				I	Analog	
AN10						
RB2/INT2/AN8	35	11	10	I/O	TTL	Digital I/O. External interrupt 2. Analog input 8.
RB2				I	ST	
INT2				I	Analog	
AN8						
RB3/AN9/CCP2	36	12	11	I/O	TTL	Digital I/O. Analog input 9. Capture 2 input/Compare 2 output/PWM 2 output.
RB3				I	Analog	
AN9				I/O	ST	
CCP2 ⁽¹⁾						
RB4/KBI0/AN11	37	14	14	I/O	TTL	Digital I/O. Interrupt-on-change pin. Analog input 11.
RB4				I	TTL	
KBI0				I	Analog	
AN11						
RB5/KBI1/PGM	38	15	15	I/O	TTL	Digital I/O. Interrupt-on-change pin. Low-Voltage ICSP™ Programming enable pin.
RB5				I	TTL	
KBI1				I/O	ST	
PGM						
RB6/KBI2/PGC	39	16	16	I/O	TTL	Digital I/O. Interrupt-on-change pin. In-Circuit Debugger and ICSP programming clock pin.
RB6				I	TTL	
KBI2				I/O	ST	
PGC						
RB7/KBI3/PGD	40	17	17	I/O	TTL	Digital I/O. Interrupt-on-change pin. In-Circuit Debugger and ICSP programming data pin.
RB7				I	TTL	
KBI3				I/O	ST	
PGD						

Legend: TTL = TTL compatible input CMOS = CMOS compatible input or output
 ST = Schmitt Trigger input with CMOS levels I = Input
 O = Output P = Power

Note 1: Default assignment for CCP2 when configuration bit CCP2MX is set.
Note 2: Alternate assignment for CCP2 when configuration bit CCP2MX is cleared.



TABLE 1-3: PIC18F4525/4620 PINOUT I/O DESCRIPTIONS (CONTINUED)

Pin Name	Pin Number			Pin Type	Buffer Type	Description
	PDIP	QFN	TQFP			
RD0/PSP0 RD0 PSP0	19	38	38	I/O I/O	ST TTL	PORTD is a bidirectional I/O port or a Parallel Slave Port (PSP) for interfacing to a microprocessor port. These pins have TTL input buffers when the PSP module is enabled. Digital I/O. Parallel Slave Port data.
RD1/PSP1 RD1 PSP1	20	39	39	I/O I/O	ST TTL	Digital I/O. Parallel Slave Port data.
RD2/PSP2 RD2 PSP2	21	40	40	I/O I/O	ST TTL	Digital I/O. Parallel Slave Port data.
RD3/PSP3 RD3 PSP3	22	41	41	I/O I/O	ST TTL	Digital I/O. Parallel Slave Port data.
RD4/PSP4 RD4 PSP4	27	2	2	I/O I/O	ST TTL	Digital I/O. Parallel Slave Port data.
RD5/PSP5/P1B RD5 PSP5 P1B	28	3	3	I/O I/O O	ST TTL —	Digital I/O. Parallel Slave Port data. Enhanced CCP1 output.
RD6/PSP6/P1C RD6 PSP6 P1C	29	4	4	I/O I/O O	ST TTL —	Digital I/O. Parallel Slave Port data. Enhanced CCP1 output.
RD7/PSP7/P1D RD7 PSP7 P1D	30	5	5	I/O I/O O	ST TTL —	Digital I/O. Parallel Slave Port data. Enhanced CCP1 output.

Legend: TTL = TTL compatible input CMOS = CMOS compatible input or output
 ST = Schmitt Trigger input with CMOS levels I = Input
 O = Output P = Power

Note 1: Default assignment for CCP2 when configuration bit CCP2MX is set.
2: Alternate assignment for CCP2 when configuration bit CCP2MX is cleared.

Data Sheet Módulo Bluetooth RN-42



RN-42/RN-42-N Data Sheet

www.rovingnetworks.com

DS-RN42-V1.0 12/6/2010

Class 2 Bluetooth® Module



Features

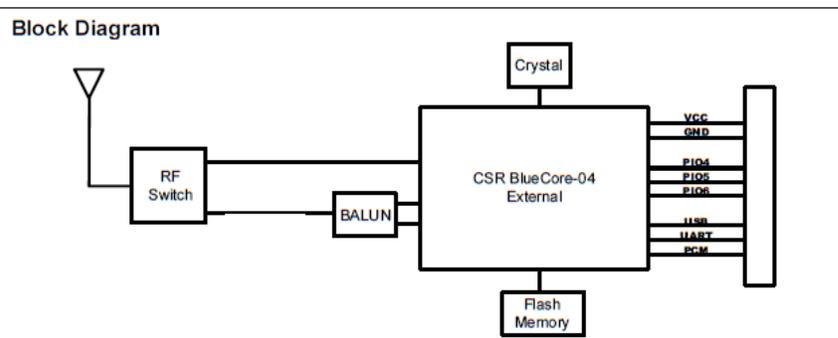
- Fully qualified Bluetooth 2.1/2.0/1.2/1.1 module
- Bluetooth v2.0+EDR support
- Available with on board chip antenna (RN-42) and without antenna (RN-42-N)
- Postage stamp sized form factor, 13.4mm x 25.8 mm x 2mm (RN-42) and 13.4mm x 20 mm x 2 mm (RN-42-N)
- Low power (*26uA sleep, 3mA connected, 30mA transmit*)
- UART (SPP or HCI) and USB (HCI only) data connection interfaces.
- Sustained SPP data rates - 240Kbps (slave), 300Kbps (master)
- HCI data rates - 1.5Mbps sustained, 3.0Mbps burst in HCI mode
- Embedded Bluetooth stack profiles included (*requires no host stack*): GAP, SDP, RFCOMM and L2CAP protocols, with SPP and DUN profile support.
- Bluetooth SIG certified
- Castellated SMT pads for easy and reliable PCB mounting
- Certifications: FCC, ICS, CE
- Environmentally friendly, RoHS compliant

