

# **UNIVERSIDAD TECNOLÓGICA ISRAEL**

## **FACULTAD DE ELECTRONICA Y TELECOMUNICACIONES**

**Estudio, diseño e implementación de un tablero electrónico sonoro, que a través del envío de un mensaje de texto, el tablero reciba, visualice y reproduzca un mensaje de una lista previamente grabada, permitiendo enviar comunicados tanto sonoros como visuales a los alumnos de la Universidad Israel.**

**Estudiante**

**Iván Alejandro Gallardo  
Pacheco**

**Tutor**

**Ing. Mauro Bolagay**

**Quito Ecuador.**

**Octubre del 2013.**

# UNIVERSIDAD TECNOLÓGICA ISRAEL

## FACULTAD DE ELECTRONICA Y TELECOMUNICACIONES

### CERTIFICADO DE RESPONSABILIDAD

Yo, Mauro Bolagay, certifico que el Señor Iván Alejandro Gallardo Pacheco con C.C N° 050335253-6 realizo la presente tesis con título **“Estudio, diseño e implementación de un tablero electrónico sonoro, que a través del envío de un mensaje de texto, el tablero reciba, visualice y reproduzca un mensaje de una lista previamente grabada, permitiendo enviar comunicados tanto sonoros como visuales a los alumnos de la Universidad Israel”**, y es autor intelectual del mismo, que es original, autentica y personal

---

Firma del Tutor

ING. MAURO BOLAGAY.

# UNIVERSIDAD TECNOLÓGICA ISRAEL

## FACULTAD DE ELECTRONICA Y TELECOMUNICACIONES

### CERTIFICADO DE AUTORIA

El documento de tesis con título **“Estudio, diseño e implementación de un tablero electrónico sonoro, que a través del envío de un mensaje de texto, el tablero reciba, visualice y reproduzca un mensaje de una lista previamente grabada, permitiendo enviar comunicados tanto sonoros como visuales a los alumnos de la Universidad Israel”**, ha sido desarrollado por Iván Alejandro Gallardo Pacheco con C.C. N° 050335253-6 persona que posee los derechos de autoría y responsabilidad, restringiéndose la copia o la utilización de cada uno de los productos de esta tesis sin previa autorización.

---

Firma del Estudiante

IVÁN GALLARDO.

## **AGRADECIMIENTOS**

A mis padres, por brindarme su apoyo incondicional durante cada una de las etapas de mi vida, por haberme enseñado a seguir adelante sin importar las dificultades que en la vida se presenten.

A mi esposa y mi hijo, por motivarme a seguir adelante, y superarme por el bien de nuestra familia.

A mi Tutor de tesis ING. Mauro Bolagay, quien supo guiarme en el desarrollo de mi proyecto.

Gracias a todas las personas que de alguna manera, hicieron que sea posible el desarrollo de este proyecto.

## DEDICATORIA

A Dios, por permitirme estar aquí luchando por conseguir mis sueños, por darme la oportunidad de superarme cada día y ser mejor persona.

A mis padres que son mi ejemplo a seguir, que me enseñaron todo lo necesario para poder ser feliz, que dedicaron cada instante de su vida, para ver a sus hijos realizados,

A mi hermano que aunque a la distancia siempre me brindo su apoyo, y me dio la fuerza para conseguir mi título.

A mi esposa y mi hijo les dedico este trabajo por ser mis compañeros en esta vida, con su amor y su cariño, me han llenado de fuerzas para continuar, seguir adelante y buscar nuestra felicidad.

## **PROLOGO**

El trabajo de investigación presentado a continuación, es un aporte a la comunidad estudiantil, trabajo con el cual se pretende que el estudiante sea capaz de extraer información de documentos técnicos y mediante los conocimientos adquiridos en la universidad, pueda elaborar prototipos ó productos terminados.

Se pretende incentivar la elaboración de proyectos; tomando como base los conocimientos adquiridos en la universidad; que ayuden a la conservación del medio ambiente. Este proyecto será la base para futuros proyectos.

La idea de este proyecto nace de la necesidad encontrada en el diario vivir de un estudiante, buscando la manera de mantener informado y al tanto de actividades importantes y de interés, Proyecto que fue elaborado para satisfacer esta necesidad.

## RESUMEN

El objetivo del presente proyecto fue estudiar, diseñar e implementar un tablero electrónico sonoro que permitan informar de eventos importantes a realizarse en la universidad tales como: fechas de matriculas, inicio de clases, pago de pensiones. En vista de los problemas que se generan al no tener una forma directa de informar eventos y actividades que se ejecutan en la universidad; que tienen que ser consultadas directamente en las oficinas de colecturía o secretaria de la universidad, significando una pérdida de tiempo.

Para la ejecución del proyecto se utilizó el método científico, puesto que se planteo variables a ser analizadas, pero primero se recurrió a la investigación documental, para todo esto se empleó los métodos: observación, deductivo-inductivo y analítico-sintético.

Procesados que fueron los datos recopilados se determinó la presencia de el problema al mismo que se lo dio una respuesta oportuna y adecuada con la implementación de un dispositivo viso-auditivo que solucionara la poca o mala información existente en la universidad para docentes, estudiantes y público en general; esperando que esta herramienta sea aprovechada y utilizada de la manera más adecuada, convirtiéndose en un modelo a seguir por otras instituciones fiscales y particulares.

## ABSTRACT

The objective of this project was to study, design and implement an electronic sound board, that could inform important event to be held in the university such as dates of registrations, classes start, pension payments. In view of the problems caused by not having a direct way to report events and activities that run in college, they have to be consulted directly on the Collections office or secretary of the university, meaning a waste of time.

For the project we used the scientific method, we analyzed variables, but first consider the documentary research, for this, we used methods like a: observation, inductive and deductive-analytic-synthetic.

After processing the collected data, it indicated the presence of the problem and at the same time we response opportunately with the implementation of a visual-auditory device that solved the little or poor information in the university for teachers, students and public in general, I hoppe that this tool could be used in the most appropriate way, becoming a model for other fiscal institutions and individuals institutions.



## INDICE

### Contenido

<b>CAPITULO I</b> .....	<b>1</b>
<b>1.1 Antecedentes</b> .....	1
<b>1.2. Formulación del problema</b> .....	3
<b>1.3. Sistematización</b> .....	3
<b>1.3.1. Diagnóstico.</b> .....	3
<b>1.3.2. Pronóstico</b> .....	4
<b>1.3.2.1 Problemática Principal</b> .....	4
<b>1.3.2.2 Problemas secundarios</b> .....	4
<b>1.3.3. Control del Pronóstico.</b> .....	5
<b>1.4. Objetivos</b> .....	6
1.4.1. Objetivo general .....	6
1.4.2. Objetivos específicos.....	6
1.5.1. Justificación Teórica.....	7
1.5.2. Justificación Práctica .....	7
1.5.3. Justificación Metodológica .....	8
<b>1.6. Alcance y Limitaciones</b> .....	9
1.6.1. Alcance .....	9
<b>1.6.2. Limitaciones</b> .....	9
<b>1.7. Estudios de factibilidad</b> .....	10
<b>1.7.1. Técnica</b> .....	10
<b>1.7.2. Operativa</b> .....	11
<b>1.7.3. Económica.</b> .....	12
<b>1.7.3.1 Análisis Costos-Beneficios</b> .....	12
<b>1.7.3.2 Costos de Hardware y Software</b> .....	12
<b>1.7.3.3 Costo de Personal.</b> .....	13
<b>1.7.3.4 Beneficios Tangibles</b> .....	14
<b>1.7.3.5 Beneficios Intangibles.</b> .....	15
<b>CAPÍTULO II</b> .....	<b>16</b>
<b>MARCO TEÓRICO</b> .....	16

2.1.1. Telefonía celular .....	16
2.1.1.1. Introducción.....	16
2.1.1.2 Breve historia de la telefonía celular .....	16
2.1.1.3 GENERACIONES DE LA TELEFONÍA INALÁMBRICA .....	17
2.1.1.3.1 PRIMERA GENERACIÓN 1G.....	17
2.1.1.3.2 SEGUNDA GENERACIÓN 2G .....	18
2.1.1.3.3 La Generación 2.5g .....	18
2.1.1.3.4 La tercera generación 3G .....	19
2.1.1.3.5 La cuarta generación 4G .....	19
2.1.1.4 REDES DE TELEFONÍA CELULAR .....	19
2.1.1.5 FUNCIONAMIENTO DE LA TELEFONÍA CELULAR.....	20
2.1.1.6 SERVICIO DE MENSAJES CORTOS (SMS).....	24
2.1.1.6.1 DEFINICIÓN .....	24
2.1.1.6.2 BENEFICIOS DEL SERVICIO DE MENSAJES CORTOS (SMS) .....	25
2.1.1.6.3 APLICACIONES PARA SMS .....	26
2.1.1.7 JAVA .....	27
2.1.1.8 COMANDOS AT.....	28
2.1.1.8.1 CONTROL DE LLAMADA .....	29
2.1.1.8.2 COMANDOS SMS .....	29
2.1.1.8.2.1 Comandos generales.....	29
<b>2.1.1.8.2.2 Comandos del servicio de red .....</b>	<b>30</b>
2.1.1.8.2.3 Comandos de seguridad: .....	30
2.1.1.8.2.4 Comandos para la agenda de teléfonos .....	30
<b>2.1.1.8.2.5 Comandos para SMS .....</b>	<b>30</b>
2.1.2 MICROCONTROLADORES .....	31
2.1.2.1 ¿QUÉ ES UN MICROCONTROLADOR? .....	31
2.1.2.2 ESTRUCTURA BÁSICA DE UN MICROCONTROLADOR.....	31
2.1.2.3 ARQUITECTURA BÁSICA DE LOS MICROCONTROLADORES.....	32
2.1.2.3.1 ARQUITECTURA VON NEUMANN.....	32
2.1.2.3.2 LA ARQUITECTURA HARVARD .....	33
2.1.2.4 COMPONENTES DE UN MICROCONTROLADOR .....	34
2.1.2.4.1 PROCESADOR.....	34

2.1.2.4.2 CISC.....	34
2.1.2.4.3 RISC.....	35
2.1.2.4.4 SISC.....	35
2.1.2.4.5 MEMORIA.....	35
2.1.2.4.5.1 ROM CON MÁSCARA.....	35
2.1.2.4.5.2 OTP.....	36
2.1.2.4.5.3 EPROM.....	36
2.1.2.4.5.4 EEPROM, E2PROM o E <sup>2</sup> PROM.....	36
2.1.2.4.5.5 FLASH.....	37
2.1.2.4.6 PUERTAS DE E/S.....	38
2.1.2.4.7 RELOJ PRINCIPAL.....	38
2.1.2.4.8 RECURSOS ESPECIALES.....	38
2.1.2.4.9 TEMPORIZADORES O TIMERS.....	39
2.1.2.4.10 PERRO GUARDIÁN O WATCHDOG.....	39
2.1.2.4.11 PROTECCIÓN ANTE FALLO DE ALIMENTACIÓN O BROWNOUT ...	40
2.1.2.4.12 ESTADO DE REPOSO Ó DE BAJO CONSUMO.....	40
2.1.2.4.13 CONVERTOR A/D (CAD).....	41
2.1.2.4.14 CONVERTOR D/A (CDA).....	41
2.1.2.4.15 COMPARADOR ANALÓGICO.....	41
2.1.2.4.16 MODULADOR DE ANCHURA DE IMPULSOS O PWM.....	41
2.1.2.4.17 PUERTOS DIGITALES DE E/S.....	42
2.1.2.4.18 PUERTAS DE COMUNICACIÓN.....	42
2.1.2.5 FAMILIA DE LOS PIC.....	43
2.1.2.5.1 GAMA ENANA. PIC12CXXX.....	43
2.1.2.5.2 GAMA BAJA. PIC16C5XX CON INSTRUCCIONES DE 12 BITS.....	44
2.1.2.5.3 GAMA MEDIA. PIC16CXXX CON INSTRUCCIONES DE 14 BITS.....	44
2.1.2.5.4 GAMA ALTA: PIC17CXXX CON INSTRUCCIONES DE 16 BITS.....	45
2.1.3 SONIDO.....	46
2.1.3.1 Propagación del sonido.....	47
2.1.3.2 Magnitudes físicas del sonido del silencio.....	48
2.1.3.3 Fisiología del sonido.....	49
2.1.3.3.1 El aparato auditivo (Artículo principal: <i>Oído</i> ).....	49

2.1.3.3.2 La voz humana .....	49
2.1.3.4 Sonidos del habla.....	49
2.1.3.5 Señal de audio.....	50
2.2 Marco conceptual .....	51
2.2.1. MICROCONTROLADORES .....	51
2.2.2. Java .....	51
2.2.3. Modem Celular.....	51
2.2.4. Interfaz RS232 .....	51
2.3. Marco Legal.....	52
2.4. Marco Espacial.....	53
<b>CAPITULO III METODOLOGÍA .....</b>	<b>54</b>
3.1. Proceso de investigación.....	54
3.1.1. Unidad de Análisis .....	54
3.1.2. Tipo de Investigación.....	54
3.1.3. Método .....	54
3.1.3. Técnica.....	54
3.1.4. Instrumento.....	55
3.2. Metodología específica .....	55
<b>CAPÍTULO IV.....</b>	<b>57</b>
4.1. Estudio del tablero electrónico sonoro .....	57
4.1.1 Introducción .....	57
4.2 Descripción de funcionamiento.....	57
4.2.1 Encendido del tablero .....	57
4.2.2 Activación del tablero electrónico .....	58
4.2.3 Respuesta del Tablero electrónico sonoro después de validar la clave enviada.....	58
4.2.4 Selección de mensaje.....	59
4.2.5 Resultado final .....	59
4.3 Delimitación de la actividad del tablero electrónico sonoro .....	60
4.4 Descripción del sistema.....	60
<b>4.5 Avance tecnológico .....</b>	<b>60</b>
<b>4.6 DISEÑO DEL TABLERO ELECTRONICO SONORO .....</b>	<b>62</b>

4.6.1 Diagrama de bloques.....	62
4.6.2 SMS de activación.....	62
4.6.3 Fuente de poder .....	62
4.6.3.2 Fuente lineal.....	63
4.6.3.3 Diseño de la fuente poder.....	63
4.6.4 Fase de recepción de mensaje .....	64
4.6.4.1 Módulo GSM .....	64
Imagen de referencia .....	64
4.6.4.2 Capacitores electrolíticos.....	65
Descripción.....	65
Capacidades .....	65
Imagen de referencia .....	66
4.6.4.3 Capacitores cerámicos.....	66
Descripción.....	66
Imagen de referencia .....	66
4.6.4.4 Diodos LED .....	67
Descripción.....	67
Aplicaciones .....	67
Figura de referencia .....	67
4.6.4.5 Resistencias.....	68
Descripción.....	68
Código de colores.....	68
Imagen de referencia .....	69
4.6.4.6 Alimentación VAC .....	69
Descripción.....	69
Figura de referencia .....	70
4.6.5 Matriz de diodos led .....	70
4.6.6 Etapa de potencia (Amplificador de sonido).....	71
4.6.6.2 Altavoz .....	72
Descripción.....	72
Figura de referencia .....	72
4.6.7 Diseño de la estructura.....	73

4.6.7.1 Diseño de las estructuras laterales .....	73
4.7. IMPLEMENTACION DEL TABLERO ELECTRÓNICO SONORO .....	74
4.7.1 Introducción .....	74
4.7.2 Matriz de diodos .....	74
4.7.3 Ensamblaje en el protoboard de la matriz de diodos.....	74
4.7.4 Módulo GSM .....	78
4.8 PRUEBAS DEL TABLERO ELECTRÓNICO SONORO.....	82
4.8.1 Prueba de activación del tablero.....	82
<b>CAPÍTULO V .....</b>	<b>84</b>
ANALISIS DE COSTOS .....	84
5.1 Introducción .....	84
5.2 Matriz FODA.....	84
5.3 Análisis de costos .....	86
5.3.1 Materia Prima Directa (MPD).....	86
5.3.2 Costos indirectos de fabricación (CIF) .....	87
5.3.3 Mano de obra directa .....	88
5.3.4 Ingresos .....	92
5.3.5 Egresos .....	92
5.3.6 Precio de venta .....	93
5.4 Análisis del punto de equilibrio .....	93
5.4.1 Punto de equilibrio.....	93
5.5 Análisis costo – beneficio .....	95
5.6 Tasa mínima aceptable de retorno (TEMAR).....	96
5.7 Valor actual neto (VAN) .....	97
5.8 Tasa interna de retorno (TIR) .....	98
<b>CAPÍTULO VI.....</b>	<b>99</b>
<b>CONCLUSIONES Y RECOMENDACIONES .....</b>	<b>99</b>
<b>BIBLIOGRAFÍA.....</b>	<b>101</b>
<b>BIBLIOGRAFÍA INTERNET .....</b>	<b>102</b>
<b>ANEXO 1 .....</b>	<b>1028</b>
<b>ANEXO 2.....</b>	<b>102</b>
<b>ANEXO 3 (PROGRAMAS).....</b>	<b>10219</b>

## Índice de Tablas

Tabla 1: Costo del Proyecto .....	13
Tabla 2: (a) Comparación en uso de ordenadores y consumo de papel, (b) costo mensual del sistema en funcionamiento.....	14
Tabla 3: Nomenclatura del microcontrolador.....	46
Tabla 4. Análisis FODA.....	86
Tabla 5. Materia Prima Directa.....	87
Tabla 6. Suministros y materiales.....	88
Tabla 7. Servicios Básicos y Arriendo.....	88
Tabla 8. Costos indirectos de fabricación.....	88
Tabla 9. Mano de obra directa.....	89
Tabla 10. Costos de producción de una unidad.....	91
Tabla 11. Cuadro de ingresos.....	92
Tabla 12. Costo de producción proyectado egresos.....	92
Tabla 13. Precio de venta al público.....	93
Tabla 14. Clasificación de cuentas.....	93
Tabla 15. Punto de equilibrio.....	95
Tabla 16. Costo – Beneficio.....	96
Tabla 17. Tasa mínima aceptable de retorno.....	96
Tabla 18. Utilidad neta esperada.....	97
Tabla 19. Análisis VAN.....	97
Tabla 20. Análisis VAN.....	97
Tabla 21. Tasa interna de retorno.....	98

**INDICE DE FIGURAS**

Figura 1: Transmisión de radio onda corta .....	21
Figura 2: Transmisión de radio dual .....	21
Figura 3: Celdas divididas por área .....	22
Figura 4: Frecuencias reutilizadas en células.....	24
Figura 5: Esquema de un microcontrolador.....	33
Figura 6: Arquitectura Von Neumann .....	32
Figura 7: Arquitectura Harvard .....	33
Figura 8: Gama Enana Pic12F675 .....	44
Figura 9: Gama Media PIC16F84.....	45



## INDICE DE ILUSTRACIONES

Ilustración 1. Diseño de la fuente .....	63
Ilustración 2. Módulo GSM .....	64
Ilustración 3. Condensadores electrolíticos .....	66
Ilustración 4. Condensadores cerámicos.....	66
Ilustración 5. Diodos LED .....	67
Ilustración 6. Código de Colores.....	68
Ilustración 7. Resistencia encapsulada axial .....	69
Ilustración 8. Forma de onda de la corriente alterna .....	70
Ilustración 9. Matriz de 5x7 con columnas de cátodos .....	70
Ilustración 10. Matriz de led.....	71
Ilustración 11. Amplificador de sonido .....	72
Ilustración 12. Altavoz.....	72
Ilustración 13. Estructura principal.....	73
Ilustración 14. Estructuras laterales.....	73
Ilustración 15. Matrices de diodos led. ....	74
Ilustración 16. Ensamblado en protoboard .....	74
Ilustración 17. Diagrama esquemático de la placa de control de la matriz de leds .....	75
Ilustración 18. Captura de simulación de la placa de control de la matriz de leds .....	76
Ilustración 19. Placa de control de la matriz de leds .....	77
Ilustración 20. Diagrama esquemático del conjunto de matrices de led.....	77
Ilustración 21. Montaje de las matrices de diodos a la placa del tablero.....	78
Ilustración 16. Ensamblado en protoboard .....	74
Ilustración 19. Placa de control de la matriz de leds .....	77
Ilustración 21. Montaje de las matrices de diodos a la placa del tablero.....	78
Ilustración 22. Módulo GSM .....	78
Ilustración 23. Diagrama esquemático de la placa de transferencia del modem GSM a la matriz de led y a módulo mp3.....	79

Ilustración 24. Captura de la simulación de la placa de transferencia del modem GSM a la matriz de led y a módulo mp3.....	80
Ilustración 25. Placa de transferencia del modem GSM a matriz de led y a módulo mp3.....	81
Ilustración 26. Montaje de placas a la estructura .....	81
Ilustración 27. Vista frontal de la matriz led montada a la estructura principal.....	82
Ilustración 28. Envío clave de activación al tablero para la primera opción .....	83
Ilustración 29. Muestra mensaje en el tablero .....	83
Ilustración 30. Punto de equilibrio.....	95

**INDICE DE DIAGRAMAS**

Diagrama 1: Inicio de funcionamiento del tablero.....	57
Diagrama 2. Activación del tablero.....	58
Diagrama 3. Respuesta del tablero.....	58
Diagrama 4. Selección de mensaje.....	59
Diagrama 5. Muestra y visualiza el mensaje.....	59
Diagrama 6. Diagrama de bloques del sistema.....	62

# CAPITULO I

## GENERALIDADES

### 1.1 Antecedentes

La Universidad Israel nace a partir de la fusión de dos Institutos Superiores de la ciudad de Quito, el Instituto Superior Israel y el Instituto Superior Italia.

El Instituto Técnico Israel fue fundado el día 7 de agosto de 1984 mediante resolución ministerial N°. 1963 para formación de Analistas de Computación, impulsando la educación técnica en el país.

En la administración del Dr. Eduardo Peña Triviño, Ministro de Educación y Cultura el día 27 de agosto de 1993 el Instituto Técnico “Israel” mediante resolución ministerial N°. 4491 pasa a ser Instituto Tecnológico Particular “Israel” y se crea la Tecnología de Informática en la especialidad de Analista de Sistemas.

Con el afán de cooperar y poner a disposición la infraestructura y la experiencia pedagógica al servicio de la comunidad, el día 10 de mayo de 1993 se reúne la Asamblea General del Instituto “Israel” con el propósito de solicitar la creación de la “FUNDACIÓN EXPERIMENTAL EDUCATIVA PARA CAPACITAR A LOS DISCAPACITADOS EN ÁREAS TECNICAS” en sus siglas (FEECDAT), siendo aceptado mediante acuerdo Ministerial N°. 3334 del 1°. De julio de 1993.

El 11 de junio de 1996 en la administración del señor Dr. León Vieira Villafuerte Secretario de Educación, se crea la especialización de Electrónica con resolución oficial N°. 2586.

Mediante oficio N°. 20-9900245 de 19 de enero de 1999, el Instituto Tecnológico Particular "ISRAEL" solicita la autorización para la creación de las especialidades de Diseño Gráfico y Administración de Empresas con tres años de estudio y la especialización de Contador Público Auditor. En la Administración del señor Dr. Carlos Quiroz Palacios, Ministro de Educación y Cultura; con resolución oficial N°. 1033 del 4 de junio de 1999, autoriza la creación de estas especialidades.

Entre los frutos del Instituto es haber obtenido desde el año 1997, por tres ocasiones consecutivas el 1° lugar en el ramo de Escuelas de Computación otorgada por el Instituto de "Calidad y Prestigio" de la ciudad de Guadalajara México.

El Instituto Técnico Italia nace un 10 de julio del 1986 mediante resolución Ministerial No. 251 con las especialidades de Programadores Analistas de Sistemas con 2 años de duración.

El 21 de Octubre de 1996 con resolución Ministerial N°.562 el Instituto Técnico Italia pasa a ser Instituto Tecnológico Superior "Italia" con las especialidades de Auditoría Informática y Análisis de Sistemas.

El 15 de diciembre de 1998 se aprueban por resolución Ministerial N°. 2546 las especialidades de Mercadotecnia, Administración de Empresas y Administración Bancaria.