Applications

- Cable replacement
- Barcode scanners
- Measurement and monitoring systems
- Industrial sensors and controls
- Medical devices
- Barcode readers
- Computer accessories

Description

The RN42 is a small form factor, low power, highly economic Bluetooth radio for OEM's adding wireless capability to their products. The RN42 supports multiple interface protocols, is simple to design in and fully certified, making it a complete embedded Bluetooth solution. The RN 42 is functionally compatible with RN 41. With its high performance on chip antenna and support for Bluetooth® Enhanced Data Rate (EDR), the RN42 delivers up to 3 Mbps data rate for distances to 20M. The RN-42 also comes in a package with no antenna (RN-42-N). Useful when the application requires an external antenna, the RN-42-N is shorter in length and has RF pads to route the antenna signal.





Overview

- Baud rate speeds: 1200bps up to 921Kbps, non-standard baud rates can be programmed.
- Class 2 radio, 60 feet (20meters) distance, 4dBm output transmitter, -80dBm typical receive sensitivity
- Frequency 2402 ~ 2480MHz,
- FHSS/GFSK modulation, 79 channels at 1MHz intervals
- Secure communications, 128 bit encryption
- Error correction for guaranteed packet delivery
- UART local and over-the-air RF configuration
- Auto-discovery/pairing requires no software configuration (instant cable replacement).
- Auto-connect master, IO pin (DTR) and character based trigger modes

Digital I/O Characteristics

2.7V ≤ VDD ≤ 3.0V	Min	Typ.	Max.	Unit
Input logic level LOW	-0.4	-	+0.8	V
Input logic level HIGH	0.7VDD	-	VDD+0.4	V
Output logic level LOW	-	-	0.2	V
Output logic level HIGH	VDD-0.2	-	-	V
All I/O's (except reset) default to weakpull down	+0.2	+1.0	+5.0	uA

Environmental Conditions

Parameter	Value
Temperature Range (Operating)	-40 °C ~ 85 °C
Temperature Range (Storage)	-40 °C ~ 85 °C
Relative Humidity (Operating)	≤90%
Relative Humidity (Storage)	≤90%

Electrical Characteristics

Parameter	Min	Typ.	Max.	Unit
Supply Voltage (DC)	3.0	3.3	3.6	V
Average power consumption				
Radio ON* (Discovery or Inquiry window time)		40		mA
Connected Idle (No Sniff)		25		mA
Connected Idle (Sniff 100 milli secs)		12		mA
Connected with data transfer	40	45	50	mA
Deep Sleep Idle mode		26		uA

* If in SLAVE mode there are bursts of radio ON time which vary with the windows. Depending on how you set the windows that determines your average current.



Radio Characteristics

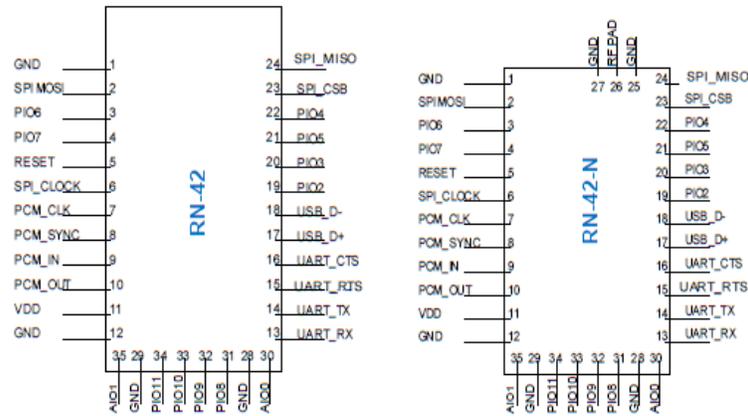
Parameter	Freq. (GHz)	Min	Typ	Max	Bluetooth Specification	Units
Sensitivity @ 0.1%BER	2.402	-	-80	-86	≤ -70	dBm
	2.441	-	-80	-86		dBm
	2.480	-	-80	-86		dBm
RF Transmit Power	2.402	0	2	4	≤ 4	dBm
	2.441	0	2	4		dBm
	2.480	0	2	4		dBm
Initial Carrier Frequency Tolerance	2.402	-	5	75	75	kHz
	2.441	-	5	75		kHz
	2.480	-	5	75		kHz
20dB bandwidth for modulated carrier		-	900	1000	≤ 1000	kHz
Drift (Five slots packet)		-	15	-	40	kHz
Drift Rate		-	13	-	20	kHz
Δf _{1avg} Max Modulation	2.402	140	165	175	>140	kHz
	2.441	140	165	175		kHz
	2.480	140	165	175		kHz
Δf _{2avg} Min Modulation	2.402	140	190	-	115	kHz
	2.441	140	190	-		kHz
	2.480	140	190	-		kHz