El Instituto Tecnológico "Israel" y el Instituto Tecnológico "Italia" con la finalidad de unificar esfuerzos, experiencias, conocimientos e infraestructuras físicas en beneficio del país crean la Universidad Tecnológica "Israel", reconocida mediante Ley de creación N°. 99-42, publicada en el Registro Oficial N°. 319 del 16 de noviembre de 1999.

El Lic. Darío Moreira Velásquez Director Ejecutivo del CONESUP el 24 de mayo de 2001 da a conocer al Consejo Nacional de Educación Superior el informe de la Comisión Administrativa del CONESUP, el mismo que resolvió aprobar el Estatuto de la Universidad.

## **1.2. Formulación del problema**

¿Con la implementación del tablero electrónico sonoro, se podrá disminuir la pérdida de tiempo al enviar comunicados al estudiante mejorando la comunicación?

## **1.3. Sistematización**

### **1.3.1. Diagnóstico.**

Uno de los problemas fundamentales tanto para la Universidad como para los estudiantes, son los procesos de comunicación que les mantengan informados de todos los acontecimientos curriculares y extracurriculares programados por la Institución educativa superior

A pesar de que la información aparezca publicada en carteleras a lo interno de la Universidad, ésta es susceptible de modificaciones por diferentes acontecimientos, por lo que se produce la desinformación, con las consecuentes pérdidas de tiempo para los estudiantes; y uno que otro docente.

De igual manera, se ha visto como una necesidad imperiosa que el estudiante cuente con una información exacta y oportuna, sobre inicio de pago de pensiones, seminarios u otros. De tal forma que pueda programar eficientemente su economía y la Universidad recabe eficientemente esos capitales.

### **1.3.2. Pronóstico.**

#### **1.3.2.1 Problemática Principal**

La Universidad Israel no cuenta con un tablero electrónico sonoro, que permitan informar de eventos importantes a realizarse en la universidad tales como: fechas de matriculas, pago de pensiones, actividades que tienen que ser consultadas directamente en las oficinas de colecturía o secretaria de la universidad, significando una pérdida de tiempo.

#### **1.3.2.2 Problemas secundarios**

- No existe un estudio acerca de un tablero electrónico sonoro que cubra la necesidad de la Universidad Israel para el anuncio de actividades sin acercarse a las oficinas respectivas.
- No existe un diseño de un tablero electrónico sonoro sencillo y manejable que permita informar las actividades realizadas en la U. Israel.

- No existe la implementación de un tablero electrónico sonoro adecuado para el estudiante que le permita estar al tanto de las actividades a realizarse en la U. Israel.
- No existe la validación del funcionamiento del tablero electrónico sonoro que cubra la necesidad de la Universidad Israel para la consulta de actividades, sin acercarse a las oficinas.

### **1.3.3. Control del Pronóstico.**

Para ir cumpliendo las propuestas planteadas en la presente investigación, se tomaran en cuenta los siguientes aspectos:

El estudio, diseño e implementación de un tablero electrónico sonoro que permitan informar de eventos importantes a realizarse en la universidad tales como: fechas de matriculas, inicio de clases, pago de pensiones.

Análisis de los contenidos teóricos, que orienten adecuadamente la investigación.

Diseño de un tablero electrónico sonoro que cubra con los requerimientos de la Universidad Israel para anuncio de actividades.

La implementación de un tablero electrónico sonoro que cumpla con las necesidades que requiere la Universidad Israel para tener una mejor comunicación y dinamizar procesos con respecto al desarrollo de actividades y cumplimiento de los pagos de las pensiones.

Y validación del tablero electrónico sonoro.



## **1.4. Objetivos**

La Universidad Israel no cuenta con un tablero electrónico sonoro, que permitan informar de eventos importantes a realizarse en la universidad tales como: fechas de matriculas, pago de pensiones, actividades que tienen que ser consultadas directamente en las oficinas de colectoría o secretaria de la universidad, significando una pérdida de tiempo.

### **1.4.1. Objetivo general**

Estudiar, diseñar e implementar un tablero electrónico sonoro que permitan informar de eventos importantes a realizarse en la universidad tales como: fechas de matriculas, inicio de clases, pago de pensiones.

### **1.4.2. Objetivos específicos**

- Analizar los contenidos teóricos, que orienten adecuadamente la investigación.
- Diseñar un circuito para un tablero electrónico sonoro que cubra con los requerimientos de la Universidad Israel para anuncio de actividades.
- Implementar un circuito para un tablero electrónico sonoro que cumpla con las necesidades que requiere la Universidad Israel para tener una mejor comunicación y dinamizar procesos con respecto al desarrollo de actividades y cumplimiento de los pagos de las pensiones.
- Validar el circuito para un tablero electrónico sonoro.

## **1.5. Justificación**

### **1.5.1. Justificación Teórica**

Para esta investigación se ha recurrido a abundante literatura que permita presentar adecuadamente el proyecto de Diseño y la construcción de un tablero electrónico sonoro que pretende incentivar la producción de tecnologías para a su vez mantener una comunicación entre el estudiante, el docente y la universidad usando microcontroladores; esta tecnología, en su parte teórica puede tener diferentes aplicaciones que servirán de base para futuros proyectos investigativos.

Debido al avance de la tecnología cada día es indispensable optimizar el tiempo, más aún cuando los usuarios, en este caso los estudiantes de la Universidad, realizan actividades de trabajo y estudio a la vez, lo cual dificulta el acceso a información indispensable. Por este motivo se ha pensado en el uso de las facilidades que ofrece actualmente la tecnología para lograr respuestas, las más rápidas posibles, además se debe mencionar que un servicio como el que se propone en el presente proyecto no existe a nivel de la Institución, ofreciendo al estudiante de esta forma un fácil acceso a su derecho de conocer información relevante, en cualquier momento y desde cualquier lugar en el que se encuentre.

### **1.5.2. Justificación Práctica**

La implementación del tablero electrónico sonoro permitirá al estudiante estar informado oportunamente de las actividades curriculares y extracurriculares a realizarse en la Universidad Israel, optimizando los cobros.

La utilidad de este proyecto radica en que al ser el celular una herramienta móvil va estar con la persona las 24 horas al día, lo que permite que la información sea oportuna y al instante consiguiendo un ahorro significativo en cuanto al tiempo y mejorando significativamente la comunicación

Es necesario señalar que este proyecto es de tipo ecológico debido a que el envío de mensajes con este sistema se lo realizaría a través de un dispositivo móvil como lo es un celular, eliminando en lo posible la utilización del papel, que está causando la deforestación en el planeta.

Con este proyecto la universidad se incluirá dentro del gran grupo de empresas e instituciones que usan esta dinámica y versátil herramienta como es el servicio de mensajes escritos.

### **1.5.3. Justificación Metodológica**

Para el presente proyecto se utilizará el método científico ya que las variables propuestas serán susceptibles de experimentación y comprobación, por tanto se procederá primeramente a la investigación de teorías y conocimientos existentes acerca de la temática, para luego pasar a la formulación de propuestas y realización esquemática, en la fase de diseño. Por último se realizará la comprobación de lo planteado en la fase de implementación y pruebas.

**Fase inicial:** En primera instancia se utilizará la investigación documental para recopilar información y fotos necesarias para el diseño del producto. Se utilizarán fichas bibliográficas.

**Diseño:** Para el diseño del producto se requerirá una investigación de tipo documental, utilizando el método de análisis y síntesis para determinar los resultados de los datos, dentro de las técnicas de procesamiento de datos se encuentran diferentes tipos de diagramas esquemáticos, físico, por bloques, etc.

**Implementación y pruebas:** En esta fase se utilizarán los métodos: observación, deductivo-inductivo y la experimentación, para determinar si el producto cumple con todas las características definidas en la fase del diseño.

## **1.6. Alcance y Limitaciones**

### **1.6.1. Alcance**

Obtener un estudio, diseño e implementación de un tablero electrónico sonoro que permitan informar de eventos importantes a realizarse en la universidad tales como: fechas de matriculas, inicio de clases, pago de pensiones. Para mantener informado al estudiante de las actividades realizadas por la Universidad de una manera rápida.

### **1.6.2. Limitaciones**

El sistema dependerá de una persona que se encargue de modificar la información como son fechas, mensajes y sonidos, también dependerá del saldo disponible para el envío de mensajes escritos, otra de las limitaciones será la disponibilidad de la red de telefonía celular.

## **1.7. Estudios de factibilidad**

### **1.7.1. Técnica**

La factibilidad técnica consistió en realizar una evaluación de la tecnología existente en la Universidad, este estudio estuvo destinado a recolectar información sobre los componentes técnicos que posee la universidad y en la posibilidad de utilizar los mismos en el desarrollo e implementación del circuito, y de ser necesarios, los requerimientos tecnológicos que deben ser obtenidos para el desarrollo y ejecución del circuito.

De acuerdo a la tecnología necesaria para la implementación del circuito electrónico para una plataforma de envío y recepción de mensajes escritos (SMS) , que mantenga al estudiante informado de las actividades a realizarse y además pueda consultar el saldo que adeuda a la Universidad Israel, se evaluó bajo dos enfoques: Software y hardware.

#### **Hardware.**

En cuanto hardware, específicamente al circuito:

- Microcontrolador
- Condensadores
- Resistencias
- Placas
- Matrices led
- Leds

- Relé
- Fuentes de alimentación
- Parlante y amplificador
- Chip GSM
- Memoria SD
- Modulo MP3
- Modulo para conexión GSM con pic
- Cables de alimentación eléctrica
- Interruptor
- Caja de madera

Respecto a estos materiales, se los puede encontrar fácilmente en el mercado.

### **1.7.2. Operativa**

El circuito propuesto se pondrá en marcha, aprovechando los beneficios que brinda, a los usuarios involucrados, ya sean los que interactúan en forma directa con este, como también aquellos que reciben información producida por el modulo del circuito. Por otra parte, el correcto funcionamiento del circuito en cuestión, siempre dependerá de la persona encargada de alimentar y actualizar los mensajes.

La necesidad y deseo de una alternativa de consulta y acceso a información importante, expresada por los estudiantes y el personal involucrado con el mismo, llevó a la aceptación de una nueva forma de informar a los estudiantes , que de una manera más sencilla y amigable, cumpla con todos sus requeri-

mientos, expectativas y proporciona la información en forma oportuna y confiable. Basándose en conversaciones sostenidas con el personal involucrado se demostró que estos no representan ninguna oposición al cambio, por lo que el circuito es factible operacionalmente.

### **1.7.3. Económica.**

A continuación se presenta un estudio que da como resultado la factibilidad económica del desarrollo del nuevo sistema de información. Se determinaron los recursos para desarrollar, implementar, y mantener en operación el sistema programado, haciendo una evaluación donde se puso de manifiesto el equilibrio existente entre los costos específicos del sistema y los beneficios que provinieron de éste. Un análisis económico mas detallado se va a realizar en el Capítulo V

#### **1.7.3.1 Análisis Costos-Beneficios**

Como se menciona anteriormente en el estudio de factibilidad técnica, la Universidad cuenta con una gran parte de las herramientas necesarias para la ejecución del sistema, por lo tanto el desarrollo del proyecto propuesto no requiere de una gran inversión inicial.

#### **1.7.3.2 Costos de Hardware y Software.**

Debido a que en la Universidad cuentan con los equipos y recursos técnicos necesarios, para el desarrollo del circuito, no se requiere una fuerte inversión en este aspecto. Esta situación facilitó la puesta en marcha del proyecto, ofre-

ciéndole a la Universidad la posibilidad de realizar inversiones en otros requerimientos para el proyecto.

La siguiente tabla muestra el costo aproximado de construcción del sistema:

Material	Costo
Microcontroladores	48.5
Modem GSM	90
Condensadores	1.23
Resistencias	0.12
Leds	0.4
Materiales varios	39.78
Mano de obra	300
<b>Total</b>	<b>480.03</b>

**Tabla 1: Costo del Proyecto**

**Realizado por:** Iván Gallardo

### 1.7.3.3 Costo de Personal.

El sistema propuesto no incluyó variaciones en cuanto a las personas bajo cuya responsabilidad está la operación y/o funcionamiento del circuito. El equipo de desarrollo no generó inversión, ya que por tratarse de un proyecto elaborado como trabajo de grado, las personas encargadas de impulsar el mismo, no implicó gasto alguno; aspecto que favoreció aún más el proyecto en cuestión, pero cabe destacar que al implementar el circuito conjuntamente con el software permitirá un ahorro de tiempo para el personal encargada de estas labores.



### 1.7.3.4 Beneficios Tangibles

Los beneficios tangibles aportados por el sistema propuesto vienen dados por los siguientes aspectos:

- Reducción de costos en papelerías y espacio físico.
- Ahorro en suministros para los equipos utilizados.

(a)

	Sistema actual	Costo	Sistema propuesto	Costo
Uso de papel en anuncios, memos, etc. ( por mes)	500 hojas por consulta	4 dólares	0 hojas	0 dólares
Numero de ordenadores (consumo energético por ordenador 3 Kwh, costo del Kwh 0.06\$)	3	10.8 dólares	0	0 dólares
	<b>Total</b>	<b>14.8 dólares</b>	<b>Total</b>	<b>0 dólares</b>

Kwh= consumo (Kw) x Hora(h)  
 0.20 Kw x 15h=3Kwh consumo de un ordenador por día  
 Al mes 60Kwh  
 60 Kwh x 0.06= 3.6 dólares por ordenador al mes

(b)

	Cantidad	Costo
Costo mensual por el servicio de mensajes escritos	20 x 0.67 \$	13.4 dólares
Numero de ordenadores (consumo energético por ordenador 3 Kwh, costo del Kwh 0.06\$)	0	0 dólares
	<b>Total</b>	<b>13.4 dólares</b>

**Tabla 2:** (a) Comparación en uso de ordenadores y consumo de papel, (b) costo mensual del sistema en funcionamiento

Realizado por: Iván Gallardo

### **1.7.3.5 Beneficios Intangibles.**

Entre los beneficios intangibles del sistema propuesto que se pueden incluir:

- Asistencia oportuna a eventos de gran importancia realizadas por la Universidad, que permite un mejor y más efectivo empleo de los recursos, tanto financieros como materiales.
- La facilidad al manejar gran volumen y diversidad de información con rapidez, oportunidad y precisión, lo que brinda una potencial herramienta de trabajo al personal, que facilitará sus labores.
- Generar información más eficiente y confiable.
- Mejor capacidad de búsqueda y actualización de información, reduciendo el esfuerzo en el proceso y el control de recursos.
- Formar parte de una Sociedad de la Información, a través de la cual se logra la comunicación directa con el estudiante.

## **CAPÍTULO II**

### **MARCO TEÓRICO**

#### **2.1.1. Telefonía celular**

##### **2.1.1.1. Introducción**

Las tecnologías inalámbricas están teniendo mucho auge y desarrollos en los últimos años, siendo la telefonía celular la de más desarrollo. Desde sus inicios en la época de los 70's ha revolucionado considerablemente las actividades que se realizan diariamente. Los celulares se han convertido en una herramienta fundamental tanto como para la gente común y como para la de negocio. Las hace mas productivas y las hace sentir más seguras.

A pesar que la telefonía celular fue concebida para la voz únicamente, debido a las limitaciones tecnológicas de esa época, le tecnología celular de hoy en día es capaz de ofrecer otro tipo de servicios tales como audio, video y datos con algunas restricciones, pero la tecnología inalámbrica en el futuro hará posible aplicaciones que demanden mayor ancho de banda para su funcionamiento.

##### **2.1.1.2 Breve historia de la telefonía celular**

Martin Cooper fue el pionero en la telefonía celular, al introducir el primer radioteléfono en el año de 1973 en Estados Unidos mientras se encontraba trabajando para Motorola por esta razón se le considera "el padre de la telefonía celular"; pero no fue hasta el año 1979 en que surge el primer sistema comercial en Japón por la compañía Nippon Telegraph & Telephone Corp (NTT).

En el año de 1981 en los países Nórdicos se introduce un sistema celular de

características similares a Advanced Mobile Phone System (AMPS). Mientras que, en Estados Unidos se acoge reglas para la creación de un servicio de telefonía celular comercial, en octubre de 1983, y se pone en marcha en la ciudad de Chicago el primer sistema comercial.

Desde aquel momento en varios países se esparció la telefonía celular como una opción a la telefonía convencional alámbrica.

La tecnología inalámbrica tuvo gran aceptación, por lo que al poco tiempo de establecerse se comenzó a saturar el servicio, lo que obligo a desarrollar e implementar formas alternas de múltiple acceso al canal y transformar de un sistemas analógicos a digitales para dar mayor capacidad de usuarios.

A la telefonía celular se ha categorizado por generaciones. Que son las que se describen a continuación.

### **2.1.1.3 GENERACIONES DE LA TELEFONÍA INALÁMBRICA**

#### **2.1.1.3.1 PRIMERA GENERACIÓN 1G**

La telefonía móvil 1G apareció en el año de 1979, se caracterizó por ser únicamente para voz y su señal analógica. La calidad de los enlaces de voz no eran muy buenos, con una velocidad de (2400 bauds), la transferencia entre celdas era ineficaz, poseían baja capacidad (basadas en FDMA, Frequency Divison Múltiple Access) y no existía seguridad. La tecnología sobresaliente de esta generación es la AMPS de sus siglas en ingles (Advanced Mobile Phone System).

### **2.1.1.3.2 SEGUNDA GENERACIÓN 2G**

La generación 2G llegó hasta el año de 1990 y a diferencia de la 1G se destacó por ser digital. Los protocolos de codificación que la 2G utiliza son más avanzados y son los sistemas de telefonía celular que se utilizan en la actualidad. Las tecnologías predominantes son: CDMA (Code División Multiple Access); PDC (Personal Digital Communications), GSM (Global System for Mobile Communications) y IS-136 (conocido también como TIA/EIA-136 o ANSI-136), PDC es utilizado en Japón.

En los sistemas 2G los protocolos utilizados soportan mayores velocidades de información para voz pero reducidos en comunicaciones de datos. Se pueden brindar servicios complementarios tales como SMS (Short Message Service), datos y fax.

La mayor parte de los protocolos de 2G brindan diferentes niveles de encriptación.

### **2.1.1.3.3 La Generación 2.5G**

Muchas de las empresas que prestan servicios de telecomunicaciones migraron a las redes 2.5G antes de entrar completamente a 3G. Debido a que es más rápido y económico actualizar de 2.5 G a 3G.

Las características que brinda la generación 2.5G son desarrolladas para proporcionar capacidades adicionales a los sistemas 2G, tales como High Speed Circuit Switched Data (HSCSD), IS-136B, IS-95B, Enhanced Data Rates for Global Evolution (EDGE), General Packet Radio System (GPRS), entre otros.

#### **2.1.1.3.4 La tercera generación 3G**

La 3G es representada por la convergencia de datos y voz con acceso a Internet inalámbricamente, altas transmisiones de datos y aplicaciones multimedia. Los sistemas 3G utiliza protocolos que soportan altas velocidades de información orientados para aplicaciones que van mas allá de la voz como son acceso rápido a Internet, audio (MP3), video en movimiento y video conferencia, entre otros. Se empezó a operar las redes 3G desde el año 2001 en Japón por NTT DoCoMo, parte de Asia y en Europa a partir del 2002, a continuación en Estados Unidos y otros países del mundo.

#### **2.1.1.3.5 La cuarta generación 4G**

La cuarta generación es un proyecto a largo plazo y será 50 veces más veloz que la tercera generación.

#### **2.1.1.4 REDES DE TELEFONÍA CELULAR**

Podemos distinguir dos tipos de redes de telefonía móvil, la existencia de estas redes es fundamental para el uso de nuestro teléfono celular, para navegar en internet o para enviar mensajes de texto.

La primera es la red de telefonía móvil de tipo analógica (TMA), la misma establece la comunicación mediante señales vocales analógicas, tanto en el tramo radioeléctrico como el tramo terrestre; la primera versión de la misma funcionó en la banda radioeléctrica de los 450 MHz, posteriormente trabajaría en la banda de 900 MHz.

Luego tenemos la red de telefonía móvil digital, en esta la comunicación se lleva a cabo mediante señales digitales, esto permite optimizar el uso de las

bandas de radiofrecuencia como también la calidad de la transmisión de las señales. El exponente más característico con la que cuenta esta red actualmente es el GSM y su tercera generación UMTS, tanto la GSM como la UMTS funcionan en las bandas de 850/900 MHz.

En América Latina existe otro estándar digital conocido como CDMA.

#### **2.1.1.5 FUNCIONAMIENTO DE LA TELEFONÍA CELULAR**

Un teléfono celular utiliza una frecuencia para transmitir la voz, y una segunda frecuencia independiente para recibir la voz, por lo que se le conoce como dispositivo dual. Un teléfono celular puede utilizar 1664 canales. Estos teléfonos también operan con "células" (o "celdas") y pueden alternar la celda usada a medida que el teléfono se encuentra en movimiento.

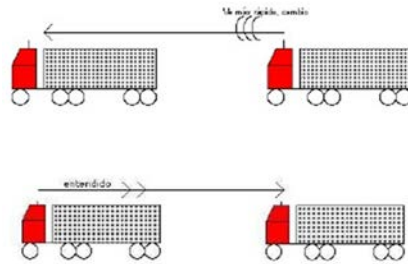
Las celdas le dan a los teléfonos un rango mayor a los dispositivos que lo comparamos. Una radio OC, puede transmitir hasta 5 millas.

Una persona que utiliza un teléfono celular, puede desplazarse por toda la ciudad y sin perder la conversación. Las celdas son las que permiten a los teléfonos celulares un mayor rango.

Siendo un sistema de comunicación telefónica totalmente inalámbrica, los sonidos se convierten en señales electromagnéticas, que viajan a través del aire, estas son recibidas y transformadas nuevamente en mensaje a través de antenas repetidoras o vía satélite.

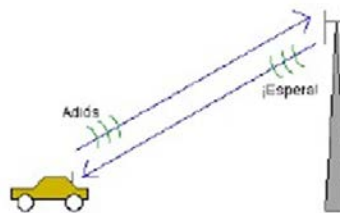
Para entender mejor el funcionamiento de estos aparatos sofisticados, ayuda compararlos con una radio de onda corta. Un radio de onda corta permite que dos personas se comuniquen usando la misma frecuencia, así sólo una

persona puede hablar al tiempo.



**Figura 1:** Transmisión de radio onda corta<sup>1</sup>

En una radio simple, los dos transmisores usan la misma frecuencia. Sólo uno puede hablar al tiempo



**Figura 2:** Transmisión de radio dual<sup>2</sup>

En una radio dual, ambos transmisores usan diferentes frecuencias, haciendo posible que las dos personas puedan hablar simultáneamente. Los teléfonos celulares son dispositivos duales.

El teléfono celular estándar de la primera generación estableció un rango de frecuencias que se encuentran entre los 824 y los 894 Megahertz para comunicaciones analógicas.

Para enfrentar la competencia, este estándar estableció el concepto de dos

<sup>1</sup> <http://jorgesalim.blogspot.com/2011/07/telefonía-celular-parte-2.html>

<sup>2</sup> <http://jorgesalim.blogspot.com/2011/07/telefonía-celular-parte-2.html>



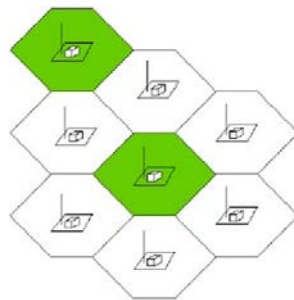
portadores conocidos como A y B. Para cada portador se asigna 832 frecuencias de voz, cada una de las cuales posee una amplitud de 30 KHz.

Una frecuencia para enviar y otra para recibir, son usadas para proporcionar un canal dual por teléfono. Las frecuencias de transmisión y recepción de cada canal están separadas por 45 Mhz. Además cada portador cuenta con 21 canales de datos para utilizarlos en otras actividades.

El teléfono celular puede dividir una ciudad en pequeñas celdas, que permiten ampliar la frecuencia por toda una ciudad, o incluso un país entero.

Esto permite que en un amplio territorio, millones de usuarios utilicen el servicio sin problema alguno.