Range Characteristics (Approximate range in office environment)

Range	RN-42
After One Wall	55 feet
After Two Walls	60 feet
After Three Walls	36 feet

The above readings are approximate and may vary depending upon the RF environment. Bluetooth hops in a pseudo-random fashion over the 79 frequencies in the ISM band to adapt to the interference. Data throughput and range vary depending on the RF interference environment.

Pin Description



Pin	Name	Description	Default	Voltage
1	GND			0V
2	SPI MOSI	Programming only	No Connect	3V
3	PIO6	Set BT master (HIGH=auto-master mode)	Input to RN42with weak pulldown	0V-3.3V
4	PIO7	Set Baud rate (HIGH = force 9600, LOW = 115K or firmware setting)	Input to RN42 with weak pulldown	0V-3.3V
5	RESET	Active LOW reset	Input to RN42 with 1K pulkup	
6	SPI_CLK	Programming only	No Connect	
7	PCM_CLK	PCM interface	No Connect	
8	PCM_SYNC	PCM interface	No Connect	
9	PCM_IN	PCM interface	No Connect	
10	PCM_OUT	PCM interface	No Connect	
11	VDD	3.3V regulated power input		
12	GND			
13	UART_RX	UART receive Input	Input to RN42	0V-3.3V
14	UART_TX	UART transmit output	High level output from RN42	0V-3.3V
15	UART_RTS	UART RTS, goes HIGH to disable host transmitter	Low level output from RN42	0V-3.3V
16	UART_CTS	UART CTS, if set HIGH, disables transmitter	Low level input to RN42	0V-3.3V
17	USB_D+	USB port	Pull up 1.5K when active	0V-3.3V
18	USB_D-	USB port		0V-3.3V
19	PIO2	Status, HIGH when connected, LOW otherwise	Output from RN42	0V-3.3V
20	PIO3	Auto discovery = HIGH	Input to RN42 with weak pulldown	0V-3.3V
21	PIO5	Status, toggles based on state, LOW on connect	Output from RN42	0V-3.3V
22	PIO4	Set factory defaults	Input to RN42 with weak pulldown	0V-3.3V
23	SPI_CSB	Programming only	No Connect	
24	SPI_MISO	Programming only	No Connect	
25	GND	GND for RN42-N		
26	RF Pad	RF Pad for RN42-N		
27	GND	GND for RN42-N		
30	AIO0	Optional analog input	Not Used	
31	PIO8	Status (RF data rx/bx)	Output from RN42	0V-3.3V
32	PIO9	IO	Input to RN42 with weak pulldown	0V-3.3V
33	PIO10	IO (remote DTR signal)	Input to RN42 with weak pulldown	0V-3.3V
34	PIO11	IO (remote RTS signal)	Input to RN42 with weak pulldown	0V-3.3V
35	AIO1	Optional analog input	Not Used	



Data Sheet de la pantalla CI CD Toshiba



Distar Technology Limited

[Http://www.distarltd.com](http://www.distarltd.com)

Tel:86-755-26645997

Fax:86-755-26645987

SPECIFICATIONS FOR LCD MODULE

DS-G160128STBWW

PIXELS: 160 X 128 DOTS

OUTLINE DIMENSION: 129.0 X 102.0 MM

VIEWING AREA: 101.0 X 82.0 MM

DOT SIZE: 0.54 X 0.54 MM

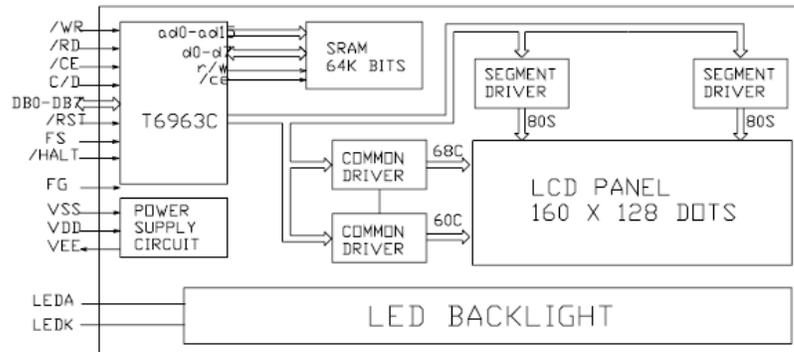
DOT PITCH: 0.58 X 0.58 MM

REVISION RECORD

REV.	DATE	PAGE	COMMENT
A	2007-3-15		NEW RELEASE

3. CIRCUIT BLOCK DIAGRAM

3.1 Electrical Block Diagram



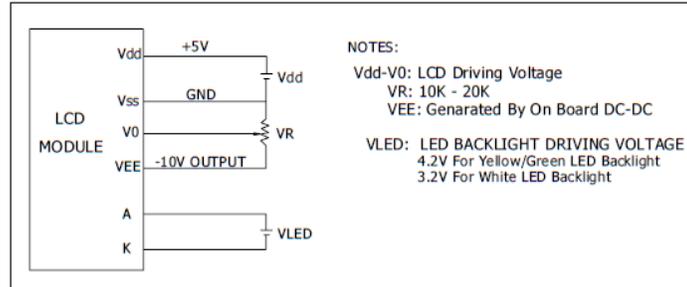
3.2 Pins Definition

PIN	SYMBOL	FUNCTION
1	FG	FRAME GROUND
2	Vss	Power Supply(GND)
3	Vdd	Power Supply For Logic(+5V)
4	Vo	Power Supply For LCD Driving (Contrast Adjust)
5	VEE	NEGATIVE VOLTAGE INPUT/OUTPUT
6	/WR	DATA WRITE
7	/RD	DATA READI
8	/CE	CHIP ENABLE FOR T6963C
9	C/D	COMMAND/DATA SELECTION
10	/HALT	CLOCK OPERATING STOP SIGNAL
11	/RST	RESET T6963C(LOW EFFECTIVE)
12--19	DB0—DB7	DATA BUS
20	NC	NO CONNECTION
21	LEDA	LED BACKLIGHT POWER SUPPLY(+)(+5V)

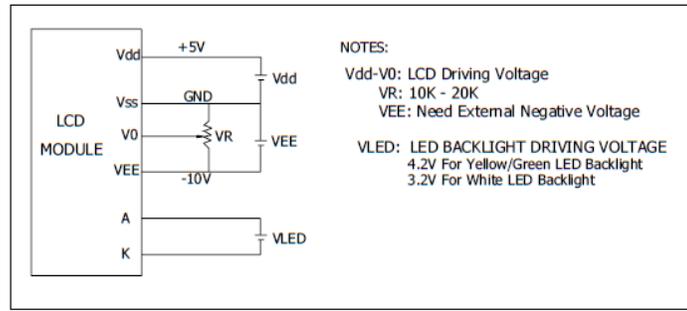
22	LEDK	LED BACKLIGHT POWER SUPPLY(-)(0V)
----	------	-----------------------------------

3.3 Power Supply For LCM Driving

3.3.1 For LCM With DC/DC on Board(Internal Negative Voltage)



3.3.2 For LCM without DC/DC on Board(Negative Voltage input)



4. ABSOLUTE MAXIMUM RATINGS

4.1 Electrical Absolute Maximum Ratings

ITEM	SYMBOL	CONDITION	MIN	MAX	UNIT
Supply Voltage (Logic)	Vdd - Vss	-	-0.3	7.0	V
Supply Voltage (LCD Drive)	Vdd - V0	-	0	25.0	V



Input Voltage	Vi	-	-0.3	Vdd +0.3	V
---------------	----	---	------	----------	---

4.2 Enviromental Absolute Maximum Ratings

ITEM	SYMBOL	CONDITIONS	MIN	MAX	UNIT
Operating Temp	Topr	-Normal temp.	-20	70	deg C
Storage Temp	Ttsg	version-	-30	80	deg C
Humidity Endurance	RH	no ondensation Ta<=40 deg	-	95	%
Vibration	-	100-300Hz, X/Y/Z directions, 1 hour	-	4.9m/ss 0.5g	-
Shock	-	10 mS X/Y/Z direction 1 time each	-	29.4m/ss 3.0g	-

5. ELECTRICAL CHARACTERISTICS

5.1 DC Characteristics

Electrical Characteristics at Ta=25 deg C, Vdd = 5V + / - 5%

ITEM	SYMBOL	CONDITION	MIN	TYP	MAX	UNIT
Supply Voltage (logic)	Vdd-Vss	-	4.5	5.0	5.5	V
Supply Voltage (LCD)	Vdd-V0	Vdd = 5V	-	18	-	V
Input signalVoltage (for CD, DB0-7,/WR,/R/ CS)	V-ih	"H" level	2.2	-	Vdd	V
	V-il	"L" level	0	-	0.6	V
Supply Current (logic)	Icc	-	-	1	1.2	mA
Supply Current	Io	-	0.15	0.22	0.27	mA