Se puede dividir una ciudad en celdas. Cada celda es de un tamaño de 26Km cuadrados. Las celdas se imaginan como unos hexágonos en un campo hexagonal grande, como se muestra en Figura 3:



**Figura 3:** Celdas divididas por área<sup>3</sup>

Sin embargo, el tamaño de las celdas pueden variar dependiendo del sitio en que se encuentren. Las radio bases se encuentran separadas de 1 a 3 Km. en zonas urbanas, y en zonas rurales pueden llegar a separarse por más de 35Km.

<sup>3</sup> <http://www.youbioit.com/es/article/shared-information/2228/como-funcionan-los-telefonos-celulares-telefonos-moviles>

En zonas densamente pobladas o áreas con muchos obstáculos, las celdas pueden encontrarse en distancias menores. En consecuencia de su alta frecuencia y baja potencia en el que operan, algunas tecnologías, como los PCS (Personal Communication Services), necesitan celdas más cercanas unas de otras.

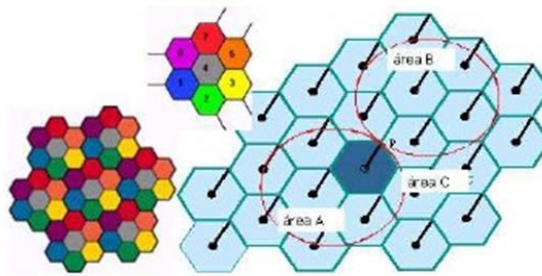
Los edificios pueden interferir con el envío de las señales entre las celdas que se encuentren más lejanas, por lo que es necesario que algunos edificios cuenten con su propia "microcelda."

Microceldas pueden ser utilizadas para amplificar la capacidad general de la red en algunas zonas con alta densidad poblacional como pueden ser los centros capitales.

Debido al uso de transmisores de bajo poder en las estaciones de base y en los teléfonos celulares, las mismas frecuencias pueden ser reutilizadas en celdas no adyacentes.

En un sistema análogo cada celda usa un séptimo de todos los canales de voz disponibles.

Una celda, más un arreglo hexagonal formado por las seis celdas que la rodean, cada una usando un séptimo de los canales para que cada una de las celdas tenga un grupo exclusivo de frecuencias y no dar paso a colisiones entre celdas adyacentes. Como puede verse en forma gráfica en la Figura 4:



**Figura 4:** Frecuencias reutilizadas en células<sup>4</sup>

En la figura 4 puede observarse un grupo de celdas numerado en la parte superior. De tal forma, en un sistema analógico, en cada una de las celdas pueden hablar 59 personas en sus teléfonos celulares de forma simultánea.

Con la transmisión digital, se incrementa el número de canales disponibles. Como por ejemplo el sistema digital TDMA puede abarcar el triple de llamadas por celda, dejando así alrededor de 168 canales disponibles al mismo tiempo.

Cada celda tiene una estación base que se constituyen de una torre y un edificio pequeño en donde se encuentra el equipo de radio. Cada celda usa un séptimo de los 416 canales duales. Dejando así por celda 59 canales disponibles nombrados anteriormente.

Los teléfonos celulares en su interior poseen un transmisor de bajo poder. Varios teléfonos celulares tienen 2 fuerzas de señal: 0.6 Watts y 3 Watts. La estación base también transmite a bajo poder.

## **2.1.1.6 SERVICIO DE MENSAJES CORTOS (SMS)**

### **2.1.1.6.1 DEFINICIÓN**

Servicio de Mensajes Cortos (SMS) es servicio inalámbrico aceptado

<sup>4</sup> <http://jorgesalim.blogspot.com/2011/07/telefonía-celular-parte-2.html>

globalmente este permite la transmisión de mensajes alfanuméricos entre clientes de teléfonos móviles y sistemas externos tales como correo electrónico, paging y sistemas de mensajes de voz.

Se debe enunciar que teorías se usarán y como se las aplicara para dar solución al problema planteado, Adicionalmente un análisis e interpretación de que se ha escrito y que se ha hecho sobre el tema propuesto. Al inicio del marco teórico se deberá presentar un mapa conceptual que resuma las teorías a aplicadas y que presenta la relación entre ellas.

#### **2.1.1.6.2 BENEFICIOS DEL SERVICIO DE MENSAJES CORTOS (SMS)**

Entre los beneficios del servicio de mensajes escritos SMS para el proveedor describimos los siguientes:

- Aumento de llamadas debido a la capacidad de notificación del SMS en las redes inalámbricas.
- Activa el acceso inalámbrico a datos para usuarios de empresas.
- Una alternativa al servicio de búsqueda de personas alfanumérico "Paging".
- Proporciona una herramienta administrativa para servicios como descargas en forma inalámbrica, avisos de precios.
- Provisiones de servicios con valor agregado como la integración de fax, buzón de voz, el e-mail, etc.

Los beneficios del mensaje de texto SMS a los clientes se orientan en la flexibilidad, conveniencia, acceso a datos y la integración de servicios de mensajes.

Desde esta perspectiva, el beneficio es la capacidad de utilizar un equipo móvil como una extensión del computador.

#### **2.1.1.6.3 APLICACIONES PARA SMS**

Los SMS fueron inicialmente diseñados para soportar mensajes de tamaño limitado, en la mayoría de los casos notificaciones o paginas alfanuméricas, pero se están descubriendo nuevos usos, que han hecho que este mercado se potencialice.

- **Servicios de notificación:** Los servicios de notificación son servicios SMS utilizados ampliamente. Ejemplos de servicios de notificación usando mensajes de texto SMS son los mensajes de notificación de correo de voz, notificación de correo electrónico, recordatorio de citas, horarios de reuniones, etc.
- **Interconexión de redes de correo electrónico:** Los servicios de correo electrónico existentes pueden ser fácilmente integrados con SMS para proveer correo electrónico bidireccional a la mensajería corta.
- **Interconexión de redes de búsqueda:** Servicios de búsqueda integrados con el SMS pueden permitir a los abonados inalámbricos digitales ser accesibles a través de interfaces de búsqueda existentes en otras redes.
- **Servicios de información:** Se puede suministrar una gran variedad de servicios de información, incluyendo partes meteorológicos, información del tráfico, información de entretenimiento (cines, teatros, conciertos), directorios, e información financiera ( servicios bancarios, servicios de corretaje, cotizaciones de bolsa , etc).
- **Servicios de datos móviles:** El SMSC también puede ser usado para en-

viar datos inalámbricos cortos.

- Algunos ejemplos de servicios de este tipo incluyen despachos rápidos, manejo de inventarios, confirmación de itinerarios, manejo de contactos de clientes y procesamiento de órdenes de ventas.
- Atención de clientes y administración: El SMSC también puede ser utilizado para transferir datos binarios que deben ser interpretados por la estación móvil, sin ser presentados al cliente.
- Servicios de localización: La habilidad de rastrear la localización de un objeto móvil, o de un usuario, es muy importante tanto para los clientes como para los proveedores. Esta aplicación, de nuevo, solo necesita un intercambio de pequeñas cantidades de información, tales como la latitud y longitud en un momento preciso del día, y quizás otros parámetros como velocidad, temperatura o humedad.

#### **2.1.1.7 JAVA**

Java es el primer lenguaje compilado e interpretado de forma simultánea. Cuando un programador realiza una aplicación en Java y lo compila, el compilador no funciona como compilador de un lenguaje al uso. Este compilador únicamente genera el denominado ByteCode.

Este código es un código intermedio entre el lenguaje máquina del procesador y Java.

Evidentemente este código no es ejecutable por sí mismo en ninguna plataforma hardware, puesto que no corresponde con el lenguaje de procesadores que actualmente se conocen.

Por lo tanto, para poner en funcionamiento una aplicación Java se necesita un mecanismo que permita ejecutar el Byte Code. Este mecanismo es la denominada Máquina Virtual Java. Cada plataforma ( Windows 95/NT, Macintosh, Unix, Linux, etc.) posee una máquina virtual específica. Así que cuando el Byte Code llega a la máquina virtual, ésta lo interpreta pasándolo a código máquina del procesador donde se esté trabajando, y ejecutando las instrucciones en lenguaje máquina que se deriven de la aplicación Java.

De este modo, cuando el Byte Code llega a diferentes plataformas, éste se ejecutará de forma correcta, puesto a que en cada una de esas plataformas existirá la máquina virtual adecuada. Debido a este mecanismo se consigue la conocida multiplataforma de Java, que con sólo codificar una vez, se podrá ejecutar en varias plataformas sin problemas.

#### **2.1.1.8 COMANDOS AT**

Los comandos AT son un conjunto de instrucciones codificadas que forman un lenguaje de comunicación entre un terminal modem y el hombre. El juego de comandos AT fueron desarrollados en el año de 1977 por Dennis Hayes como una interfaz de comunicación con un modem que permite configurarlo y proporcionarle instrucciones.

Con el avance del budio, las compañías US Robotics y Microcomm las que siguieron con la expansión y el desarrollo del juego de comandos hasta generalizarlo. Los comandos AT vienen de la abreviatura de attention.

Aunque el objetivo fundamental de los comandos AT es la comunicación con el módem, la telefonía móvil GSM adopto también como estándar este lenguaje

para permitir la comunicación con sus terminales.

Así, todos los teléfonos móviles GSM cuentan con su juego de comandos AT específico que funciona como interfaz para proporcionar instrucciones y configurar a los terminales. Estas instrucciones pueden encontrarse en la documentación técnica de los terminales GSM y permite acciones tales como escribir en la agenda de contactos, leer, enviar mensajes SMS y realizar llamadas de datos o de voz, entre otras opciones de configuración del terminal. La implementación de los comandos AT depende del dispositivo GSM y es indistinto del canal de comunicación por el cual estos comandos sean enviados, ya sea Bluetooth, canal Infrarrojos, cable de serie, etc. De esta manera, es posible distinguir distintos teléfonos móviles del mercado que permiten la ejecución total del juego de comandos AT o sólo parcialmente.

#### **2.1.1.8.1 CONTROL DE LLAMADA**

Estos son algunos de los comandos más comunes para el control de llamadas.

- AT Atención
- ATH Desconectar una llamada
- ATD Comando para Llamar
- ATA Contestar llamada

#### **2.1.1.8.2 COMANDOS SMS**

Un resumen de los comandos para GSM es:

##### **2.1.1.8.2.1 Comandos generales**

- AT+CPAS: Leer estado del modem



- AT+CIMI: Obtener el IMSI.
- AT+CGSN: Obtener número de serie
- AT+CGMI: Identificación del fabricante

#### **2.1.1.8.2.2 Comandos del servicio de red**

- AT+WOPN: Leer nombre del operador
- AT+CREG: Registrarse en una red
- AT+COPS: Selección de un operador
- AT+CSQ: Obtener calidad de la señal

#### **2.1.1.8.2.3 Comandos de seguridad:**

- AT+CPWD: Cambiar password
- AT+CPINC: Obtener el número de reintentos que quedan
- AT+CPIN: Introducir el PIN

#### **2.1.1.8.2.4 Comandos para la agenda de teléfonos**

- AT+CPBS: Buscar una entrada
- AT+CPBW: Almacenar una entrada
- AT+CPBF: Encontrar una entrada
- AT+CPBR: Leer todas las entradas

#### **2.1.1.8.2.5 Comandos para SMS**

- AT+ WMSC: Modificar el estado de un mensaje
- AT+CSCA: Establecer el Centro de mensajes a usar
- AT+CMSS: Enviar mensaje almacenado

- AT+CMGW: Almacenar mensaje en memoria
- AT+CMGS: Enviar mensaje SMS
- AT+CMGL: Listar los mensajes almacenados
- AT+CMGR: Leer un mensaje SMS almacenado
- AT+CMGF: Seleccionar formato de los mensajes SMS
- AT+CPMS: Seleccionar lugar de almacenamiento de los SMS.

## **2.1.2 MICROCONTROLADORES**

### **2.1.2.1 ¿QUÉ ES UN MICROCONTROLADOR?**

Al dispositivo electrónico capaz de llevar a cabo procesos lógicos se le conoce como microcontrolador. Estos procesos o acciones son programados en un lenguaje definido por el usuario, y son introducidos en este a través de un programador.

En el interior de un microcontrolador se encuentran las tres unidades funcionales de una computadora: unidades de E/S (entrada/salida), memoria y unidad central de procesamiento.

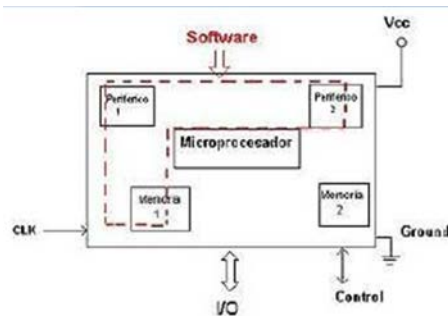
### **2.1.2.2 ESTRUCTURA BÁSICA DE UN MICROCONTROLADOR**

En la figura 5, vemos al microcontrolador dentro de un encapsulado, con su procesador (CPU), puertos de entrada salida , periféricos, memoria y buses.

Fuera del encapsulado se localizan otros circuitos para completar periféricos internos y otros dispositivos que se pueden conectar a los pines de entrada/salida (e/s).

También se conectarán a los pines del encapsulado la alimentación, masa, circuito de complementario del oscilador y otros circuitos indispensables para

el trabajo normal del microcontrolador.



**Figura 5:** Esquema de un microcontrolador.<sup>5</sup>

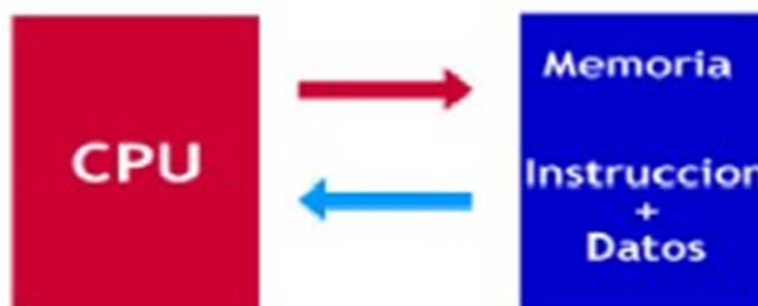
Aun cuando el microcontrolador es una computadora encapsulada, se encuentra formada por un núcleo y un conjunto de circuitos adicionales.

Dentro del núcleo se encuentran la memoria y el procesador.

### 2.1.2.3 ARQUITECTURA BÁSICA DE LOS MICROCONTROLADORES

#### 2.1.2.3.1 ARQUITECTURA VON NEUMANN

El esquema propuesto por John Von Neumann es la base de La arquitectura tradicional de computadoras y microprocesadores, en el cual la unidad central de proceso (CPU), está conectada a una memoria única que contiene los datos y las instrucciones del programa.



**Figura 6:** Arquitectura Von Neumann

Realizado por: Iván Gallardo

<sup>5</sup> <http://axnm.galeon.com/>

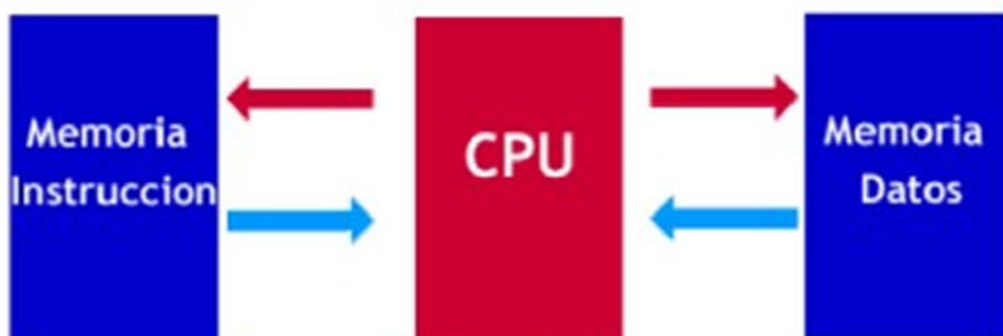
Las principales limitaciones de esta arquitectura son:

- La velocidad de operación está limitada tener un único bus de datos e instrucciones.
- La longitud de las instrucciones es reducida por la unidad de longitud de datos, por lo que requiere hacer varios accesos a memoria para buscar instrucciones complejas.

### 2.1.2.3.2 LA ARQUITECTURA HARVARD

La arquitectura conocida como Harvard, consiste en un esquema simple en el que el CPU está conectado a dos memorias por medio de dos buses separados. Una de las memorias solo almacena los datos y es llamada Memoria de Datos. La otra memoria contiene solamente las instrucciones del programa, y es llamada Memoria de Programa.

Los dos buses son independientes y de anchos distintos.



**Figura 7:** Arquitectura Harvard

**Realizado por:** Iván Gallardo

Las ventajas de esta arquitectura son:

- o La velocidad de acceso se incrementa al poseer acceso a los datos separado del acceso a instrucciones.
- o El tamaño de las instrucciones no está relacionado con el de los datos, por lo que permite que cada instrucción solo ocupe una sola posición de memoria de programa.

## **2.1.2.4 COMPONENTES DE UN MICROCONTROLADOR**

### **2.1.2.4.1 PROCESADOR**

Es el elemento más relevante del microcontrolador y establece sus principales características, tanto a nivel software como hardware. Es el encargado de recibir el código OP de la instrucción en curso direccionar, la memoria de instrucciones, su decodificación y la ejecución de la operación que implica la instrucción, así como la búsqueda de los operandos y el almacenamiento del resultado en un espacio de memoria. Existen tres orientaciones en cuanto a la arquitectura y funcionalidad de los procesadores actuales.

### **2.1.2.4.2 CISC**

Una diversidad de procesadores empleados en los microcontroladores están basados en la filosofía Computadores de Juego de Instrucciones Complejo (CISC). Cuentan con más de 80 instrucciones máquina en su recopilación, algunas de las cuales son muy potentes y sofisticadas, requiriendo muchos ciclos para la ejecución de procesos.

Una de las ventajas de los procesadores CISC es que brindan al programador instrucciones complejas que actúan como macros.

### **2.1.2.4.3 RISC**

La industria de los ordenadores comerciales como la industria de los microcontroladores están adoptando la filosofía Computadores de Juego de Instrucciones Reducido (RISC). La recopilación de instrucciones máquina en estos procesadores es bastante reducido y las instrucciones no son complicadas y, por lo general, se ejecutan en un solo ciclo.

La rapidez y sencillez de las instrucciones ayudan a optimizar el software y el hardware del procesador.

### **2.1.2.4.4 SISC**

Para los microcontroladores orientados a aplicaciones específicas, el juego de instrucciones, se acopla a las necesidades de dicha aplicación. A esta filosofía se la conoce con el nombre de Computadores de Juego de Instrucciones Específico (SISC).

### **2.1.2.4.5 MEMORIA**

#### **2.1.2.4.5.1 ROM CON MÁSCARA**

Es una memoria no volátil de sólo lectura, esto quiere decir que el contenido es grabado durante la fabricación del chip. Son fabricados en obleas que están compuestas por varias docenas de chips.

Estas obleas son fabricadas a partir de procesos fotoquímicos, donde son impregnadas con capas de óxido de silicio y silicio, y de acuerdo a la conveniencia, se erosionan al contacto con la luz. Para que no todos los puntos sean erosionados, se coloca una máscara con orificios entre la luz y la oblea, para que la luz incida sobre una determinada superficie.

#### **2.1.2.4.5.2 OTP**

Este microcontrolador contiene en su interior una memoria no volátil de sólo lectura, y esta solo puede ser programada una vez por el usuario. One Time Programmable (OTP). Es el usuario quien se encarga de escribir el programa dentro del chip mediante un grabador controlado por un software desde un ordenador.

La versión OTP es recomendable en series muy pequeñas o cuando el ciclo de diseño del producto es muy corto, y también en la construcción de prototipos. Para proteger el código contenido al igual que se lo realiza en la EPROM se suele usar encriptación mediante fusibles.

#### **2.1.2.4.5.3 EPROM**

Los microcontroladores que disponen de memoria Erasable Programmable Read Only Memory (EPROM) pueden grabarse y borrarse varias veces. La grabación se la realiza, con un grabador controlado desde un ordenador.

Si se desea borrar el contenido, en su superficie cuenta con una ventana de cristal por la que se expone a la EPROM a rayos ultravioleta por algunos minutos y la información grabada se borra por completo.

#### **2.1.2.4.5.4 EEPROM, E2PROM o E<sup>2</sup> PROM**

Son memorias de sólo lectura, programables y borrables eléctricamente Electrical Erasable Programmable Read Only Memory (EEPROM). La programación y el borrado de una EEPROM, se realiza eléctricamente desde el grabador bajo el control programado de un ordenador.

Los microcontroladores con memoria EEPROM una vez implementados en el circuito, se puede grabar y borrar cuantas veces sea necesaria sin ser desmontados del circuito. Para lo que se usa "grabadores en circuito" que conceden una gran rapidez y flexibilidad al momento de realizar cambios en la programación del trabajo grabado.

Una memoria EEPROM puede grabarse y borrarse un número infinito de veces, por lo que no se recomienda una reprogramación constante. Son aptos para la Ingeniería de diseño y la enseñanza. Va creciendo en los fabricantes la tendencia de reservar una zona pequeña de memoria EEPROM de los circuitos programables para modificar y guardar cómodamente una serie de parámetros que adaptan al dispositivo a las condiciones del entorno.

#### **2.1.2.4.5.5 FLASH**

Es una memoria no volátil, de reducido consumo, y puede ser escrita y borrada.

Su funcionamiento es similar a una ROM o una RAM pero su consumo es menor y su tamaño es reducido. La diferencia entre la ROM y la FLASH, es que la memoria FLASH puede ser programada en el circuito. Es de mayor densidad y más rápida que la EEPROM. Cuando se requiere una gran cantidad de memoria no volátiles de programa es recomendable la alternativa FLASH frente a la EEPROM. Es más rápida y soporta más ciclos de escritura/borrado. El permitir que los microcontroladores puedan ser reprogramados en el circuito hacen de las memorias EEPROM y FLASH muy útiles.



#### **2.1.2.4.6 PUERTAS DE E/S**

Las puertas E/S (Entrada y Salida) permiten la comunicación del procesador con el mundo exterior, por medio de interfaces, o a su vez con otros dispositivos. La utilidad primordial de las patas o pines de un microprocesador son estas puertas ó puertos. Según los controladores de periféricos que posee cada modelo de microcontrolador, las líneas de E/S se destinan a suministrar el soporte a las señales de control, entrada y salida.

#### **2.1.2.4.7 RELOJ PRINCIPAL**

Todos los microcontroladores cuentan con un circuito oscilador que se encarga de generar una onda cuadrada de frecuencia alta, que configura los pulsos de reloj usados para la sincronización de las operaciones del circuito. La señal del reloj hace que avancen el programa y los contadores.

Por lo general, el circuito de reloj es incorporado en el Microcontrolador y sólo es necesario añadir pocos componentes adicionales exteriores para estabilizar y seleccionar la frecuencia de trabajo. Estos componentes consisten en un resonador cerámico o una red R-C o bien un cristal de cuarzo complementado con elementos pasivos. El aumentar la frecuencia de los pulsos de reloj supone disminuir el tiempo de ejecución de las instrucciones pero como consecuencia se da un incremento de calor generado y del consumo de energía.

#### **2.1.2.4.8 RECURSOS ESPECIALES**

Los principales recursos específicos incorporados en los microcontroladores son:

- Timers o Temporizadores.
- Watchdog o Perro guardián.
- Brownout o Protección ante fallo de alimentación.
- Sleep mode (Estado de bajo consumo o de reposo).
- Conversor D/A (Digital - Analógico).
- Conversor A/D (Analógico - Digital).
- Puertas de comunicación.
- Puertas de E/S digitales.
- PWM (Pulse Wide Modulation) o Modulador de anchura de impulsos.
- Comparador analógico.

#### **2.1.2.4.9 TIMERS O TEMPORIZADORES**

Se utilizan para el control de periodos de tiempo y para contabilizar los sucesos que ocurren en el exterior del circuito. Para medir los tiempos se alimenta un registro con el valor correspondiente y a continuación el valor va aumentando o disminuyendo al ritmo de los pulsos de reloj.

Cuando se requiere contar sucesos que se plasman por cambios de nivel o flancos en alguno de los pines del microcontrolador, el registro se va aumentando o disminuyendo al ritmo de estos pulsos.