ANEXO NRO. 3.
Imágenes del prototipo y sus partes













APÉNDICES



APÉNDICE A: CÓDIGO FUENTE

Elaborado en Mikrobasic:

```
program MyProject

include __Lib_T6963C_Consts

dim T6963C_dataPort as byte at PORTB           ' DATA port
dim T6963C_ctrlwr as sbit at RC2_bit          ' WR write signal
dim T6963C_ctrlrd as sbit at RC1_bit          ' RD read signal
dim T6963C_ctrlcd as sbit at RC0_bit          ' CD command/data signal
dim T6963C_ctrlrst as sbit at RC4_bit         ' RST reset signal

dim T6963C_ctrlwr_Direction as sbit at TRISC2_bit      ' WR write signal
direction
dim T6963C_ctrlrd_Direction as sbit at TRISC1_bit      ' RD read signal
direction
dim T6963C_ctrlcd_Direction as sbit at TRISC0_bit      ' CD command/data
signal direction
dim T6963C_ctrlrst_Direction as sbit at TRISC4_bit     ' RST reset signal
direction

dim T6963C_ctrlce as sbit at RC3_bit                 ' CE signal
dim T6963C_ctrlfs as sbit at RA7_bit                 ' FS signal
dim T6963C_ctrlmd as sbit at RA6_bit                 ' MD signal
dim T6963C_ctrlce_Direction as sbit at TRISC3_bit     ' CE signal
direction
dim T6963C_ctrlfs_Direction as sbit at TRISA7_bit     ' FS signal
direction
dim T6963C_ctrlmd_Direction as sbit at TRISA6_bit     ' MD signal
direction
```



```
Dim L1, L2, L3, L4, L5, L6, L7, L8, L9, L10, L11, L12, L13, L14, L15,
U_Data_Str As String[20]
    S1, S2, S3, Time_Str, Speed_Text_Str As String[1]
    Pause As Word
    i as Byte
    Speed_Text, Res_Str As Integer

Sub Procedure Print_Delay(Dim byref Time_Str as String[1])
If Strcmp(Time_Str, "1") = 0 Then
    Delay_ms(1000)
    Goto End_Print_Delay
End If
If Strcmp(Time_Str, "3") = 0 Then
    Delay_ms(3000)
    Goto End_Print_Delay
End If
If Strcmp(Time_Str, "5") = 0 Then
    Delay_ms(5000)
    Goto End_Print_Delay
End If
If Strcmp(Time_Str, "7") = 0 Then
    Delay_ms(7000)
    Goto End_Print_Delay
End If
End_Print_Delay:
End Sub

Main:
    i=0
    UART1_Init(19200)           ' Initialize UART module at 19200 bps
    Delay_ms(100)              ' Wait for UART module to stabilize
    ADCON1 = 0x06
    TRISA = 0x00
    PORTA = 0x00
```



```
T6963C_ctrlce_Direction = 0
T6963C_ctrlce = 0
T6963C_ctrlfs_Direction = 0
T6963C_ctrlfs = 0          ' Font Select 8x8
T6963C_ctrlmd_Direction = 0
T6963C_ctrlmd = 0         ' Column number select

T6963C_init(160, 128, 8)
T6963C_text(1)
T6963C_cursor_height(8)   ' 8 pixel height
T6963C_set_cursor(0, 0)  ' Move cursor to top left
T6963C_cursor(0)         ' Cursor off
```

Main:

```
PORTA.0 = 1
```

```
If (UART1_Data_Ready() <> 0) Then
  UART1_Read_Text(Time_Str,"*",2)      ' reads text until '*' is found
  UART1_Read_Text(L1,"#",18)          ' reads text until '#' is found
  UART1_Read_Text(L2,"#",34)          ' reads text until '#' is found
  UART1_Read_Text(L3,"#",50)          ' reads text until '#' is found
  UART1_Read_Text(L4,"#",66)          ' reads text until '#' is found
  UART1_Read_Text(L5,"#",82)          ' reads text until '#' is found
  UART1_Read_Text(L6,"#",98)          ' reads text until '#' is found
  UART1_Read_Text(L7,"#",114)         ' reads text until '#' is found
  UART1_Read_Text(L8,"#",130)         ' reads text until '#' is found
  UART1_Read_Text(L9,"#",146)         ' reads text until '#' is found
  UART1_Read_Text(L10,"#",162)        ' reads text until '#' is found
  UART1_Read_Text(L11,"#",178)        ' reads text until '#' is found
  UART1_Read_Text(L12,"#",194)        ' reads text until '#' is found
  UART1_Read_Text(L13,"#",210)        ' reads text until '#' is found
  UART1_Read_Text(L14,"#",226)        ' reads text until '#' is found
  UART1_Read_Text(L15,"#",242)        ' reads text until '#' is found
```



Print_Str:

```
T6963C_write_text(L1, 0, 0, T6963C_ROM_MODE_XOR)
Print_Delay(Time_Str)
T6963C_write_text(L2, 0, 1, T6963C_ROM_MODE_XOR)
Print_Delay(Time_Str)
T6963C_write_text(L3, 0, 2, T6963C_ROM_MODE_XOR)
Print_Delay(Time_Str)
T6963C_write_text(L4, 0, 3, T6963C_ROM_MODE_XOR)
Print_Delay(Time_Str)
T6963C_write_text(L5, 0, 4, T6963C_ROM_MODE_XOR)
Print_Delay(Time_Str)
T6963C_write_text(L6, 0, 5, T6963C_ROM_MODE_XOR)
Print_Delay(Time_Str)
T6963C_write_text(L7, 0, 6, T6963C_ROM_MODE_XOR)
Print_Delay(Time_Str)
T6963C_write_text(L8, 0, 7, T6963C_ROM_MODE_XOR)
Print_Delay(Time_Str)
T6963C_write_text(L9, 0, 8, T6963C_ROM_MODE_XOR)
Print_Delay(Time_Str)
T6963C_write_text(L10, 0, 9, T6963C_ROM_MODE_XOR)
Print_Delay(Time_Str)
T6963C_write_text(L11, 0, 10, T6963C_ROM_MODE_XOR)
Print_Delay(Time_Str)
T6963C_write_text(L12, 0, 11, T6963C_ROM_MODE_XOR)
Print_Delay(Time_Str)
T6963C_write_text(L13, 0, 12, T6963C_ROM_MODE_XOR)
Print_Delay(Time_Str)
T6963C_write_text(L14, 0, 13, T6963C_ROM_MODE_XOR)
Print_Delay(Time_Str)
T6963C_write_text(L15, 0, 14, T6963C_ROM_MODE_XOR)
Print_Delay(Time_Str)
```



```
For i=0 to 14
    T6963C_write_text("                ", 0, i, T6963C_ROM_MODE_XOR)
Next i
End if

Goto Main

End.
```