#### **2.1.2.4.10 WATCHDOG O PERRO GUARDIÁN**

Un microcontrolador funciona de forma continua las 24 horas del día y sin el control de un supervisor. La función Perro Guardián radica en un contador, que al llegar al máximo, provoca un reset automáticamente en el sistema.

Se debe diseñar el programa de trabajo que controle la tarea de forma que resetee al WatchDog de vez en cuando antes de que se provoque el reset. Al fallar el programa o al producirse un bloqueo (si cae en bucle infinito), no se refrescará al WatchDog y, al completar su temporización, provocará el reset del sistema.

#### **2.1.2.4.11 BROWNOUT O PROTECCIÓN ANTE FALLO DE ALIMENTACIÓN**

Consiste en un circuito encargado de resetear al microcontrolador cuando el voltaje de alimentación (VDD) sea inferior al voltaje mínimo (brownout). Mientras el voltaje de alimentación se mantenga por debajo al de brownout el dispositivo se mantendrá reseteado, empezando su funcionamiento normal cuando el voltaje sobrepase este valor. Es útil para evitar datos erróneos por transiciones y ruidos en la línea de alimentación.

#### **2.1.2.4.12 ESTADO DE REPOSO Ó DE BAJO CONSUMO**

Son cuantiosas las situaciones de trabajo en las que el microcontrolador debe esperar, inactivo, hasta que se produzca algún acontecimiento externo que le ponga nuevamente en actividad.

Para ahorrar energía, (factor indispensable en aparatos portátiles), los microcontroladores disponen de una instrucción especial (SLEEP en los PIC), que pasa al estado de bajo consumo o de reposo, en el que los requerimientos de potencia son reducidos. En dicho estado el reloj principal se detiene y se congelan sus circuitos asociados, quedando inmerso en un profundo sueño.

Al presentarse una interrupción ocasionada por el suceso esperado, el

microcontrolador se despierta y reanuda su actividad.

#### **2.1.2.4.13 CONVERTOR A/D (CAD)**

Los microcontroladores que cuentan con un Conversor A/D (Analógico/Digital) pueden procesar señales analógicas.

Algunos disponen de un multiplexor que permite inyectar a la entrada del CAD diversas señales analógicas desde los pines del circuito integrado.

#### **2.1.2.4.14 CONVERTOR D/A (CDA)**

Su principal función es transformar datos digitales provenientes del procesamiento del computador en su respectiva señal analógica que saca al exterior por uno de los pines del chip.

#### **2.1.2.4.15 COMPARADOR ANALÓGICO**

Algunos modelos de microcontroladores cuentan internamente con un Amplificador Operacional que actúa como comparador entre una señal fija de referencia y otra señal variable que es inyectada a uno de los pines del encapsulado. La salida del comparador da como resultado un nivel lógico 1 ó 0 dependiendo si es mayor o menor una señal de la otra.

Existen modelos de microcontroladores con un módulo de tensión de referencia que proporciona gran variedad de tensiones de referencia que pueden ser aplicadas en los comparadores (Amplificador Operacional).

#### **2.1.2.4.16 MODULADOR DE ANCHURA DE IMPULSOS O PWM**

Son circuitos que entregan a la salida pulsos variables en anchura, que se

entregan al exterior a través de los pines del encapsulado.

#### **2.1.2.4.17 PUERTOS DIGITALES DE E/S**

Todos los microcontroladores reservan parte de sus pines a soportar líneas de E/S digitales. Comúnmente, estas líneas se agrupan de ocho en ocho para formar puertos.

Las líneas digitales de los Puertos pueden ser configurados como Entrada o como Salida cargando un 1 ó un 0 en el bit indicado de un registro destinado para su configuración.

#### **2.1.2.4.18 PUERTAS DE COMUNICACIÓN**

Con el fin de permitir al microcontrolador comunicarse con distintos dispositivos externos, buses de redes, buses de sistemas, otros buses de microprocesadores, y poder acoplarlos con otros elementos bajo otros protocolos y normas.

Algunos modelos cuentan con recursos que posibilitan directamente esta tarea, entre los que se destacan:

**Universal Serial Bus (USB)**, es un bus serie moderno para los PC.

**UART**, adaptador de comunicación serie asíncrona. (Ej: Puerto Serie)

**USART**, adaptador de comunicación serie síncrona y asíncrona

**PUERTA PARALELA ESCLAVA** para permitir la conexión con los buses de otros microprocesadores.

**Controller Área Network (CAN)**, para permitir la adaptación con redes de conexasión multiplexado elaborado conjuntamente por Intel y Bosch para el cableado de dispositivos en automóviles.

**Bus I2C**, es una interfaz serie de dos hilos elaborados por Philips.

#### **2.1.2.5 FAMILIA DE LOS PIC**

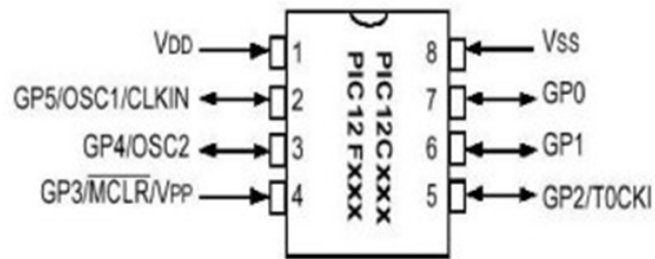
Los PIC son microcontroladores elaborados por Microchip cuya arquitectura, juego de instrucciones, capacidades y en especial su bajo costo lo hacen muy útil en pequeñas aplicaciones así como parte de otras aplicaciones de mayor dimensión sustituyendo a una gran parte de circuitos lógicos convencionales.

Existen PIC's de cuatro gamas diferentes:

##### **2.1.2.5.1 GAMA ENANA. PIC12CXXX**

Es un grupo de PIC de reciente aparición que ha acaparado la atención del mercado. Su principal característica es su reducido tamaño, al contar todos sus componentes de 8 pines además de disponer de 6 líneas de Entrada y Salida.

Se alimentan con un voltaje entre 2,5 V y 5,5 V de corriente continua, y su consumo cuando trabajan a 5 V y 4 MHz es menor a 2 mA. El formato de instrucciones puede ser de 12 o de 14 bits y su compilación es de 33 o 35 instrucciones.



**Figura 8:** Gama Enana Pic12F675<sup>6</sup>

#### 2.1.2.5.2 GAMA BAJA. PIC16C5XX

Con una memoria de programa (EPROM o ROM) de 12 bits, 512, 1024 o 2048 palabras y memoria de datos de 8 bits, 25, 72 ó 73 bytes. Trabajan hasta 20 MHz y disponen de 12 o 20 líneas de E/S de alta corriente, un temporizador y de 33 instrucciones.

Además, como el resto de los PICs cuentan con un perro guardián, Auto inicialización (POR ó Power on Reset), modo de bajo consumo (SLEEP), reloj interno mediante una red RC ó cristal, además protección contra lectura del código.

#### 2.1.2.5.3 GAMA MEDIA. PIC16CXXX CON INSTRUCCIONES DE 14 BITS

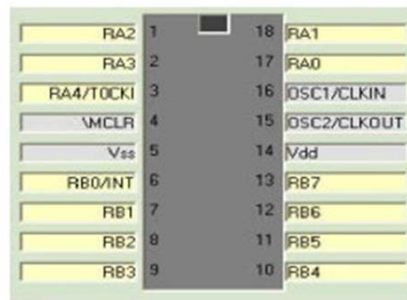
Esta gama es la más completa y variada de los PIC. Comprende modelos con encapsulados desde 18 pines hasta 68, cubriendo varias opciones que integran abundantes periféricos.

En esta gama se encuentra el famoso PIC16X84 y sus variantes.

La memoria de programa tiene un ancho de 14 bits, pudiendo ser ROM,

<sup>6</sup> <http://www.info-ab.uclm.es/labelec/solar/Microcontroladores/Conexiones%20de%20Pic's.htm>

EPROM o EEPROM. Incluyen interrupciones, conversores A/D, mayor número de temporizadores y otras características según el modelo. Su juego de instrucciones es de 35.



**Figura 9:** Gama Media PIC16F84<sup>7</sup>

#### 2.1.2.5.4 GAMA ALTA: PIC17CXXX CON INSTRUCCIONES DE 16 BITS

Los microcontroladores de gama alta disponen de un verdadero bus de datos y direcciones, pudiendo trabajar con memoria externa. La memoria de programa tiene un ancho de 16 bits y el número de instrucciones aumenta a 55 o 58 según el modelo.

La característica más destacable de esta gama es su arquitectura abierta, esta radica en la posibilidad de ampliación del microcontrolador con elementos externos. Para esto, los pines sacan al exterior las líneas de los buses de datos, control y direcciones, a las que se conectan controladores de periféricos o memorias.

Esta facultad obliga a los componentes a tener un gran número de pines comprendido entre 40 y 44. Esta filosofía de construcción del sistema es la que

<sup>7</sup> <http://www.info-ab.uclm.es/labelec/solar/Microcontroladores/Conexiones%20de%20Pic's.htm>



se emplea los microprocesadores y no suele ser una práctica usual cuando se emplean microcontroladores.

Microchip dividió sus microcontroladores en tres subfamilias grandes de acuerdo al número de bits de su bus de instrucciones:

Subfamilia	instrucciones	nomenclatura
Base - Line	33 instrucciones de 12 bits	PIC12XXX y PIC14XXX
Mid - Range	35 instrucciones de 14 bits	PIC16XXX
High - End	58 instrucciones de 16 bits	PIC17XXX y PIC18XXX

**Tabla 3:** Nomenclatura del microcontrolador<sup>8</sup>

Existen algunas excepciones, como el PIC16C5XX que utiliza 33 instrucciones de 12 bits (cuenta con encapsulados de 18 y 28 pines, se energiza con 2.5 V)

Algunos autores manejan una “gama enana” que son los PIC12C508 y PIC12C509 en empaquetados de 8 pines y con un bus de instrucciones de 12 ó 14 bits.

### 2.1.3 SONIDO

El sonido, es cualquier fenómeno que involucra la propagación en forma de ondas elásticas (sean audibles o no), por lo general a través de un fluido (o algún otro medio elástico) que esté generando el movimiento vibratorio de un cuerpo.

Los sonidos audibles por el humano consisten en ondas sonoras que provocan oscilaciones de la presión del aire, estas oscilaciones son convertidas en ondas

<sup>8</sup> <http://www.info-ab.uclm.es/labelec/solar/Microcontroladores/Conexiones%20de%20Pic's.htm>

mecánicas en el oído humano y percibidas por el cerebro. En los fluidos la propagación del sonido es similar, donde el sonido toma la forma de variaciones de presión. En los cuerpos sólidos la propagación del sonido involucra variaciones del estado de tensión del medio.

La transmisión del sonido involucra transporte de energía sin transporte de masa, en forma de ondas mecánicas que se propagan a través de la materia sólida, líquida o gaseosa. Como el sonido se propaga en la misma dirección que las vibraciones, se trata de una onda longitudinal.

Para generar un sonido es necesaria la vibración de alguna fuente. Las vibraciones pueden transmitirse a través de diversos medios elásticos, entre los más comunes se encuentran el agua y el aire. La fonética acústica concentra su interés especialmente en los sonidos del habla: el cómo estos son percibidos, cómo se generan, y cómo se pueden describir gráfica y/o cuantitativamente.

### **2.1.3.1 Propagación del sonido**

Ciertas características de los sólidos y de los fluidos se interponen en la onda de sonido. Por esta razón el sonido se propaga en los sólidos y en los líquidos con mayor velocidad que en los gases. En general cuanto mayor sea la compresibilidad ( $1/K$ ) del medio tan menor es la velocidad de propagación del sonido. La densidad también es un factor fundamental en la velocidad de propagación, y por general entre mayor sea la densidad ( $\rho$ ), menor es la velocidad de la propagación del sonido en el fluido. La velocidad del sonido se

relaciona con estas magnitudes mediante la siguiente fórmula:

$$v \propto \sqrt{\frac{K}{\rho}}$$

La temperatura en los gases, influye tanto en la compresibilidad como en la densidad, de tal forma que el factor de importancia es la temperatura.

Para la transmisión del sonido es necesario que las moléculas vibren alterando sus posiciones de equilibrio.

### **2.1.3.2 Magnitudes físicas del sonido del silencio**

El sonido puede ser representado por la suma de curvas sinusoides: Frecuencia ( $f$ ) o inversa del período ( $T$ ), longitud de onda ( $\lambda$ ), amplitud (indica la cantidad de energía que contiene una señal sonora). Cuando se considera la superposición de distintas ondas es importante considerar la fase, que representa el retraso relativo en la posición de una onda con respecto a otra.

No obstante, un sonido complejo no está caracterizado por los parámetros mencionados, ya que en general un sonido es la combinación de ondas sonoras que se diferencian en los cinco parámetros anteriores. La caracterización de un sonido caprichosamente complejo implica analizar tanto la energía transmitida como la colocación de dicha energía entre las diversas ondas componentes.

**Potencia acústica:** El nivel de potencia acústica es la cantidad de energía irradiada por una fuente determinada en forma de ondas por unidad de tiempo.

La potencia acústica depende directamente de la amplitud.

Espectro de frecuencias: Este permite conocer en qué frecuencias se transmite una cantidad considerable de la energía irradiada.

### **2.1.3.3 Fisiología del sonido**

#### **2.1.3.3.1 Aparato auditivo**

Los sonidos son receptados a través del aparato auditivo que percibe las ondas sonoras, que son transformadas en movimientos de los osteocillos óticos y percibidas en el oído interno al mismo tiempo son transmitidas mediante el sistema nervioso al cerebro.

#### **2.1.3.3.2 La voz humana**

La voz humana se produce por la vibración de las cuerdas vocales, lo que genera una onda sonora que es el resultado de la combinación de diversas frecuencias y sus respectivos armónicos. La cavidad buco-nasal crea ondas casi estacionarias. Cada fragmento de sonido del habla está caracterizado por un determinado espectro de frecuencias o distribución de la energía sonora en las distintas frecuencias.

#### **2.1.3.4 Sonidos del habla**

Las lenguas humanas usan segmentos homogéneos identificables de unas decenas de milisegundos de duración, que conforman los sonidos del habla, técnicamente conocidos como fonos. Lingüísticamente no todas las diferencias acústicas son importantes, por ejemplo las mujeres y los niños tienen tonos más agudos, por lo que todos los sonidos elaborados tienen una frecuencia fundamental y algunos armónicos más intensos y altos.

Los hablantes expertos de alguna lengua aprenden a "catalogar" distintos sonidos cualitativamente semejantes en clases de simetría de rasgos relevantes. Esas clases de simetrías reconocidas por los hablantes son los constructos mentales que se denominan fonemas. La mayoría de lenguas naturales poseen unas pocas decenas de fonemas característicos, a pesar de que las variaciones acústicas de los fonos y sonidos son marcadas.

### **2.1.3.5 Señal de audio**

La señal de audio es una señal analógica eléctricamente exacta a una señal sonora; por lo general está limitada al rango audible de frecuencias para el humanos que se encuentra aproximadamente entre los 20 y los 20.000 Hz, (el equivalente, cuasi exacto a 10 octavas).

Debido a que el sonido es una onda de presión se necesita un micrófono (un transductor de presión) que convierte las ondas sonoras en señales analógicas (señales eléctricas).

La conversión inversa se realiza por medio de un altavoz —también conocido como altoparlante, que transforma las señales eléctricas en ondas de presión de aire (ondas sonoras).

Un sólo micrófono tiene la capacidad de captar adecuadamente todo el rango de frecuencias audibles, por el contrario para reproducir fidedignamente el mismo rango de frecuencias se requiere dos altavoces (graves y agudos).

Una señal de audio se puede distinguir, superficialmente, por su dinámica (rango dinámico, relación señal-ruido, potencia, valor de pico) o por su composición espectral (distorsión armónica, frecuencia fundamental, ancho de

banda, armónicos, etc.).

Así, por ejemplo, una señal que represente voz humana (señal vocal) no suele tener información relevante más allá de los 10 kHz, y de hecho en telefonía fija se toman sólo los primeros 3.8 kHz. Basta 2 kHz para que la voz sea perceptible, pero no para distinguir al hablante.

## **2.2 Marco conceptual**

### **2.2.1. MICROCONTROLADORES**

Se utilizará para llevar a cabo procesos lógicos y la sincronización de la base de datos con el circuito que es el encargado de enviar la información a través del modem GSM, a los distintos dispositivos móviles registrados.

### **2.2.2. Java**

Se utilizará esta herramienta para el desarrollo de los controles del programa este se usara como compilador en que se creara los enlaces y acceso a las bases de datos.

### **2.2.3. Modem Celular**

El modulador/demodulador que se utilizará para la comunicación entre el servidor que está en SQL Server y el celular para obtener la información de notas del estudiante.

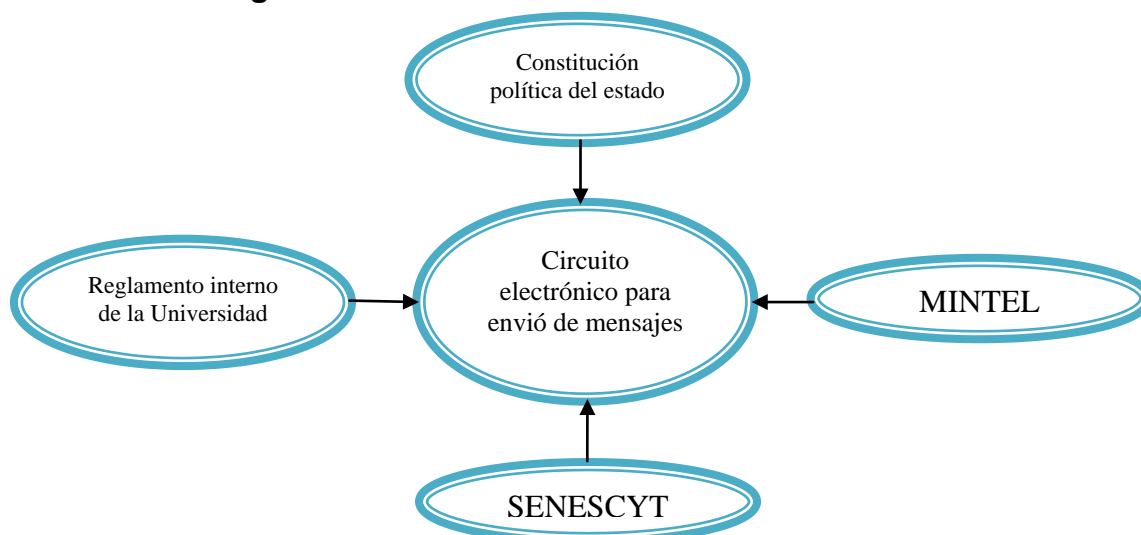
### **2.2.4. Interfaz RS232**

Es el puerto que se usará para la comunicación del modem con el celular para enviar la consulta de saldos y adicionalmente enviar la información relevante al estudiante.

Este comando se aplicará en el proyecto para la configuración del modem:

- Para inicializar el modem se utiliza el siguiente comando. **At+csmp**
- Para el envío a formato sms se utilizara este comando. **At+cmgf**
- Para recibir el mensaje sms con el siguiente comando **At+cnmi**
- Para guardar la configuración de los sms se usara **At+csas**
- Para inicializar las conversación para el envío de notas es el siguiente comando. **At+cmgs**
- Para iniacilizar un indicador de mensaje de entrada se utilizara **At+cmti**
- Para leer mensajes guardados que llego, se usara este comando. **At+cmgr**
- Para leer mensajes guardados que han estado almacenados se usara el siguiente comando. **At+cmss**
- Para ver la lista de mensajes almacenados en la tarjeta sim se usara el siguiente comando. **At+cmgl**
- Para borrar mensajes que están guardados en la sim en una localización específica se usara el siguiente comando. **At+cmgd**

### 2.3. Marco Legal



Realizado por: Iván Gallardo

Cabe aclarar que el tipo de servicio que se va a prestar no requiere un permiso de funcionamiento especial, ya que al trabajar con alguna de las operadoras existentes en el país, estas ya cuentan con los permisos respectivos para este tipo de servicio.

#### **2.4. Marco Espacial**

Este servicio será implementado en la Universidad Tecnológica “Israel”, estará disponible para su uso las 24 horas del día de lunes a domingos, y podrá ser utilizado tanto por docentes como estudiantes para mantener una comunicación oportuna y eficiente.



## **CAPÍTULO III**

### **METODOLOGÍA**

#### **3.1. Proceso de investigación**

##### **3.1.1. Unidad de Análisis**

Para llevar a cabo la realización del proyecto se necesita conocer claramente la problemática actual que posee la Universidad Israel.

##### **3.1.2. Tipo de Investigación.**

En primera instancia se utilizará la investigación documental para recopilar información y fotos necesarias para el diseño del producto. Se utilizarán fichas bibliográficas

##### **3.1.3. Método**

El método a utilizar en el desarrollo será el método inductivo, con el propósito de mejorar el sistema de anuncios y consulta de saldos en la Universidad Israel, ya que en la actualidad se puede realizar esta consulta sólo en las instalaciones de la Universidad, provocando así inconformidad, pérdida de tiempo y en ocasiones problemas por parte de los estudiantes al no conocer a tiempo información importante sobre actividades a realizarse.

##### **3.1.3. Técnica**

Para desarrollar la investigación del proyecto se utilizara la técnica de la entrevista, esta se realizara en la Universidad Israel, las unidades de análisis para el

proyecto serán el área Administrativa, decanos de las diferentes Facultades, esto con el fin de conocer si se cuenta con el apoyo y aceptación por medio de la Universidad, además se utilizara la técnica de la encuesta, ésta se la realizara a los estudiantes de octavo nivel, sección nocturna de la Universidad Israel, para así conocer la aceptación que tiene la implementación del sistema de anuncios y consulta de saldos.

#### **3.1.4. Instrumento**

El instrumento utilizado para la recolección de datos en las encuestas será el cuestionario, esto con el objetivo de conocer el grado de aceptación de la nueva alternativa de envío de anuncios y consulta de saldos por parte de los estudiantes de modalidad nocturna. Tanto la entrevista como la encuesta serán realizadas de una forma confiable y estricta para que tengan validez a la hora de obtener los datos y realizar el proyecto. Para la investigación documental se utilizará la lectura científica que permita manejar conceptos básicos y esenciales, los cuales nos permiten conocer más a fondo los términos a utilizar en la elaboración del prototipo del sistema.

#### **3.2. Metodología específica**

Para el presente proyecto se utilizará el método científico ya que las variables propuestas serán susceptibles de experimentación y comprobación, por tanto se procederá primeramente a la investigación de teorías y conocimientos existentes acerca de la temática, para luego pasar a la formulación de propuestas y realización esquemática, en la fase de diseño. Por último se

realizará la comprobación de lo planteado en la fase de implementación y pruebas.

**Diseño:** Para el diseño del producto se requerirá una investigación de tipo documental, utilizando el método de análisis y síntesis para determinar los resultados de los datos, dentro de las técnicas de procesamiento de datos se encuentran diferentes tipos de diagramas esquemáticos, físico, por bloques, etc.

**Implementación y pruebas:** En esta fase se utilizarán los métodos: observación, deductivo-inductivo y la experimentación, para determinar si el producto cumple con todas las características definidas en la fase del diseño.

## CAPÍTULO IV

### DESARROLLO

#### 4.1. Estudio del tablero electrónico sonoro

##### 4.1.1 Introducción

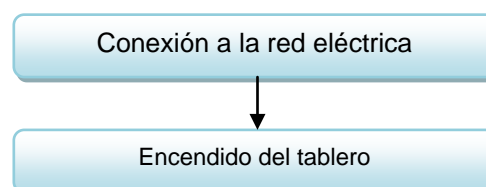
En el presente capítulo se describe el proceso realizado para elaborar un tablero electrónico sonoro, activado mediante un mensaje de texto, capaz de mostrar fechas de inicio de matriculas, pago de pensiones, inicio de clases. Así como también la descripción del sistema lógico implementado, sus limitaciones y funcionamiento.

#### 4.2 Descripción de funcionamiento

En la descripción de funcionamiento se explica la forma de operación del tablero electrónico sonoro, las conexiones entre los elementos y la operación de los códigos programados en el microprocesador utilizado.

##### 4.2.1 Encendido del tablero

En el diagrama siguiente se describe las acciones del tablero electrónico sonoro al encenderse, indicando su inicio de funcionamiento, encendiendo los leds del tablero.

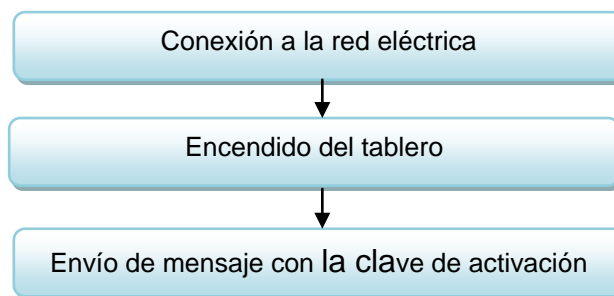


**Diagrama1:** Inicio de funcionamiento del tablero

**Realizado por:** Iván Gallardo

#### 4.2.2 Activación del tablero electrónico

Luego de haber iniciado su funcionamiento, la activación del tablero se realiza enviando un mensaje de texto a través de un teléfono celular, con la clave para ser validada.

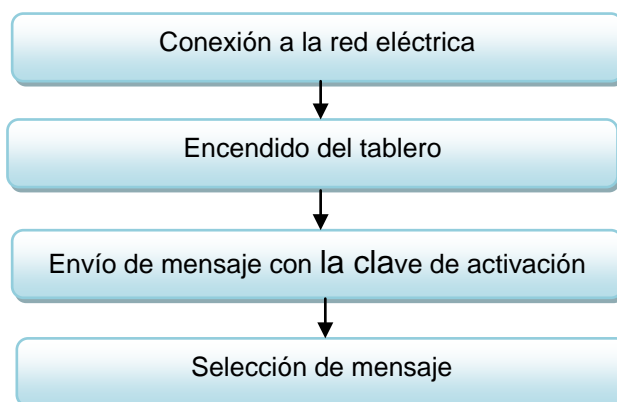


**Diagrama 2.** Activación del tablero

Realizado por: Iván Gallardo

#### 4.2.3 Respuesta del Tablero electrónico sonoro después de validar la clave enviada

El tablero al recibir el mensaje con la clave, valida y extrae la información que requiere para mostrar el mensaje en el tablero.

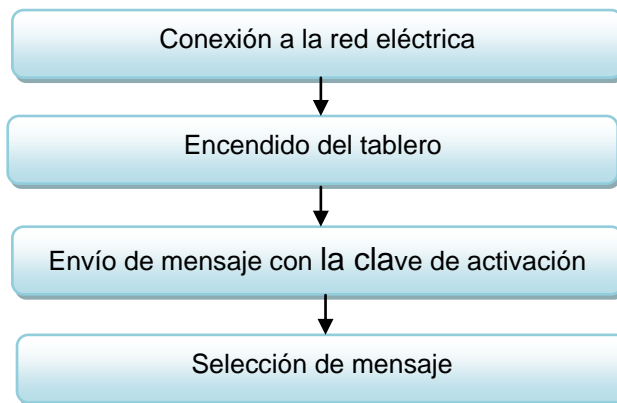


**Diagrama 3.** Respuesta del tablero

Realizado por: Iván Gallardo

#### 4.2.4 Selección de mensaje

Para seleccionar el mensaje es necesario enviar el mensaje con el formato correcto para poder mostrar en el tablero y reproducir el sonido correspondiente al mensaje.

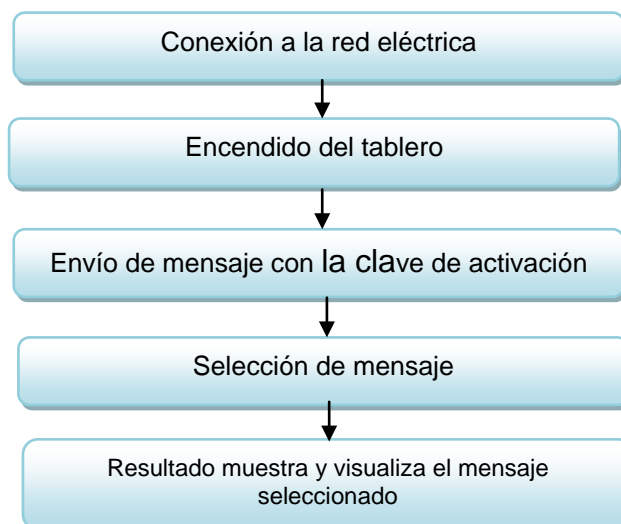


**Diagrama 4.** Selección de mensaje

Realizado por: Iván Gallardo

#### 4.2.5 Resultado final

El tablero muestra el mensaje seleccionado con la información que se desea y reproduce un sonido pregrabado que acompaña la información que se muestra en el tablero



**Diagrama 5.** Muestra y visualiza el mensaje

Realizado por: Iván Gallardo

### **4.3 Delimitación de la actividad del tablero electrónico sonoro**

Es necesario para el desarrollo, conocer las restricciones que se tienen, el medio donde va a instalarse el tablero.

- Los mensajes a mostrarse en el tablero deben ser previamente establecidos.
- Los mensajes que van a desplegarse en el tablero son de carácter informativo, por lo que se debe validar el ingreso al menú de opciones del tablero, para que no cualquiera pueda modificarlas.
- El objetivo del mensaje sonoro es llamar la atención de los estudiantes, y se mantengan informados.

### **4.4 Descripción del sistema**

El tablero electrónico sonoro permitirá mantener informado a los estudiantes de una manera diferente y práctica, llamando la atención de los mismos por su diseño innovador, haciendo que muestren interés en la información mostrada en el tablero.

### **4.5 Avance tecnológico**

Actualmente el uso de pantallas de LEDs se ha extendido y masificado. Su uso principal es mostrar Información y Publicidad a largas distancias. En comparación con otras formas de publicidad impresa, las Pantallas de LED ofrecen una forma de cambio de Información más rápido y más fácil. También el cambio se puede realizar de una forma centrada, que significa una ventaja grande para la comercialización de Publicidad. En áreas públicas las pantallas

de Led ofrecen una ventaja por su posibilidad de actualizar su información mediante un ordenador o dispositivo que pueda comunicarse.

Los usos pueden ser tan simples como un Indicador de Turno hasta Pantallas que indiquen información actualizada en Instituciones, Cines, Transporte Público hasta en Vías ó Autopistas.

Por su consumo bajo en energía y su durabilidad las Pantallas de LED son una buena alternativa para visualizar Información en áreas públicas.

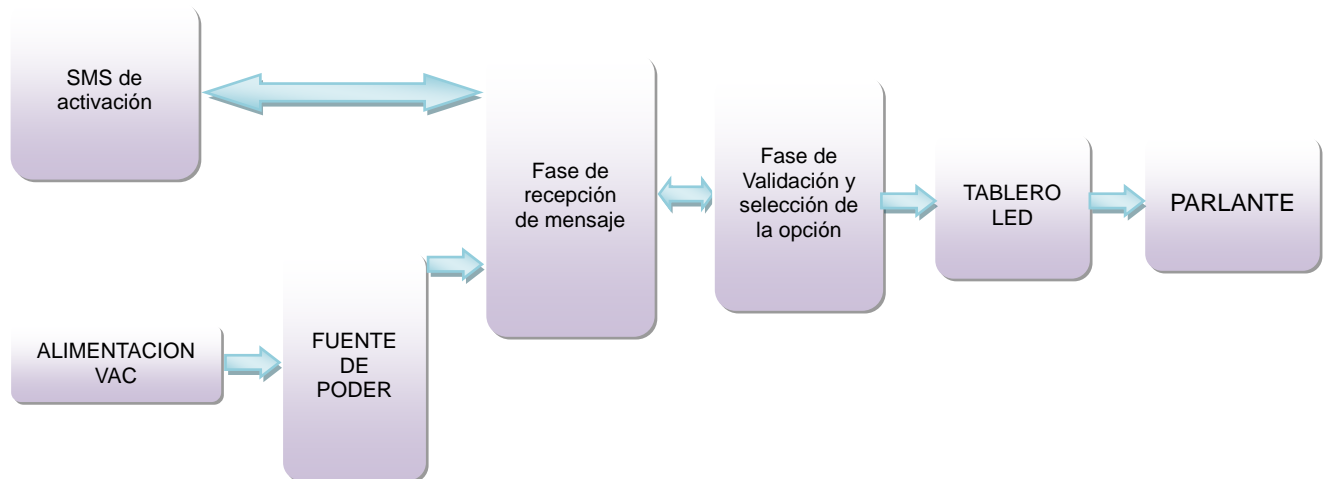
Las pantallas electrónicas de LED se pueden encontrar en diversas presentaciones como lo son:

- Pantallas electrónicas de LED de instalación fija.
- Pantallas electrónicas de LED para renta o uso móvil.
- Pantallas electrónicas de LED perimetrales o bien para los laterales de estadios.
- Pantallas electrónicas de LED tipo cortinas.
- Pantallas electrónicas de LED Flexibles.
- Pantallas electrónicas de LED tipo para piso o pista de baile



## 4.6 DISEÑO DEL TABLERO ELECTRONICO SONORO

### 4.6.1 Diagrama de bloques.



Realizado por: Iván Gallardo

**Diagrama 6.** Diagrama de bloques del sistema

### 4.6.2 SMS de activación

#### Introducción

Un SMS es un servicio de mensajes cortos de texto que en la actualidad están siendo usados con varios propósitos, en bancos, activación de alarmas, casas inteligentes, etc. En nuestro caso será utilizado para activar y seleccionar la opción a mostrar en el tablero electrónico sonoro.

### 4.6.3 Fuente de poder

#### Introducción

En electrónica, una fuente de alimentación es un dispositivo que convierte el voltaje alterno de la red de suministro, en voltaje continua, que alimentan los distintos circuitos del aparato electrónico al que se conecta.

La fuente de alimentación, está diseñada para proporcionar voltaje de alimentación para el PIC, módulo GSM, los diodos led y el sistema de audio.

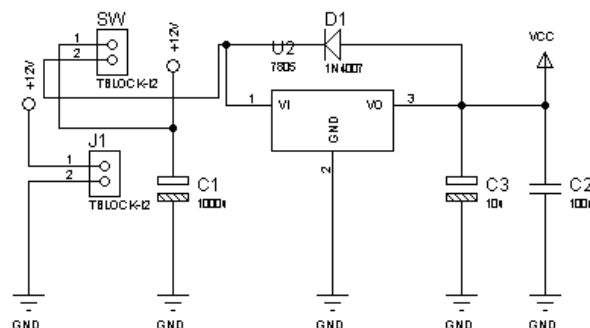
#### 4.6.3.2 Fuente lineal

Las fuentes lineales siguen el esquema: transformador, rectificador, filtro, regulación y salida.

En primer lugar el transformador adapta los niveles de tensión y proporciona aislamiento galvánico. El circuito que convierte la corriente alterna en continua se llama rectificador, después suelen llevar un circuito que disminuye el rizado como un filtro de condensador.

La regulación, o estabilización de la tensión a un valor establecido, se consigue con un componente denominado regulador de tensión, la salida puede ser simplemente un condensador. Esta corriente abarca toda la energía del circuito, razón por la que debe proporcionar toda la energía (voltaje y amperaje) para el funcionamiento del tablero.

#### 4.6.3.3 Diseño de la fuente poder



**Ilustración 1.** Diseño de la fuente

**Realizado por:** Iván Gallardo

## 4.6.4 Fase de recepción de mensaje

### 4.6.4.1 Módulo GSM

#### Descripción

Un módulo GSM se controla mediante comandos AT, este módulo recibe el mensaje y lo transfiere a través de un puerto serial, a un microcontrolador que se el encargara de activar los demás circuitos.

#### Especificación

- Fuente de alimentación: 12v
- Bajo consumo de energía (100 mA @ 7v - Modo GSM)
- Quad-Band 850/900/1800/1900MHz

#### Imagen de referencia



**Ilustración 2.** Módulo GSM

#### **4.6.4.2 Capacitores electrolíticos**

##### **Descripción**

Un condensador electrolítico es un tipo de condensador que usa un líquido iónico conductor como una de sus placas. Típicamente con más capacidad por unidad de volumen que otros tipos de condensadores, son valiosos en circuitos eléctricos con relativa alta corriente y baja frecuencia. Este es especialmente el caso en los filtros de alimentadores de corriente, donde se usan para almacenar la carga, y moderar el voltaje de salida y las fluctuaciones de corriente en la salida rectificada.

También son muy usados en los circuitos que deben conducir corriente alterna pero no corriente continua.

##### **Capacidades**

El valor de la capacidad de cualquier condensador (también conocido como capacitancia) es una medida de la cantidad de carga almacenada, por unidad de diferencia de potencial entre sus placas.

La unidad básica de capacidad en el sistema internacional de unidades es el faradio que es un culombio por voltio. Sin embargo, esta unidad es muy grande para las capacidades típicas de los condensadores reales, de forma que el microfaradio, nanofaradio y picofaradio se usan más comúnmente, unidades que se abrevian como  $\mu\text{F}$  o  $\text{uF}$ ,  $\text{nF}$  y  $\text{pF}$ .

### Imagen de referencia



**Ilustración 3.** Condensadores electrolíticos<sup>9</sup>

#### 4.6.4.3 Capacitores cerámicos

### Descripción

Capacitores o también llamados condensadores cerámicos, son los que tienen un mayor rango de valores de su constante dieléctrica, pudiendo llegar a un valor de 50000 veces superior a la del vacío. Se basan en varias mezclas de óxido de titanio y zirconio, o bien en titanatos o zirconatos de calcio, bario, estroncio o magnesio, y atendiendo a esta variedad de compuestos, dan un rango amplísimo de constantes dieléctricas.

### Imagen de referencia



**Ilustración 4.** Condensadores cerámicos<sup>10</sup>

<sup>9</sup> <http://rubkev.wordpress.com/category/uncategorized/>

<sup>10</sup> <http://rubkev.wordpress.com/category/uncategorized/>

#### 4.6.4.4 Diodos LED

##### Descripción

El LED es un tipo especial de diodo, que trabaja como un diodo común, pero que al ser atravesado por la corriente eléctrica, emite luz. Existen diodos LED de varios colores que dependen del material con el cual fueron construidos. Hay de color rojo, verde, amarillo, ámbar, infrarrojo, entre otros. Eléctricamente el diodo LED se comporta igual que un diodo de silicio o germanio, si se pasa una corriente a través del diodo semiconductor, se inyectan electrones y huecos en las regiones P y N, respectivamente.

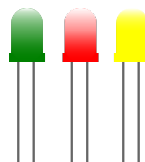
##### Aplicaciones

Se utiliza ampliamente en aplicaciones visuales, como indicadores de cierta situación específica de funcionamiento.

##### Ejemplos

- Se utilizan para desplegar contadores.
- Para indicar la polaridad de una fuente de alimentación de corriente continua.
- Para indicar la actividad de una fuente de alimentación de corriente alterna.
- En dispositivos de alarma, etc.

##### Figura de referencia



**Ilustración 5.** Diodos LED<sup>11</sup>

---

<sup>11</sup> <http://es.wikipedia.org/wiki/Led>

#### 4.6.4.5 Resistencias

##### Descripción

Se denomina resistor al componente electrónico diseñado para introducir una resistencia eléctrica determinada entre dos puntos de un circuito. En el propio argot eléctrico y electrónico, son conocidos simplemente como resistencias.

Es un material formado por carbón y otros elementos resistivos para disminuir la corriente que pasa. Se opone al paso de la corriente. La corriente máxima en un resistor viene condicionada por la máxima potencia que puede disipar su cuerpo, esta potencia se puede identificar visualmente a partir del diámetro sin que sea necesaria otra indicación.

##### Código de colores

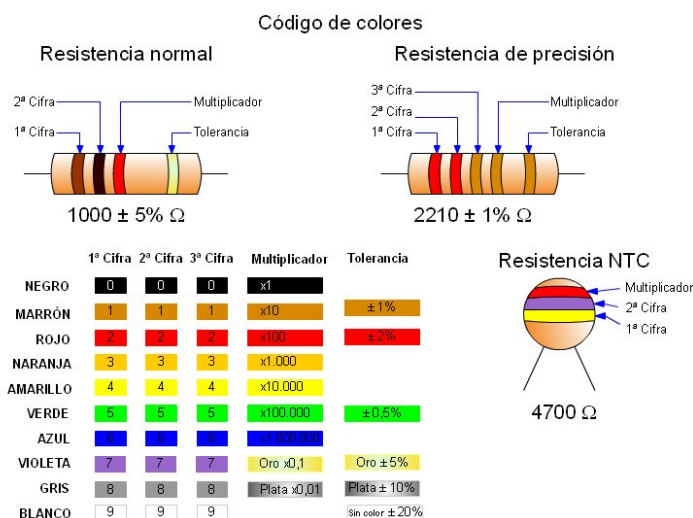


Ilustración 6. Código de Colores<sup>12</sup>

<sup>12</sup> [http://www.portaleso.com/usuarios/Toni/web\\_electronica\\_3/electronica\\_indice.html](http://www.portaleso.com/usuarios/Toni/web_electronica_3/electronica_indice.html)

## Imagen de referencia



**Ilustración 7.** Resistencia encapsulada axial<sup>13</sup>

### 4.6.4.6 Alimentación VAC

#### Descripción

O también denomina corriente alterna, es la corriente eléctrica en la que la magnitud y el sentido varían cíclicamente. La forma de oscilación de la corriente alterna más comúnmente utilizada es la de una oscilación sinusoidal, puesto que se consigue una transmisión más eficiente de la energía. Sin embargo, en ciertas aplicaciones se utilizan otras formas de oscilación periódicas, tales como la triangular o la cuadrada.

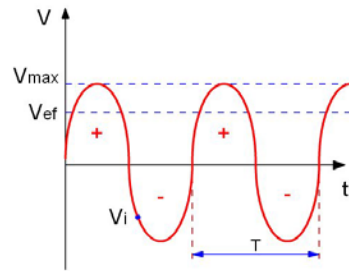
Utilizada genéricamente, la CA se refiere a la forma en la cual la electricidad llega a los hogares a las empresas. Sin embargo, las señales de audio y de radio transmitidas por los cables eléctricos, son también ejemplos de corriente alterna.

---

<sup>13</sup> [http://www.portaleso.com/usuarios/Toni/web\\_electronica\\_3/electronica\\_indice.html](http://www.portaleso.com/usuarios/Toni/web_electronica_3/electronica_indice.html)



## Figura de referencia



**Ilustración 8.** Forma de onda de la corriente alterna<sup>14</sup>

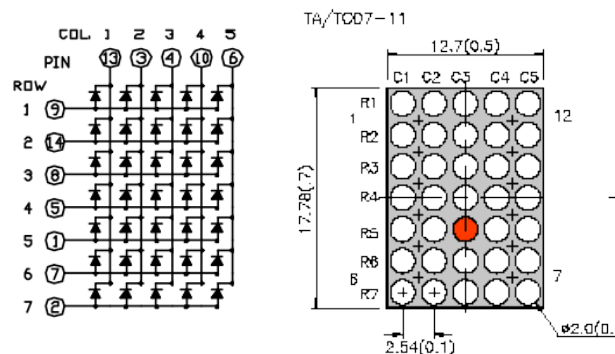
### 4.6.5 Matriz de diodos led

#### Descripción

Una matriz de LEDs consiste en un arreglo de LEDs que pueden ser encendidos y apagados individualmente desde un microcontrolador. Se puede pensar como una pantalla de pocos pixeles en los cuales pueden presentar gráficos y textos, tanto estáticos como en movimiento.

#### Conexión

El siguiente gráfico presenta una matriz de 5x7 LEDs, donde las columnas corresponden a los cátodos (deben ir conectados a tierra) y las filas corresponden a los ánodos (deben ir conectados a voltaje).



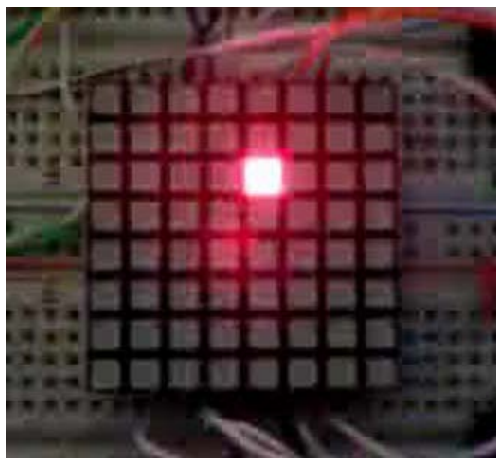
**Ilustración 9.** Matriz de 5x7 con columnas de cátodos<sup>15</sup>

<sup>14</sup> <http://lafisicaparatodos.wikispaces.com/Corriente+Alterna>

<sup>15</sup> <http://galaxi0.wordpress.com/about/salidas-y-entradas-digitales/matriz-de-leds/>

Para encender un LED específico debe ubicarse la intersección entre la columna y la línea correspondiente. Por ejemplo, para encender el LED que se muestra en la figura siguiente, la columna 3 (C3) debe ser conectada a tierra (a través de una resistencia de 220 ohm) mientras que la fila 5 (R5) es conectada a voltaje (5v).

### Ejemplo



**Ilustración 10.** Matriz de led

Realizado por: Iván Gallardo

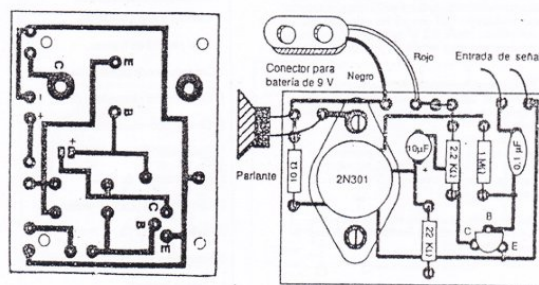
#### 4.6.6 Etapa de potencia (Amplificador de sonido)

##### Descripción

La etapa de potencia aumenta el nivel de una señal, incrementando, para ello, la amplitud de la señal de entrada mediante corrientes de polarización (voltaje negativo, voltaje positivo) en el transistor de salida.

El amplificador trabaja, internamente, con corriente continua; en caso de ser alimentado con la tensión entregada por la red domiciliaria se necesita un transformador y rectificador para adaptar el nivel de voltaje y tipo de corriente a los valores necesarios para el buen funcionamiento del equipo.

## Figura de referencia



**Ilustración 11.** Amplificador de sonido

Realizado por: Iván Gallardo

### 4.6.6.2 Altavoz

#### Descripción

Un altavoz es un transductor electroacústico utilizado para la reproducción de sonido. Uno o varios altavoces pueden formar una pantalla acústica.

La transducción sigue un doble procedimiento: eléctrico-mecánico-acústico. En la primera etapa convierte las ondas eléctricas en energía mecánica, y en la segunda convierte la energía mecánica en ondas de frecuencia acústica. Es por tanto la puerta por donde sale el sonido al exterior desde los aparatos que posibilitaron su amplificación, su transmisión por medios telefónicos o radioeléctricos, o su tratamiento.

## Figura de referencia

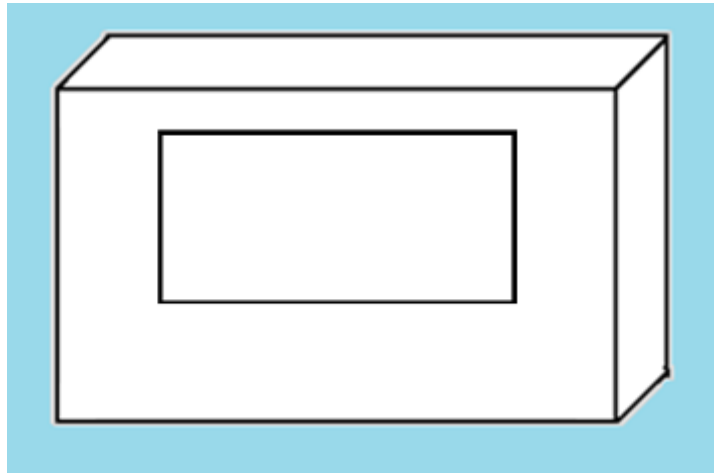


**Ilustración 12.** Altavoz

Realizado por: Iván Gallardo

#### 4.6.7 Diseño de la estructura

El diseño de la estructura del tablero electrónico sonoro, en su interior irán fijadas las placas necesarias para su funcionamiento.