Elaborado en Eclipse:

- Código en Java de la aplicación de Android:

```
package com.example.teleprompter;

import java.io.BufferedReader;
import java.io.File;
import java.io.FileInputStream;
import java.io.IOException;
import java.io.InputStreamReader;
import java.io.OutputStream;
import java.util.ArrayList;
import java.util.Set;
import java.util.UUID;
import java.util.Vector;

import android.annotation.SuppressLint;
import android.app.Activity;
import android.app.AlertDialog;
import android.bluetooth.BluetoothAdapter;
import android.bluetooth.BluetoothDevice;
import android.bluetooth.BluetoothSocket;
import android.content.BroadcastReceiver;
import android.content.Context;
import android.content.DialogInterface;
import android.content.Intent;
import android.content.IntentFilter;
import android.os.Bundle;
import android.os.Environment;
import android.util.Log;
import android.view.View;
import android.widget.ArrayAdapter;
import android.widget.Button;
```



```
import android.widget.EditText;
import android.widget.ListView;
import android.widget.RadioButton;
import android.widget.TextView;
import android.widget.Toast;

@SuppressLint("NewApi")
public class MainActivity extends Activity {

    public EditText msg;
    public RadioButton op1, op2, op3, op4;
    public Button archivo;
    public TextView status;
    public int estado1 = 0;
    ListView lstDispositivos;
    ListView lstDispositivos2;
    EditText seleccion1;
    ArrayList<?> dispositivos;

    /// para el bluetooth
    TextView out;
    // Debugging
    private static final UUID MY_UUID = UUID.fromString("00001101-0000-
1000-8000-00805F9B34FB");
    private BluetoothAdapter mBluetoothAdapter = null;
    private BluetoothSocket mmSocket=null;
    private static final String TAG = "BluetoothChat";
    private static final boolean D = true;
    private static final int REQUEST_ENABLE_BT = 3;
    ArrayAdapter<String> adaptador;
    final Vector<String> dispositivosBT = new Vector<String>();
    OutputStream ons;

    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        //String texto=null;
        setContentView(R.layout.activity_main);
        msg= (EditText) findViewById(R.id.msg);
        op1= (RadioButton) findViewById(R.id.op1);
        op2= (RadioButton) findViewById(R.id.op2);
        op3= (RadioButton) findViewById(R.id.op3);
        op4= (RadioButton) findViewById(R.id.op4);
        status = (TextView) findViewById(R.id.estado);

        if(D) Log.e(TAG, "+++ ON CREATE +++");

        mBluetoothAdapter = BluetoothAdapter.getDefaultAdapter();
        // If the adapter is null, then Bluetooth is not supported
```



```
        if (mBluetoothAdapter == null) {
            Toast.makeText(this, "Bluetooth no Soportado",
                Toast.LENGTH_LONG).show();
            finish();
            return;
        }
        else
        {
            if (!mBluetoothAdapter.isEnabled()) {
                Intent enableIntent = new
                Intent(BluetoothAdapter.ACTION_REQUEST_ENABLE);
                startActivityForResult(enableIntent, REQUEST_ENABLE_BT);
                Toast.makeText(this, "conectando...",
                    Toast.LENGTH_LONG).show();
            }else{
                Toast.makeText(this, "conectado",
                    Toast.LENGTH_LONG).show();
            }
        }
    }

    public void select1(View arg0){
        op3.setChecked(false);
        op4.setChecked(false);
    }
    public void select2(View arg0){
        op2.setChecked(false);
        op1.setChecked(false);
    }
    public void Archivo(View arg0)
    {
        Boolean ubi=false;
        try
        {
            BufferedReader fin =
                new BufferedReader(
                    new InputStreamReader(
                        openFileInput("/prueba_texto.txt")));
            String linea="", texto="";
            while((linea = fin.readLine()) != null){
                texto= texto + "" +(linea) + "\n";
            }
            fin.close();

            msg.setText(texto);
        }
        catch (Exception ex){
            ubi= true;
        }
    }
}
```



```
        if (ubi!=false){
            try
            {
                File ruta_sd =
Environment.getExternalStorageDirectory();

                File f = new File(ruta_sd.getAbsolutePath(),
"prueba_texto.txt");

                BufferedReader fin =
                    new BufferedReader(
                        new InputStreamReader(
                            new
FileInputStream(f)));

                String linea="", texto="";
                while((linea = fin.readLine()) != null){
                    texto= texto + " " +(linea) + "\n";
                }
                fin.close();

                msg.setText(texto);
            }
            catch (Exception exce)
            {
                Log.e("Ficheros", "Error al leer fichero");
            }
        }
    }
}
```

```
public void Enviar(View arg0)
{
    //Separa texto en 16 caracteres
    String texto = msg.getText().toString();
    String linea;
    String formateado;
    String opt;
    Integer n;
    opt= "";
    if (op1.isChecked()){
        opt= op1.getText().toString();
    }else if(op2.isChecked()){
        opt= op2.getText().toString();
    }else if(op3.isChecked()){
        opt= op3.getText().toString();
    }else if(op4.isChecked()){
```



```
        opt= op4.getText().toString();
    }else{
        opt= op2.getText().toString();
    }
    opt= opt.substring(0, opt.indexOf(" "));
    formateado= "";
    formateado= opt.concat("*");
    for(Integer i=0; texto != ""; i++){
        n=texto.length();
        if (n<16){
            if (n==0){
                texto="";
            }else{
                formateado= formateado.concat(texto).concat("#");
                texto="";
            }
        }else{
            linea = texto.substring(0, 16);
            texto= texto.substring(16);
            formateado= formateado.concat(linea).concat("#");
        }
    }
    msg.setText(formateado);
    try {
        OutputStream ons = mmSocket.getOutputStream();
        ons.write(formateado.getBytes());
    } catch (IOException e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
        Toast.makeText(getApplicationContext(), "No se Realizo
el Envio", Toast.LENGTH_SHORT).show();
    }
}

public void pCerrar(View view){
    Toast.makeText(getApplicationContext(), "Cerrando....",
Toast.LENGTH_SHORT).show();
    try {
        if (estadol==1)
            mmSocket.close();
    } catch (IOException e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
        Toast.makeText(getApplicationContext(), "Error
Cerrando Comunicación", Toast.LENGTH_SHORT).show();
    }
    this.finish();
}
```



```
public void buscar(View view){
    if (estado1==0)
    {
        sincro();
        adaptador = new
ArrayAdapter<String>(this,android.R.layout.simple_list_item_1,
dispositivosBT);
        new AlertDialog.Builder(this)
        .setTitle("Conectar a...!")
        .setAdapter(adaptador,new
DialogInterface.OnClickListener() {
            @Override
            public void onClick(DialogInterface dialog,
int which) {
                // TODO Auto-generated method stub
                String text
=dispositivosBT.elementAt(which).toString();
                text=(text.substring(text.length()-
17,text.length()));
                //String address = "00:06:66:02:0F:FB";
                BluetoothDevice device =
mBluetoothAdapter.getRemoteDevice(text);
                Toast.makeText(getApplicationContext(),
"CONNECTING", Toast.LENGTH_SHORT).show();
                try {
                    mmSocket =
device.createRfcommSocketToServiceRecord(MY_UUID);
                    mmSocket.connect();
                    status.setText("Conectado");
                    estado1=1;

                    Toast.makeText(getApplicationContext(), "Conectado con " +
device.getName(), Toast.LENGTH_SHORT).show();
                } catch (IOException e) {
                    // TODO Auto-generated catch block
                    e.printStackTrace();

                    Toast.makeText(getApplicationContext(), "No se Realizo la
Conexión\n"+"con "+text, Toast.LENGTH_SHORT).show();
                }
            }
        })
        .show();
    }
}

public void busqueda()
{
    if (mBluetoothAdapter.isDiscovering()) {
        Toast.makeText(this, "\nCancel discovery...",
Toast.LENGTH_LONG).show();
    }
}
```



```
        mBluetoothAdapter.cancelDiscovery();
    }else{
        Toast.makeText(this, "\nStarting discovery...",
Toast.LENGTH_LONG).show();
        //mBluetoothAdapter.
        mBluetoothAdapter.startDiscovery();
        Toast.makeText(this, "\nDone with discovery...",
Toast.LENGTH_LONG).show();
        encontrar();
    }
}
public void encontrar()
{
    /// buscar dispositivos
    //TextView dispositivos = (TextView) findViewById(R.id.dispositivos);

//final TextView dispositivos1 = (TextView) findViewById(R.id.textView2);
//seleccion1 = (EditText) findViewById(R.id.msg);
//lstDispositivos = (ListView) findViewById(R.id.dispositivos);
    final BroadcastReceiver mReceiver = new BroadcastReceiver() {
        public void onReceive(Context context, Intent intent) {
            //String dispositivosBT[] = new String[]{};
            String action = intent.getAction();
            // When discovery finds a device
            if (BluetoothDevice.ACTION_FOUND.equals(action)) {

//setContentview(R.layout.dispositivos_bluetooth);
//determinamos que al iniciar la aplicacion aparezca esta pantalla
// Get the BluetoothDevice object from the Intent
                BluetoothDevice device =
intent.getParcelableExtra(BluetoothDevice.EXTRA_DEVICE);
                // Add the name and address to an array adapter to show in a ListView
                //dispositivos1.append("\n" + device.getName() + "\n" +
device.getAddress());
                    dispositivosBT.add("\n" + device.getName() +
"\n" + device.getAddress());
            }
        }
    };

    // Register the BroadcastReceiver
    IntentFilter filter = new
IntentFilter(BluetoothDevice.ACTION_FOUND);
        registerReceiver(mReceiver, filter); // Don't forget to
unregister during onDestroy
    //// fin buscar dispositivo
}

public void sincro()
```



```
{
    dispositivosBT.clear();
    Set<BluetoothDevice> pairedDevices =
mBluetoothAdapter.getBondedDevices();
    // If there are paired devices
    if (pairedDevices.size() > 0) {
        // Loop through paired devices
        for (BluetoothDevice device : pairedDevices) {
// Add the name and address to an array adapter to show in a ListView
            dispositivosBT.add("\n" + device.getName() + "\n" +
device.getAddress());
        }
    }
}
```