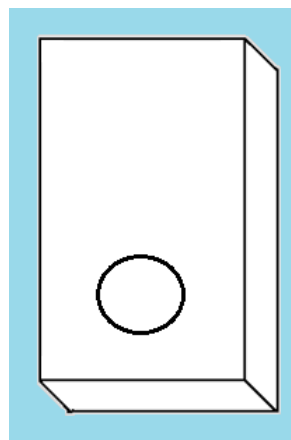


**Ilustración 13.** Estructura principal.

Realizado por: Iván Gallardo

##### 4.6.7.1 Diseño de las estructuras laterales

La estructura lateral está diseñada de manera que se pueda montar y asegurar el altavoz



**Ilustración 14.** Estructuras laterales.

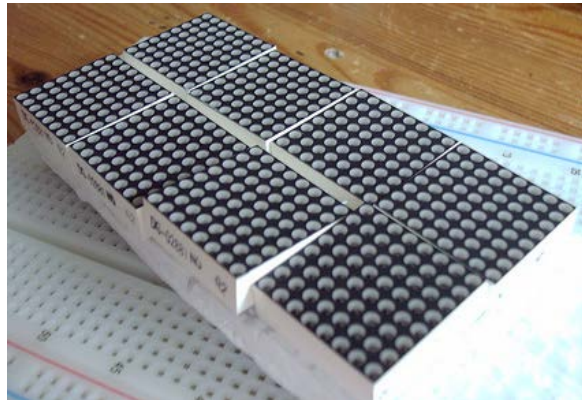
Realizado por: Iván Gallardo

## 4.7. IMPLEMENTACION DEL TABLERO ELECTRÓNICO SONORO

### 4.7.1 Introducción

En el presente capítulo se muestra el proceso de elaboración del tablero electrónico sonoro.

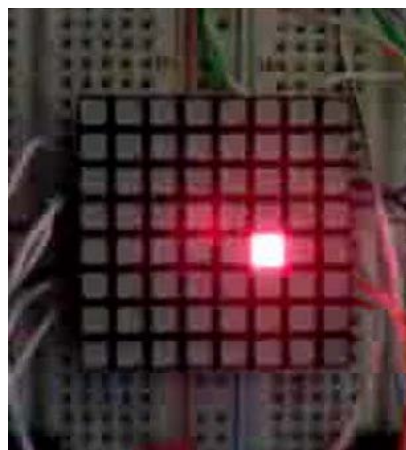
### 4.7.2 Matriz de diodos



**Ilustración 15.** Matrices de diodos led.

Realizado por: Iván Gallardo

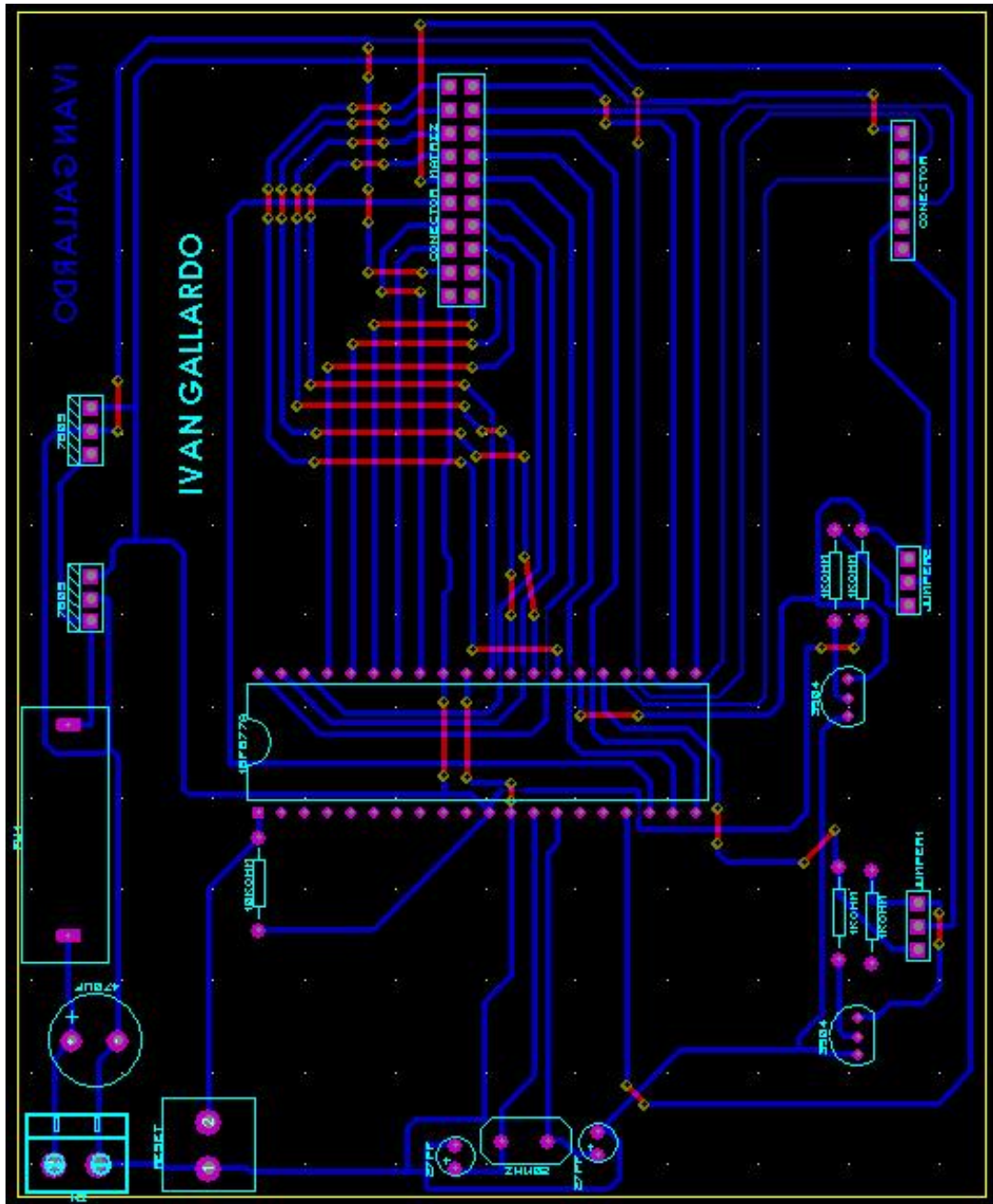
### 4.7.3 Ensamblaje en el protoboard de la matriz de diodos



**Ilustración 16.** Ensamblado en protoboard

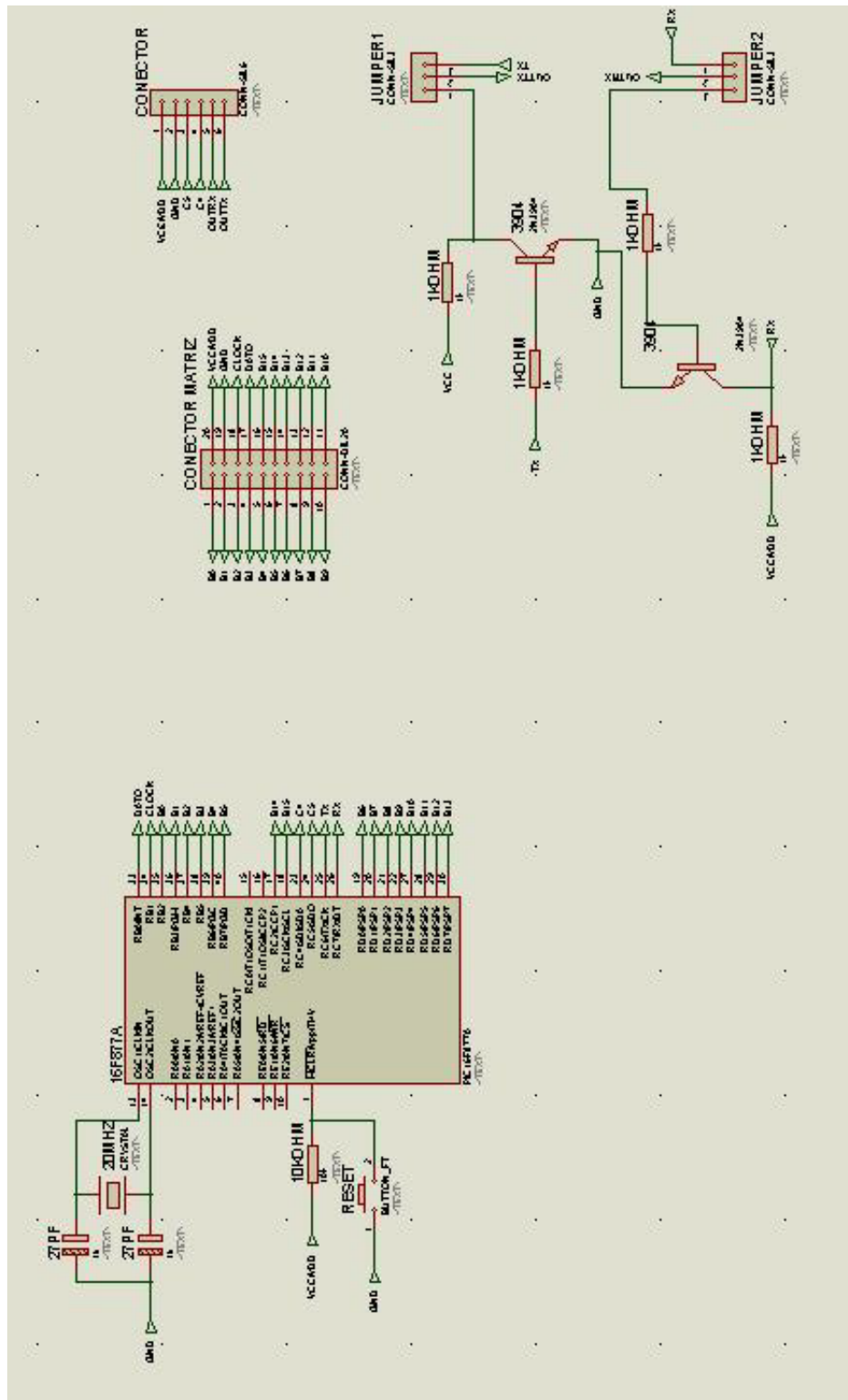
Realizado por: Iván Gallardo

Se puede observar el funcionamiento independiente de una matriz de diodos led.



**Ilustración 17.** Diagrama esquemático de la placa de control de la matriz de leds

Realizado por: Iván Gallardo



**Ilustración 18.** Captura de simulación de la placa de control de la matriz de leds

Realizado por: Iván Gallardo

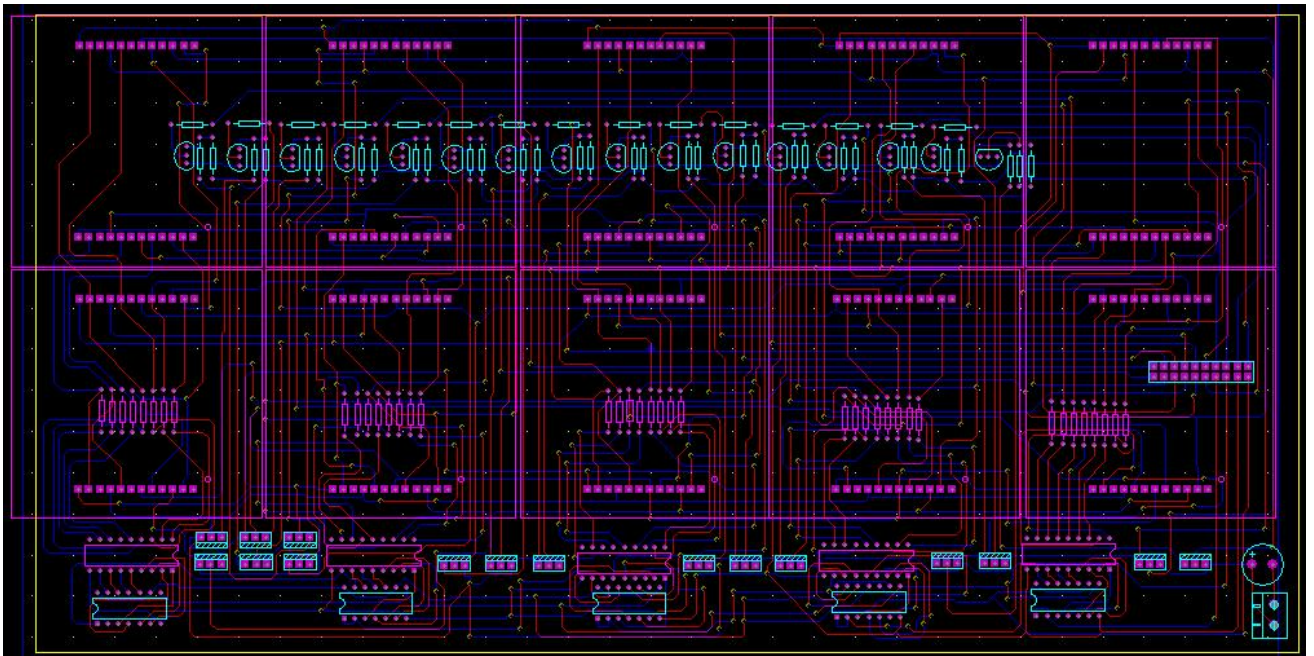




**Ilustración 19.** Placa de control de la matriz de leds

**Realizado por:** Iván Gallardo

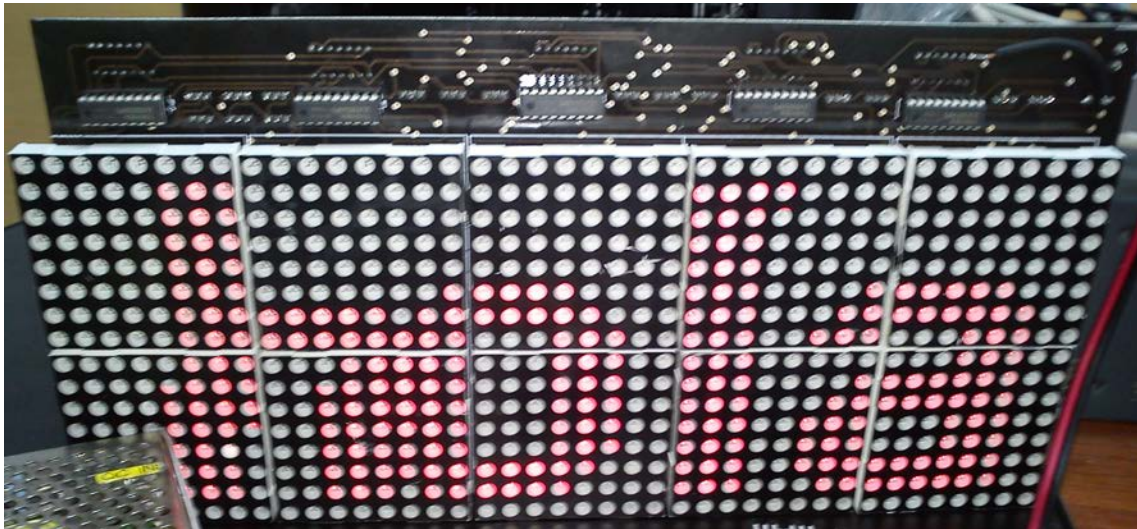
Esta placa es la encargada de recibir los caracteres uno a uno y mostrarlos en las matrices de leds.



**Ilustración 20.** Diagrama esquemático del conjunto de matrices de led

**Realizado por:** Iván Gallardo





**Ilustración 21.** Montaje de las matrices de diodos a la placa del tablero

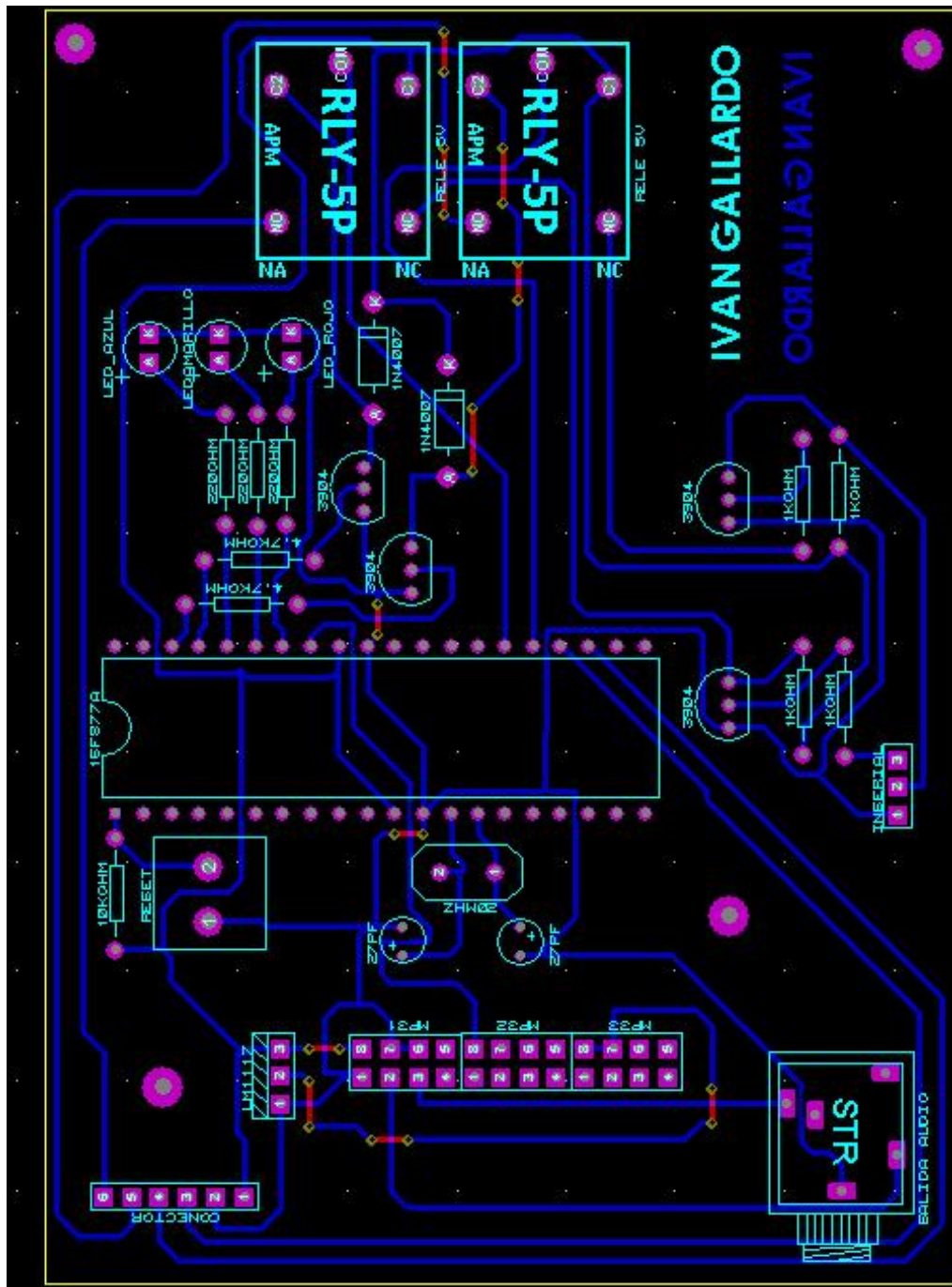
Realizado por: Iván Gallardo

#### 4.7.4 Módulo GSM



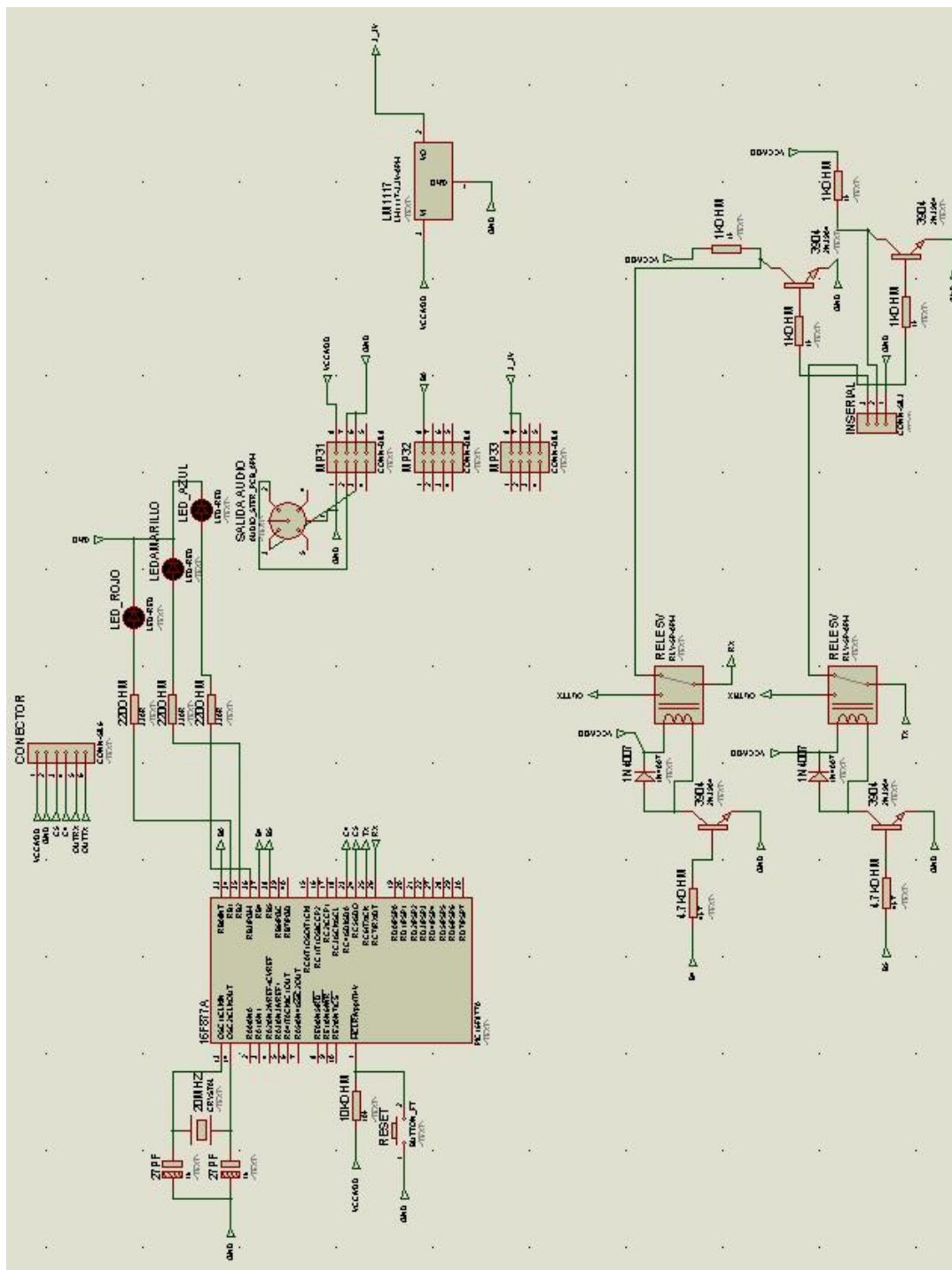
**Ilustración 22.** Módulo GSM

Realizado por: Iván Gallardo



**Ilustración 23.** Diagrama esquemático de la placa de transferencia del modem GSM a la matriz de led y a módulo mp3

Realizado por: Iván Gallardo



**Ilustración 24.** Captura de la simulación de la placa de transferencia del modem GSM a la matriz de led y a módulo mp3

Realizado por: Iván Gallardo





**Ilustración 25.** Placa de transferencia del modem GSM a matriz de led y a módulo mp3

**Realizado por:** Iván Gallardo



**Ilustración 26.** Montaje de placas a la estructura

**Realizado por:** Iván Gallardo



**Ilustración 27.** Vista frontal de la matriz led montada a la estructura principal

**Realizado por:** Iván Gallardo

## **4.8 PRUEBAS DEL TABLERO ELECTRÓNICO SONORO**

### **4.8.1 Prueba de activación del tablero**

Envío de clave para la activación de las opciones a mostrar en el tablero electrónico sonoro, para que el mensaje sea mostrado es necesario escribir el mensaje comenzando con el carácter “@” seguida de la clave que es “395” el siguiente número corresponde a la opción seleccionada, "2" 'Normal, Permite enviar cualquier tipo de mensaje, "6" muestra el 'MENSAJE 1, "7" muestra el 'MENSAJE 2, "8" muestra el 'MENSAJE 3 y "9" muestra el 'MENSAJE 4. Se da por terminado el mensaje con el carácter“\*” y el siguiente carácter es el “#” seguido de un número que indicara a los circuitos la canción a reproducir dentro de un grupo de canciones grabadas en una memoria SD, a excepción de los cuatro mensajes fijos que ya tienen incluido el sonido a reproducir, cerrando el mensaje con “\*#\*”, para el modo normal se cierra la cadena de mensaje con “\*”.