- } Código de la Interfaz Gráfica de la aplicación de Android:

```
<TableLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent" >

    <TableRow
        android:id="@+id/tableRow1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="15dp"
        android:gravity="center_horizontal" >

        <TextView
            android:id="@+id/textView1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_gravity="center_horizontal"
            android:text="@string/Mensaje" />

    </TableRow>

    <TableRow
        android:id="@+id/tableRow2"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:gravity="center_horizontal" >

        <EditText
            android:id="@+id/msg"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content" />
```



```
        Android:layout_marginTop="20dp"
        Android:ems="10"
        Android:height="130dp"
        Android:inputType="textMultiLine"
        Android:width="300dp" >

        <requestFocus />
    </EditText>

</TableRow>

<RadioGroup
    Android:id="@+id/radioGroup1"
    Android:layout_width="wrap_content"
    Android:layout_height="wrap_content"
    Android:layout_marginTop="20dp"
    Android:gravity="center_horizontal"
    Android:orientation="horizontal">

    <RadioButton
        Android:id="@+id/op1"
        Android:layout_width="wrap_content"
        Android:layout_height="wrap_content"
        Android:checked="false"
        Android:onClick="select1"
        Android:paddingRight="25dp"
        Android:text="@string/opt1" />

        <RadioButton
            Android:id="@+id/op2"
            Android:layout_width="wrap_content"
            Android:layout_height="wrap_content"
            Android:onClick="select1"
            Android:text="@string/opt2" />

    </RadioGroup>

<RadioGroup
    Android:id="@+id/radioGroup2"
    Android:layout_width="wrap_content"
    Android:layout_height="wrap_content"
    Android:layout_marginTop="20dp"
    Android:gravity="center_horizontal"
    Android:orientation="horizontal" >

    <RadioButton
        Android:id="@+id/op3"
        Android:layout_width="wrap_content"
        Android:layout_height="wrap_content"
        Android:checked="false"
```



```
        Android:onClick="select2"
        Android:paddingRight="25dp"
        Android:text="@string/op3" />

    <RadioButton
        Android:id="@+id/op4"
        Android:layout_width="wrap_content"
        Android:layout_height="wrap_content"
        Android:onClick="select2"
        Android:text="@string/opt4" />

</RadioGroup>

<LinearLayout
    Android:layout_width="wrap_content"
    Android:layout_height="wrap_content"
    Android:layout_gravity="center_horizontal"
    Android:layout_marginTop="20dp"
    Android:gravity="center_horizontal" >

    <Button
        Android:id="@+id/enviar"
        Android:layout_width="wrap_content"
        Android:layout_height="wrap_content"
        Android:layout_gravity="center_horizontal"
        Android:layout_marginTop="20dp"
        Android:onClick="Enviar"
        Android:text="@string/enviar" />

    <Button
        Android:id="@+id/archivo"
        Android:layout_width="wrap_content"
        Android:layout_height="wrap_content"
        Android:layout_gravity="center_horizontal"
        Android:layout_marginTop="20dp"
        Android:onClick="Archivo"
        Android:text="@string/archivo" />

</LinearLayout>

</TableLayout>
```



APÉNDICE B: LISTA DE MATERIALES Y PRECIOS

MATERIALES	CANTIDAD	PRECIO POR UNIDAD	TOTAL (Bf)
PIC18F2525	1	185,00	185,00
Pantalla gráfica LCD Toshiba 6x5 pulgadas	1	1000,00	1000,00
Módulo Bluetooth RN-42	1	1050,00	1050,00
Base 28 Dip .3 INCH	1	3,00	3,00
Conector Snap de 9V-2	2	8,00	16,00
Acido Percloruro Férrico para PCB 100 Grs	1	35,00	35,00
Baquelita PCB 1 cara	1	16,00	16,00
Pila Energizer 9v	2	35,00	70,00
Cristal de 20 MHz	1	10,00	10,00
Mini switch bate SPDT 125v/3A 3 pin	1	9,00	9,00
Conector TDH 16l 8 pines por lado tipo IDE PCB hembra 90º	1	6,00	6,00
Carcasa del teleprompter	1	200,00	200,00
Aros portaleds 5mm	1	5,00	5,00
Conector pin 90 grado	1	5,00	5,00
LED verde	1	5,00	5,00
Regulador de voltaje NTE1904 3,3v	1	85,00	85,00
Regulador de voltaje L7805CV 5v	1	8,00	8,00
Potenciómetro 850K	1	8,00	8,00
			2716,00