Como se muestra en el ejemplo



**Ilustración 28.** Envío clave de activación al tablero para la primera opción

**Realizado por:** Iván Gallardo

El tablero tiene pregrabada 4 primeras opciones, las cuales solo hay que agregarles la fecha.

### **Tablero electrónico sonoro**



**Ilustración 29.** Muestra mensaje en el tablero

**Realizado por:** Iván Gallardo

Recibe el mensaje extrae la información necesaria y lo muestra en el tablero, reproduciendo al mismo tiempo un archivo de audio previamente grabado en el tablero.

## **CAPÍTULO V**

### **ANÁLISIS DE COSTOS**

#### **5.1 Introducción**

En el presente capítulo se desarrolla un análisis de precios de la elaboración del Tablero Electrónico Sonoro, tomando en cuenta que los temas desarrollados a continuación son dentro de un marco referencial, más no un estudio detallado de un producto para la introducción al mercado.

Se fundamenta en la innovación y valor agregado que el creador ofrece a los campos de negocio en el mercado. Dentro del costo de la elaboración también se considerará un rubro adicional por ser un producto novedoso realizando un análisis de:

- Matriz FODA
- Análisis de costos
- Punto de equilibrio
- Costo, beneficio
- VAR
- TIR

#### **5.2 Matriz FODA**

El análisis FODA se realiza observando y describiendo las características del negocio de la organización y del mercado en el cual opera, el presente método nos permite detectar las Fortalezas de la organización, las Oportunidades del mercado, las Debilidades de la empresa y las Amenazas en el entorno.

También nos permite obtener datos de salida para conocer la situación real en que se encuentra la empresa, así como el riesgo y oportunidades que existen en el mercado y que afectan directamente al funcionamiento del negocio.

Componentes internos:

**Fortalezas:** Describen los recursos y las destrezas que ha adquirido la empresa, ¿En qué nos diferenciamos de la competencia?, ¿Qué sabemos hacer mejor?

**Debilidades:** Describen los factores en los cuales poseemos una posición desfavorable respecto a la competencia.

Componentes externos:

**Oportunidades:** Describen los posibles mercados, nichos de negocio, que están a la vista de todos, pero si no son reconocidas a tiempo se convierten en una desventaja competitiva.

**Amenazas:** Describen los factores que pueden poner en peligro la supervivencia de la organización, si dichas amenazas son reconocidas a tiempo pueden aludidas o convertidas en oportunidades.

Una vez descrito las amenazas, oportunidades, fortalezas y debilidades de la organización podemos construir la Matriz FODA, matriz que nos permite visualizar y resumir la situación actual.



<b>FORTALEZAS</b>	<b>OPORTUNIDADES</b>
<ul style="list-style-type: none"> <li>* Reducido consumo de energía.</li> <li>* Muestra anuncios de una forma entretenida y llamativa.</li> <li>* Sistema innovador.</li> <li>* Reducido consumo de papel</li> </ul>	<ul style="list-style-type: none"> <li>* Producto Ecuatoriano.</li> <li>* Sistema innovador.</li> <li>* Producto ecológico</li> </ul>
<b>DEBILIDADES</b>	<b>AMENAZAS</b>
<ul style="list-style-type: none"> <li>* El tablero no dispone de alimentación propia.</li> <li>* El tablero no envía una interface con el menú de opciones.</li> </ul>	<ul style="list-style-type: none"> <li>* Innovación del sistema.</li> </ul>

**Tabla 1.** Análisis FODA

### 5.3 Análisis de costos

El análisis presentado a continuación está basado en la comparación de costos de materiales y mano de obra, el costo de elementos adicionales utilizados, al igual el costo por hora empleada en la elaboración del Tablero.

#### 5.3.1 Materia Prima Directa (MPD)

Para el cálculo de los costos de la materia prima directa (MPD), se considera los materiales sujetos a la transformación identificables y cuantificables en el producto terminado detallados a continuación:

<b>Cantidad</b>	<b>Producto</b>	<b>C. Unitario</b>	<b>C. Total</b>
4	Resistencia 1K 1/4W	0.03	0.12
1	Condensador Electrolitico 470uF / 35V	0.35	0.35
1	Condensador Ceramico 27pF	0.22	0.22
3	Condensador Electrolitico 10uF / 16V	0.22	0.66
1	18F452	5.5	5.5
1	LM1117	0.5	0.5
1	L293D	0.8	0.8
1	ZOCALO 40P	0.9	0.9
1	ZOCALO 40P	0.25	0.25
9	1N4007	0.12	1.08
4	LED	0.1	0.4
1	74ls164	3	3
7	MOLEX 2P	0.25	1.75
1	Modulo GSM	90	90
1	METRO CABLE BUZ	1.4	1.4
1	PLACA DE COBRE	4	4
1	CONECTOR	1.1	1.1
8	Matriz 8x8	1	8
1	Modulo MP3	40	40
1	Fuente (Accesorios y elementos)	20	20
		<b>TOTAL</b>	<b>180.03</b>

**Tabla 2.** Materia Prima Directa

### 5.3.2 Costos indirectos de fabricación (CIF)

Dentro de los costos indirectos de fabricación (CIF) se recalca los suministros, materiales, servicios básicos y arriendo utilizados para la elaboración del producto terminado detallado a continuación:

<b>SUMINISTROS Y MATERIALES</b>			
<b>CANT.</b>	<b>DETALLE</b>	<b>V. UNIT</b>	<b>V. TOTAL</b>
1	Percloruro férrico	0.50	0.50
3	Pasta	0.20	0.60
1	Silicona en barra	0.15	0.15
1	Lija de agua	0.50	0.50
1	Tiras de Balsa	2.00	2.00
1	Caja de madera	10.00	10.00
<b>TOTAL MENSUAL</b>			<b>13.75</b>

**Tabla 3.** Suministros y materiales

<b>SERVICIOS BÁSICOS Y ARRIENDO</b>		
<b>CANT.</b>	<b>DETALLE</b>	<b>VALOR MENSUAL</b>
1	Servicios Básicos	20.00
1	Arriendo	150.00
<b>TOTAL</b>		<b>170.00</b>

**Tabla 4.** Servicios Básicos y Arriendo

Una vez revisado los suministros, materiales, servicios básicos y arriendo, los costos indirectos de fabricación (CIF) se demuestra a continuación:

<b>COSTOS INDIRECTOS DE FABRICACIÓN</b>			
<b>CANT.</b>	<b>DETALLE</b>	<b>V. UNIT</b>	<b>V. TOTAL</b>
1	Suministros y materiales	10.00	10.00
1	Servicios Básicos	20.00	20.00
1	Arriendo	150.00	150.00
<b>TOTAL MENSUAL</b>			<b>180.00</b>

**Tabla 5.** Costos indirectos de fabricación

### 5.3.3 Mano de obra directa

La mano de obra directa está determinada de acuerdo a las prestaciones de los técnicos quienes elaboran el producto terminado que se detalla a continuación:

<b>MANO DE OBRA DIRECTA</b>			
<b>CANT. HORAS</b>	<b>DETALLE</b>	<b>COSTO HORA</b>	<b>COSTO MENSUAL</b>
30	Técnico	10	300
<b>TOTAL</b>			<b>300</b>

**Tabla 6.** Mano de obra directa

Para una mejor demostración de los costos de elaboración del tablero electrónico sonoro activado por mensajes escritos, se demuestra el siguiente cuadro donde establece la materia prima directa, costos indirectos de fabricación y mano de obra directa con el 25% adicional para el precio de venta al público, siendo así un valor aceptable considerando las características del equipo.

HOJA DE COSTO DE PRODUCCIÓN			
<b>Artículo:</b>		<b>Producción:</b>	1 Tablero

MATERIA PRIMA DIRECTA				MANO DE OBRA DIRECTA			
Cant.	Producto	C. Unitario	C. Total	Cant. Horas	Detalle	Costo Hora	Costo mensual
4	Resistencia 1K 1/4W	0.03	0.12	30	Técnico	10.00	300.00
1	Condensador Electrolitico 470uF / 35V	0.35	0.35				
1	Condensador Ceramico 27pF	0.22	0.22				
3	Condensador Electrolitico 10uF / 16V	0.22	0.66				
1	18F452	5.50	5.50				
1	LM1117	0.50	0.50				
1	L293D	0.80	0.80				
1	ZOCALO 40P	0.90	0.90				
1	ZOCALO 40P	0.25	0.25				
9	1N4007	0.12	1.08				
4	LED	0.10	0.40				
1	74ls164	3.00	3.00				
7	MOLEX 2P	0.25	1.75				
1	Modulo GSM	90.00	90.00				
1	METRO CABLE BUZ	1.40	1.40				
1	PLACA DE COBRE	4.00	4.00				
1	CONECTOR	1.10	1.10				
8	Matriz 8x8	8.00	8.00				
1	Modulo MP3	40.00	40.00				

1	Fuente (Accesorios y elementos)	20.00	20.00			
<b>TOTAL MATERIA PRIMA DIRECTA</b>			<b>180.03</b>	<b>TOTAL MANO DE OBRA DIRECTA</b>		<b>300.00</b>
<b>COSTOS INDIRECTOS DE FABRICACIÓN</b>				<b>RESUMEN DE COSTOS</b>		
<b>Cant.</b>	<b>Detalle</b>	<b>V. Unit.</b>	<b>V. Total</b>	<b>CÁLCULO</b>	<b>DETALLE</b>	<b>COSTO TOTAL</b>
1	Suministros y materiales	10	10		Materia prima directa	<b>180.03</b>
1	Servicios Básicos	20.00	20	(+)	Costos indirectos de fabricación	<b>180.00</b>
1	Arriendo	150.00	150.00	(+)	Mano de obra directa	<b>300.00</b>
				(=)	Costo Total	<b>660.03</b>
				(/)	Unidades producidas	<b>1</b>
				(=)	Costo unitario de producción	<b>660.03</b>
				(+)	25% utilidad	<b>165.00</b>
<b>TOTAL COSTOS INDIRECTOS DE FABRICACIÓN</b>			<b>180.00</b>	(=)	PVP	<b>825.04</b>

**Tabla 7.** Costos de producción de una unidad

### 5.3.4 Ingresos

Para el cálculo de una proyección de ingresos a 5 meses siguientes, tomando en consideración que en cada año se aumentaría la venta de una unidad más, es necesario tomar en cuenta el PIB del Ecuador, pero, en este caso y para un mejor análisis se toma atención al PIB correspondiente a diciembre de 2011, demostrado así en la siguiente tabla:

INGRESOS							
COD	CONCEPTO	T0	T1	T2	T3	T4	T5
1.1	Cantidad		1	2	3	4	5
1.2	Precio Unitario de venta	825.04	825.04	866.87	910.82	956.99	1005.50
1.3	<b>INGRESO TOTAL</b>		825.04	1733.74	2732.46	3827.99	5027.55

**Tabla 8.** Cuadro de ingresos

### 5.3.5 Egresos

En la presente tabla se muestra los egresos producidos por el costo de producción con una proyección a 5 años, tomando en consideración de que cada mes se aumentaría una unidad vendida y determinando de que cada mes se aumentaría el 1.20% adicional al pago de horas para la mano de obra:

EGRESOS							
COD	CONCEPTO	T0	T1	T2	T3	T4	T5
	Cantidad		1	2	3	4	5
2	Costo de producción						
2.1	Materia prima						
2.1.1	Costo total Materia Prima	180.03	180.03	360.06	540.09	720.12	900.15
2.2	Mano de obra directa						
2.2.1	Técnico	300	300	600	900	1200	1500
2.3	Costos indirectos de fabricación						
2.3.1	Suministros y materiales	10	10	20	30	40	50
2.3.2	Servicios Básicos	20.00	20.00	40.00	60.00	80.00	100.00
2.3.3	Arriendo	150.00	150.00	150.00	150.00	150.00	150.00
2.4	<b>TOTAL COSTO DE PRODUCCIÓN</b>		<b>660.03</b>	<b>1170.06</b>	<b>1680.04</b>	<b>2190.12</b>	<b>2700.15</b>

**Tabla 9.** Costo de producción proyectado egresos

### 5.3.6 Precio de venta

Se considera que para obtener el precio de venta al público se tome en cuenta el 25% de utilidad observado en la hoja de producción y en resumen en la siguiente tabla:

DETALLE	VALOR
Costo unitario de producción	660.03
25% utilidad	165.00
<b>PVP (USD)</b>	<b>825.04</b>

**Tabla 10.** Precio de venta al público

De esta manera se logra un precio de venta al público aceptable en el mercado,

### 5.4 Análisis del punto de equilibrio

Para determinar el punto de equilibrio es necesario tomar en cuenta la materia prima directa (MPD), la mano de obra directa (MOD), los costos indirectos de fabricación (CIF) y el precio de venta al público (PVP), demostrado la clasificación de las cuentas en la siguiente tabla:

CLASIFICACIÓN DE CUENTAS		
CUENTA	TIPO	VALOR
Materia prima directa	CV	180.03
Mano de obra directa	CV	300.00
Suministros y materiales	CV	13.75
Arriendo	CF	150.00
Servicios Básicos	CV	20.00

**Tabla 11.** Clasificación de cuentas

#### 5.4.1 Punto de equilibrio

El punto de equilibrio permite determinar el volumen de ventas, es decir, en donde no obtiene ni beneficios ni pérdidas, para lo cual se da el enfoque de



acuerdo a las unidades monetarias y producidas detalladas a continuación:

Para el cálculo del punto de equilibrio se aplica la siguiente fórmula:

DATOS:	
CF=	150.00
CV=	513.78
PVP=	825.04
\$=	Punto de equilibrio en dólares

$$\$ = \frac{CF}{1 - \frac{CV}{PVP}}$$

Reemplazando la fórmula se obtiene:

\$=	$\frac{150.00}{1 - \frac{513.78}{825.04}}$	397.59686
-----	--	-----------

Obteniendo así el punto de equilibrio en dólares de: **\$397.60**

Para el cálculo del punto de equilibrio en cantidades a producir se aplica la siguiente fórmula:

DATOS:	
CF=	150.00
CV=	513.78
PVP=	825.04
\$=	Punto de equilibrio en unidades

$$x = \frac{CF}{PVP - CV}$$

Reemplazando la fórmula se obtiene:

x=	$\frac{150.00}{825.04 - 513.78}$	0.481912227
----	----------------------------------	-------------

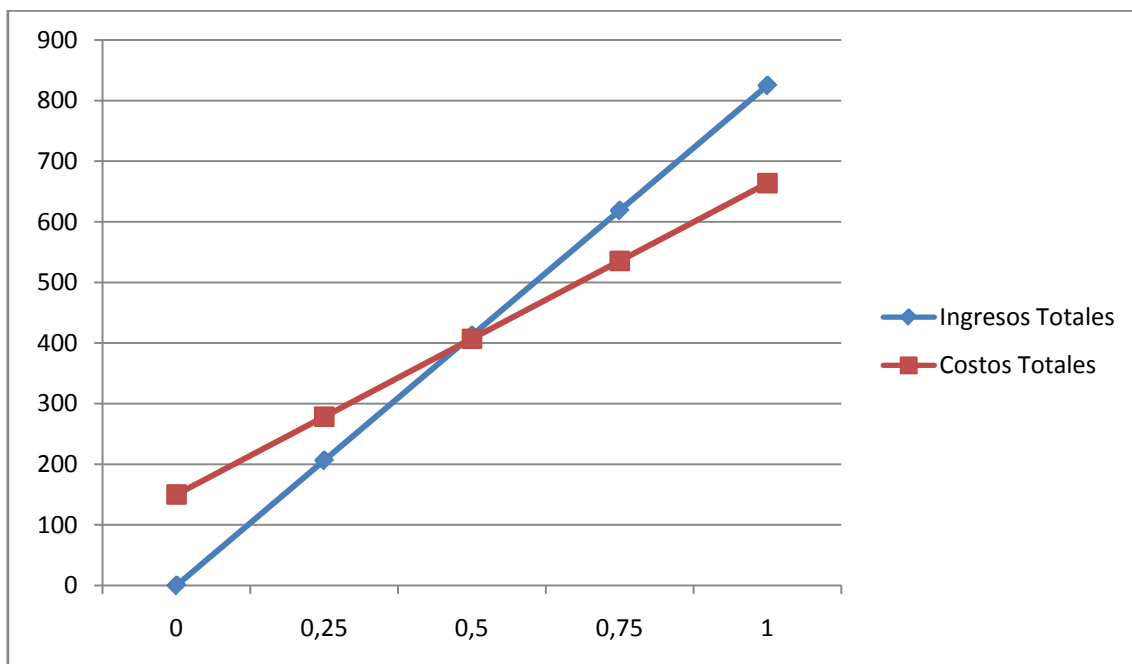
Obteniendo así el punto de equilibrio en cantidades de: **0.48**

Aunque 0.48 no se puede producir, es lógico que produciendo una unidad se está ganando con el tablero electrónico sonoro activado por mensajes escritos, es así que se demuestra la siguiente tabla y gráfico:

Proporción de unidad de prod.	Costo Variable	Costo Fijo	Costos Totales	Ingresos Totales
0		150.00	150.00	0.00
0.25	128.44	150.00	278.44	206.26
0.50	256.89	150.00	406.89	412.52
0.75	385.33	150.00	535.33	618.78
1.00	513.78	150.00	663.78	825.04

POR UNIDAD	
PRECIO	825.04
COSTO VARIABLE	513.78

**Tabla 12.** Punto de equilibrio



**Ilustración 30.** Punto de equilibrio

Para el punto de equilibrio se considera a los costos fijos, variables e ingresos del producto final, es ahí de donde se obtuvo el gráfico anterior, esto quiere decir que es rentable la producción del producto.

### 5.5 Análisis costo – beneficio

El presente análisis en el balance de la toma de decisiones de acuerdo al proyecto presentado, tomando en consideración los movimientos positivos (ingresos) y negativos (costos), demostrado en la siguiente tabla:

	<b>Flujo de costos</b>	<b>Flujo de Beneficios</b>		
1	660.03	825.04		
2	1170.06	1733.74	<b>COSTO BENEFICIO</b>	Flujo de costos / Flujo de Beneficios 14146.83 / 8400.4
3	1680.04	2732.45		
4	2190.12	3828.00		
5	2700.15	5027.60		
<b>TOTAL</b>	<b>8400.4</b>	<b>14146.83</b>		

**Tabla 13.** Costo – Beneficio

De acuerdo a la interpretación se verifica que en 0.60 de mes se habrá recuperado los costos del producto por los beneficios obtenidos, esto es inevitablemente un cálculo subjetivo.

### **5.6 Tasa mínima aceptable de retorno (TEMAR)**

Para el cálculo de la tasa mínima aceptable de retorno se toma en consideración la tasa de interés, el riesgo país y la inflación de acuerdo a datos proporcionados por el Banco central del Ecuador de diciembre 2011 y enero 2012.

<b>TASA MÍNIMA ACEPTABLE DE RETORNO</b>		
Tasa interés bancaria	16.30%	Banco Central del Ecuador
Riesgo País	5.07%	Banco Central del Ecuador
Inflación	5.29%	Banco Central del Ecuador
<b>TOTAL</b>	<b>26.66%</b>	Aprox. 27%

**Tabla 14.** Tasa mínima aceptable de retorno

<b>UTILIDAD NETA ESPERADA</b>		
(+)	Ingresos esperados al quimestre	14146.83
(-)	Costo total al quimestre	8400.4
(=)	Utilidad Operacional	5746.43
(-)	15% Participación trabajadores	861.9645
(-)	25% Impuesto a la renta	1436,6075
(=)	<b>Utilidad neta esperada</b>	<b>3447.858</b>

**Tabla 15.** Utilidad neta esperada

### 5.7 Valor actual neto (VAN)

A través del VAN permite determinar el valor presente de los flujos presentados del proyecto, representado en las siguientes tablas:

<b>VAN FINANCIERO</b>			
<b>TMAR</b>		<b>27%</b>	
<b>Tiempo</b>	<b>FCI</b>	<b>Factro Act.</b>	
0	-2692.40	1	-2692.40
1	825.04	0.78740	644.69
2	1733.74	0.62000	1074.92
3	2732.45	0.48819	1333.95
4	3828.00	0.38440	1471.48
5	5027.60	0.30268	1521.75
<b>VAN</b>			<b>3354.39</b>

**Tabla 16.** Análisis VAN

<b>TMAR</b>		<b>54%</b>	
<b>Tiempo</b>	<b>FCI</b>	<b>Factro Act.</b>	
0	-2692.40	1	-2692.40
1	825.04	0.64935	535.74
2	1733.74	0.42166	731.05
3	2732.45	0.27380	748.14
4	3828.00	0.17779	680.58
5	5027.60	0.11545	580.44
<b>VAN</b>			<b>-577.33</b>

**Tabla 17.** Análisis VAN

Las tablas anteriores nos representan que al ser el van positivo, el proyecto del tablero electrónico sonoro, es viable y rentable.

### 5.8 Tasa interna de retorno (TIR)

Sabiendo que la tasa interna de retorno (TIR) se utiliza para la aceptación o rechazo de un proyecto se presenta que:

<b>TIR FINANCIERO</b>		
	TASA	VALOR
VAN (+)	0.0027	3354.39
VAN (-)	0.0054	-577.33

Diferencia entre Tasas	0.0027	0.00267647
		0.0054
Diferencia entre VAN	3931.72	54%

**Tabla 18.** Tasa interna de retorno

## CAPÍTULO VI

### CONCLUSIONES Y RECOMENDACIONES

Los tableros electrónicos han adoptado diversas formas para acoplarse a las necesidades de los seres humanos, transformándose en una herramienta informativa, capaz de mostrar información actualizada y de forma entretenida, llamando la atención de los usuarios.

- Se consiguió analizar los contenidos teóricos, que ayudaron a la elaboración del tablero electrónico sonoro.
- Para determinar el diseño que más se adapte a las necesidades, fue necesario realizar varios borradores, hasta llegar al diseño final que cumpla satisfactoriamente con las funciones requeridas.
- Se logro diseñar un circuito que cumple con las funciones requeridas para el tablero electrónico.
- Luego de realizar algunas pruebas de funcionamiento, se llego a determinar que el tablero electrónico sonoro cumple a cabalidad con sus funciones, permitiendo implementar el tablero electrónico.
- Luego de haber seguido cada uno de los pasos para el desarrollo del proyecto, se consiguió cumplir con todos los objetivos propuestos para el tablero electrónico sonoro.
- Mediante el uso de mensajes escritos, telefonía celular, un modulo GSM y microcontroladores se logro desplegar anuncios en un tablero electrónico.
- Gracias al desarrollo de nuevas tecnologías como el modulo GSM es posible la simplificación del proyecto, ya que este es el encargado de procesar la

mayoría de la información entregándonos solo la información necesaria para las demás etapas del circuito.

- El ahorro de tiempo empleado para la publicación de eventos importantes, tales como inicio de matriculas, pago de pensiones, se reducirá considerablemente, pues con la ayuda del tablero electrónico sonoro activado por un mensaje de texto, permite actualizar la información casi al instante.
- Por la disminución de uso de anuncios impresos en papel, el presente proyecto significa una ayuda considerable para el medio ambiente, evitando el uso excesivo de papel, que a la final va a ser desechado.
- El aporte que se va a dar a la comunidad estudiantil con este proyecto, es incentivar el uso de la tecnología como herramienta para facilitar los procesos que se realizan en el día a día de todas las personas.
- Debido a la gran variedad de elementos existentes en el mercado, es necesario seleccionar con mucha presión los componentes a utilizar para la realización de cualquier proyecto, tomando en cuenta las condiciones tanto físicas como climáticas en el cual va a funcionar el proyecto.
- Es recomendable realizar varias pruebas con las cuales se puedan depurar una a una las inconsistencias existentes en cualquier proyecto.

## GLOSARIO

**Microcontrolador.** Circuito integrado de tamaño reducido que se emplea para el gobierno de uno o varios procesos, y periféricos.

**Datos digitales.** Datos expresados con valores numéricos discretos (dígitos binarios o bits) de unos y ceros que la máquina puede interpretar.

**RISC.** Computación de set de instrucciones reducidas.

**CISC.** Computación de set de instrucciones complejas.

**LED.** Diodo semiconductor que emite luz.

**LCD.** Pantalla de cristal líquido.

**CPU.** Unidad central de procesamiento.

**RAM.** Tipo de memoria de ordenador a la que se puede acceder aleatoriamente

**ROM.** Memoria sólo de lectura, memoria que se utiliza para almacenar los programas.

**PROM.** Memoria digital donde el valor de cada bit depende del estado de un fusible, que puede ser quemado una sola vez.

**EPROM.** Tipo de memoria ROM formadas por transistores de puerta flotante.

**A/D.** Analógico / Digital.

**Chip.** Pieza de silicio pequeña y con forma cuadrada o rectangular en cuyo interior hay un circuito integrado con millones de componentes; generalmente se combina con otros elementos para formar un sistema más complejo, como un ordenador.

**Arquitectura Harvard.** Hace referencia a las arquitecturas de computadoras que utilizaban dispositivos de almacenamiento físicamente separados para las instrucciones y para los datos.

**ALU.** Unidad aritmética lógica.

**USART.** Transmisor y Receptor Sincrónico/Asincrónico Universal.

**Ethernet.** Estándar de redes de área local para computadores con acceso al medio.



## BIBLIOGRAFÍA

1. BOYLESTAD, L. Electronica Teoria de circuitos. 6ª. Ed. México DF. Prentice Hall, 1997. Pp, 568-560
2. ANGULO, J.M.; ROMERO, S y ANGULO, I Microcontroladores PIC. 2da. Ed México DF. McGraw-Hill, 2003.pp.191-197.
3. REYES, C. Aprenda a programar Microcontroladores. Quito-Ecuador: Graficas Ayerve, 2004. Pp. 201

## BIBLIOGRAFÍA INTERNET

1. Telefonía móvil 3g

[http://es.wikipedia.org/wiki/Telefon%C3%ADa\\_m%C3%B3vil\\_3G](http://es.wikipedia.org/wiki/Telefon%C3%ADa_m%C3%B3vil_3G)  
(12-06-2012)

2. Comandos AT

[http://gbtcr.chileforge.cl/info\\_web/node106.html](http://gbtcr.chileforge.cl/info_web/node106.html)  
(15-06-2012)

3. Evolución de la Telefonía

[http://www.eveliux.com/mx/components/com\\_comment/joscomment/templates/default-emotop/css/css.css](http://www.eveliux.com/mx/components/com_comment/joscomment/templates/default-emotop/css/css.css)  
(10-06-2012)

4. Descripción del micontrolador PIC

<http://www.monografias.com/trabajos18/descripcion-pic/descripcion-pic.shtml>  
(17-09-2012)

5. Sonido

<http://www.monografias.com/trabajos5/also/also.shtml>  
(12-06-2012)

6. Microcontrolador

[http://catarina.udlap.mx/u\\_dl\\_a/tales/documentos/lem/escobar\\_a\\_g/apendiceB.pdf](http://catarina.udlap.mx/u_dl_a/tales/documentos/lem/escobar_a_g/apendiceB.pdf)

(16-06-2012)

7. Qué son y cómo funcionan los Comandos AT

[http://www.wikilearning.com/curso\\_gratis/que\\_son\\_y\\_como\\_funcionan\\_los\\_modems-como\\_usar\\_los\\_comandos\\_at/3477-18](http://www.wikilearning.com/curso_gratis/que_son_y_como_funcionan_los_modems-como_usar_los_comandos_at/3477-18)

(15-06-2012)

8. Microcontrolador y su funcionamiento

<http://www.monografias.com/trabajos12/microco/microco.shtml>

(17-06-2012)

9. Telefonía celular

<http://v6.yucatan.com.mx/especiales/celular/3g.asp>

(11-06-2012)

10. Modulo GPS

[http://www.dfrobot.com/wiki/index.php?title=GPS/GPRS/GSM\\_Module\\_V2%2KU:TEL0051%29](http://www.dfrobot.com/wiki/index.php?title=GPS/GPRS/GSM_Module_V2%2KU:TEL0051%29)

(18-08-2012)

11. Microcontrolador

<http://es.wikipedia.org/wiki/Microcontrolador>

(17-06-2012)

ANEXO 1

DM74LS164 8-Bit Serial In/Parallel Out Shift Register



August 1986  
Revised April 2000

## DM74LS164 8-Bit Serial In/Parallel Out Shift Register

### General Description

These 8-bit shift registers feature gated serial inputs and an asynchronous clear. A low logic level at either input inhibits entry of the new data, and resets the first flip-flop to the low level at the next clock pulse, thus providing complete control over incoming data. A high logic level on either input enables the other input, which will then determine the state of the first flip-flop. Data at the serial inputs may be changed while the clock is HIGH or LOW, but only information meeting the setup and hold time requirements will be entered. Clocking occurs on the LOW-to-HIGH level transition of the clock input. All inputs are diode-clamped to minimize transmission-line effects.

### Features

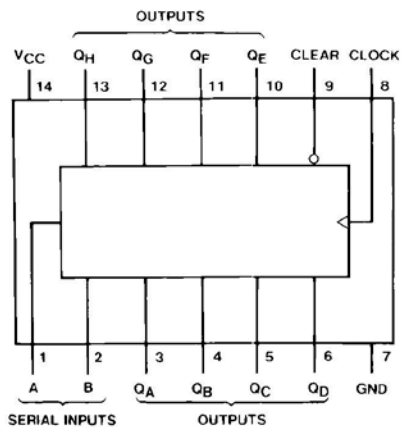
- Gated (enable/disable) serial inputs
- Fully buffered clock and serial inputs
- Asynchronous clear
- Typical clock frequency 36 MHz
- Typical power dissipation 80 mW

### Ordering Code:

Order Number	Package Number	Package Description
DM74LS164M	M14A	14-Lead Small Outline Integrated Circuit (SOIC), JEDEC MS-120, 0.150 Narrow
DM74LS164N	N14A	14-Lead Plastic Dual-In-Line Package (PDIP), JEDEC MS-001, 0.300 Wide

Devices also available in Tape and Reel. Specify by appending the suffix letter 'X' to the ordering code.

### Connection Diagram



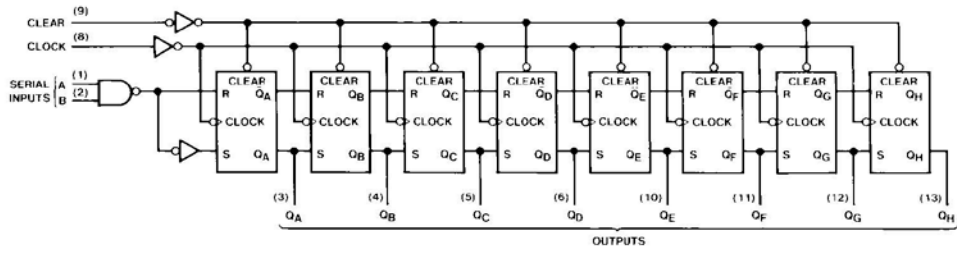
### Function Table

Inputs				Outputs			
Clear	Clock	A	B	Q <sub>A</sub>	Q <sub>B</sub>	...	Q <sub>H</sub>
L	X	X	X	L	L	...	L
H	L	X	X	Q <sub>A0</sub>	Q <sub>B0</sub>	...	Q <sub>H0</sub>
H	↑	H	H	H	Q <sub>An</sub>	...	Q <sub>Gn</sub>
H	↑	L	X	L	Q <sub>An</sub>	...	Q <sub>Gn</sub>
H	↑	X	L	L	Q <sub>An</sub>	...	Q <sub>Gn</sub>

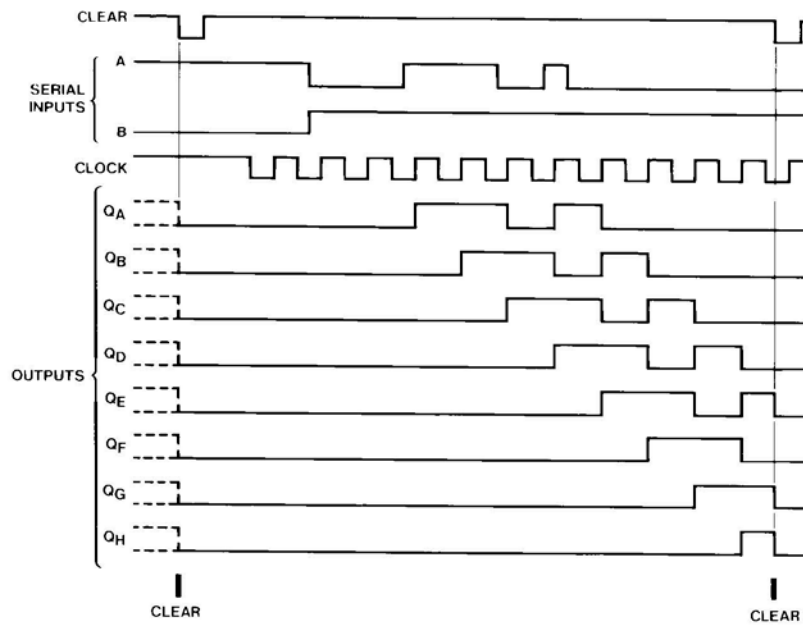
H = HIGH Level (steady state)  
L = LOW Level (steady state)  
X = Don't Care (any input, including transitions)  
↑ = Transition from LOW-to-HIGH level  
Q<sub>A0</sub>, Q<sub>B0</sub>, Q<sub>H0</sub> = The level of Q<sub>A</sub>, Q<sub>B</sub>, or Q<sub>H</sub>, respectively before the indicated steady-state input conditions were established.  
Q<sub>An</sub>, Q<sub>Gn</sub> = The level of Q<sub>A</sub> or Q<sub>G</sub> before the most recent ↑ transition of the clock; indicates a one-bit shift.

DM74LS164

Logic Diagram



Timing Diagram



### Absolute Maximum Ratings<sup>(Note 1)</sup>

Supply Voltage	7V
Input Voltage	7V
Operating Free Air Temperature Range	0°C to +70°C
Storage Temperature Range	-65°C to +150°C

**Note 1:** The "Absolute Maximum Ratings" are those values beyond which the safety of the device cannot be guaranteed. The device should not be operated at these limits. The parametric values defined in the Electrical Characteristics tables are not guaranteed at the absolute maximum ratings. The "Recommended Operating Conditions" tables will define the conditions for actual device operation.

### Recommended Operating Conditions

Symbol	Parameter	Min	Nom	Max	Units
V <sub>CC</sub>	Supply Voltage	4.75	5	5.25	V
V <sub>IH</sub>	HIGH Level Input Voltage	2			V
V <sub>IL</sub>	LOW Level Input Voltage			0.8	V
I <sub>OH</sub>	HIGH Level Output Current			-0.4	mA
I <sub>OL</sub>	LOW Level Output Current			8	mA
f <sub>CLK</sub>	Clock Frequency (Note 2)	0		25	MHz
t <sub>W</sub>	Pulse Width (Note 2)	Clock	20		ns
		Clear	20		
t <sub>SU</sub>	Data Setup Time (Note 2)	17			ns
t <sub>H</sub>	Data Hold Time (Note 2)	5			ns
t <sub>REL</sub>	Clear Release Time (Note 2)	30			ns
T <sub>A</sub>	Free Air Operating Temperature	0		70	°C

**Note 2:** T<sub>A</sub> = 25°C and V<sub>CC</sub> = 5V.

### Electrical Characteristics

over recommended operating free air temperature range (unless otherwise noted)

Symbol	Parameter	Conditions	Min	Typ (Note 3)	Max	Units
V <sub>I</sub>	Input Clamp Voltage	V <sub>CC</sub> = Min, I <sub>I</sub> = -18 mA			-1.5	V
V <sub>OH</sub>	HIGH Level Output Voltage	V <sub>CC</sub> = Min, I <sub>OH</sub> = Max	2.7	3.4		V
		V <sub>IL</sub> = Max, V <sub>IH</sub> = Min				
V <sub>OL</sub>	LOW Level Output Voltage	V <sub>CC</sub> = Min, I <sub>OL</sub> = Max		0.35	0.5	V
		V <sub>IL</sub> = Max, V <sub>IH</sub> = Min				
		I <sub>OL</sub> = 4 mA, V <sub>CC</sub> = Min		0.25	0.4	
I <sub>I</sub>	Input Current @ Max Input Voltage	V <sub>CC</sub> = Max, V <sub>I</sub> = 7V			0.1	mA
I <sub>IH</sub>	HIGH Level Input Current	V <sub>CC</sub> = Max, V <sub>I</sub> = 2.7V			20	μA
I <sub>IL</sub>	LOW Level Input Current	V <sub>CC</sub> = Max, V <sub>I</sub> = 0.4V			-0.4	mA
I <sub>OS</sub>	Short Circuit Output Current	V <sub>CC</sub> = Max (Note 4)	-20		-100	mA
I <sub>CC</sub>	Supply Current	V <sub>CC</sub> = Max (Note 5)		16	27	mA

**Note 3:** All typicals are at V<sub>CC</sub> = 5V, T<sub>A</sub> = 25°C.

**Note 4:** Not more than one output should be shorted at a time, and the duration should not exceed one second.

**Note 5:** I<sub>CC</sub> is measured with all outputs OPEN, the SERIAL input grounded, the CLOCK input at 2.4V, and a momentary ground, then 4.5V, applied to the CLEAR input.

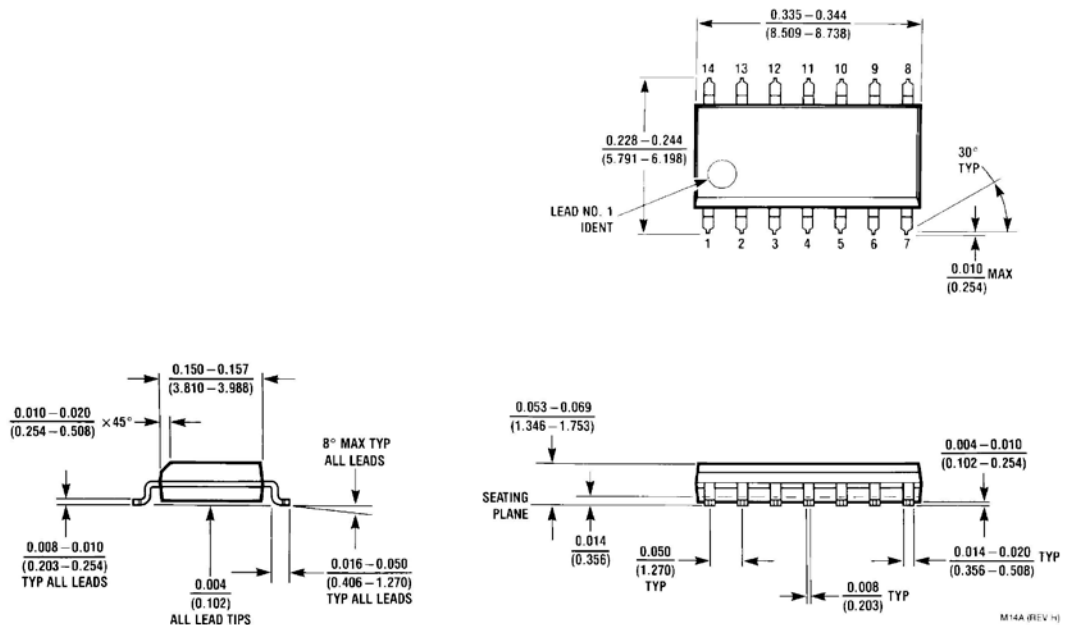
### Switching Characteristics

at V<sub>CC</sub> = 5V and T<sub>A</sub> = 25°C

Symbol	Parameter	From (Input) To (Output)	R <sub>L</sub> = 2 kΩ				Units
			C <sub>L</sub> = 15 pF		C <sub>L</sub> = 50 pF		
			Min	Max	Min	Max	
f <sub>MAX</sub>	Maximum Clock Frequency		25				MHz
t <sub>PLH</sub>	Propagation Delay Time LOW-to-HIGH Level Output	Clock to Output		27		30	ns
t <sub>PHL</sub>	Propagation Delay Time HIGH-to-LOW Level Output	Clock to Output		32		40	ns
t <sub>PHL</sub>	Propagation Delay Time HIGH-to-LOW Level Output	Clear to Output		36		45	ns

DM74LS164

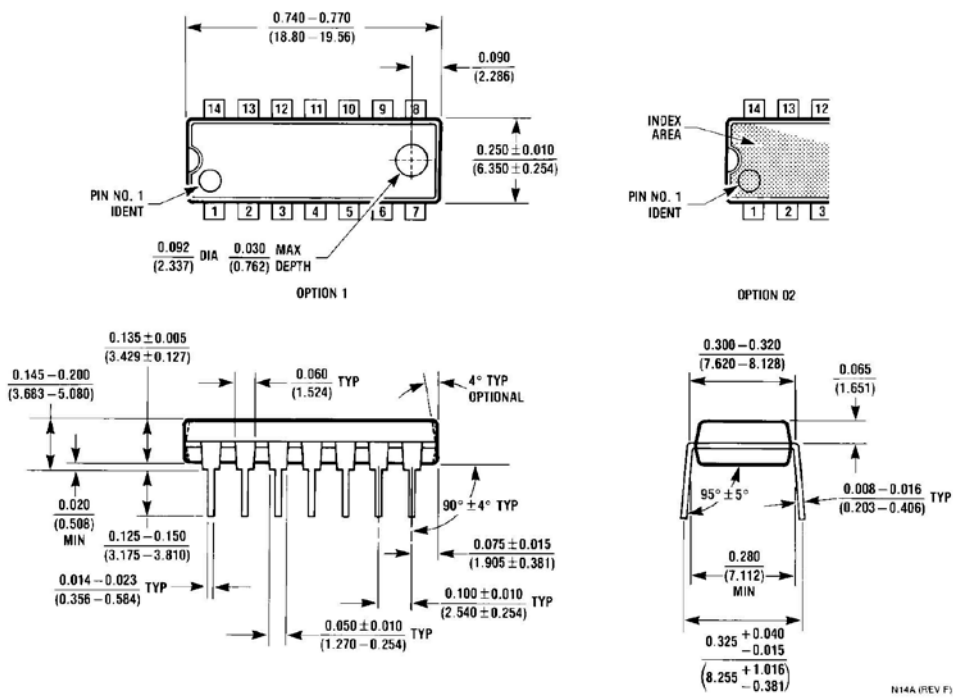
**Physical Dimensions** inches (millimeters) unless otherwise noted



**14-Lead Small Outline Integrated Circuit (SOIC), JEDEC MS-120, 0.150 Narrow  
Package Number M14A**

M14A (REV. H)

**Physical Dimensions** inches (millimeters) unless otherwise noted (Continued)



Fairchild does not assume any responsibility for use of any circuitry described, no circuit patent licenses are implied and Fairchild reserves the right at any time without notice to change said circuitry and specifications.

**LIFE SUPPORT POLICY**

FAIRCHILD'S PRODUCTS ARE NOT AUTHORIZED FOR USE AS CRITICAL COMPONENTS IN LIFE SUPPORT DEVICES OR SYSTEMS WITHOUT THE EXPRESS WRITTEN APPROVAL OF THE PRESIDENT OF FAIRCHILD SEMICONDUCTOR CORPORATION. As used herein:

1. Life support devices or systems are devices or systems which, (a) are intended for surgical implant into the body, or (b) support or sustain life, and (c) whose failure to perform when properly used in accordance with instructions for use provided in the labeling, can be reasonably expected to result in a significant injury to the user.
2. A critical component in any component of a life support device or system whose failure to perform can be reasonably expected to cause the failure of the life support device or system, or to affect its safety or effectiveness.

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

## ANEXO 2



# PIC18FXX2

## 28/40-pin High Performance, Enhanced FLASH Microcontrollers with 10-Bit A/D

### High Performance RISC CPU:

- C compiler optimized architecture/instruction set
  - Source code compatible with the PIC16 and PIC17 instruction sets
- Linear program memory addressing to 32 Kbytes
- Linear data memory addressing to 1.5 Kbytes

Device	On-Chip Program Memory		On-Chip RAM (bytes)	Data EEPROM (bytes)
	FLASH (bytes)	# Single Word Instructions		
PIC18F242	16K	8192	768	256
PIC18F252	32K	16384	1536	256
PIC18F442	16K	8192	768	256
PIC18F452	32K	16384	1536	256

- Up to 10 MIPs operation:
  - DC - 40 MHz osc./clock input
  - 4 MHz - 10 MHz osc./clock input with PLL active
- 16-bit wide instructions, 8-bit wide data path
- Priority levels for interrupts
- 8 x 8 Single Cycle Hardware Multiplier

### Peripheral Features:

- High current sink/source 25 mA/25 mA
- Three external interrupt pins
- Timer0 module: 8-bit/16-bit timer/counter with 8-bit programmable prescaler
- Timer1 module: 16-bit timer/counter
- Timer2 module: 8-bit timer/counter with 8-bit period register (time-base for PWM)
- Timer3 module: 16-bit timer/counter
- Secondary oscillator clock option - Timer1/Timer3
- Two Capture/Compare/PWM (CCP) modules. CCP pins that can be configured as:
  - Capture input: capture is 16-bit, max. resolution 6.25 ns (T<sub>CV</sub>/16)
  - Compare is 16-bit, max. resolution 100 ns (T<sub>CV</sub>)
  - PWM output: PWM resolution is 1- to 10-bit, max. PWM freq. @: 8-bit resolution = 156 kHz  
10-bit resolution = 39 kHz
- Master Synchronous Serial Port (MSSP) module, Two modes of operation:
  - 3-wire SPI™ (supports all 4 SPI modes)
  - I<sup>2</sup>C™ Master and Slave mode

### Peripheral Features (Continued):

- Addressable USART module:
  - Supports RS-485 and RS-232
- Parallel Slave Port (PSP) module

### Analog Features:

- Compatible 10-bit Analog-to-Digital Converter module (A/D) with:
  - Fast sampling rate
  - Conversion available during SLEEP
  - Linearity ≤ 1 LSB
- Programmable Low Voltage Detection (PLVD)
  - Supports interrupt on-Low Voltage Detection
- Programmable Brown-out Reset (BOR)

### Special Microcontroller Features:

- 100,000 erase/write cycle Enhanced FLASH program memory typical
- 1,000,000 erase/write cycle Data EEPROM memory
- FLASH/Data EEPROM Retention: > 40 years
- Self-reprogrammable under software control
- Power-on Reset (POR), Power-up Timer (PWRT) and Oscillator Start-up Timer (OST)
- Watchdog Timer (WDT) with its own On-Chip RC Oscillator for reliable operation
- Programmable code protection
- Power saving SLEEP mode
- Selectable oscillator options including:
  - 4X Phase Lock Loop (of primary oscillator)
  - Secondary Oscillator (32 kHz) clock input
- Single supply 5V In-Circuit Serial Programming™ (ICSP™) via two pins
- In-Circuit Debug (ICD) via two pins

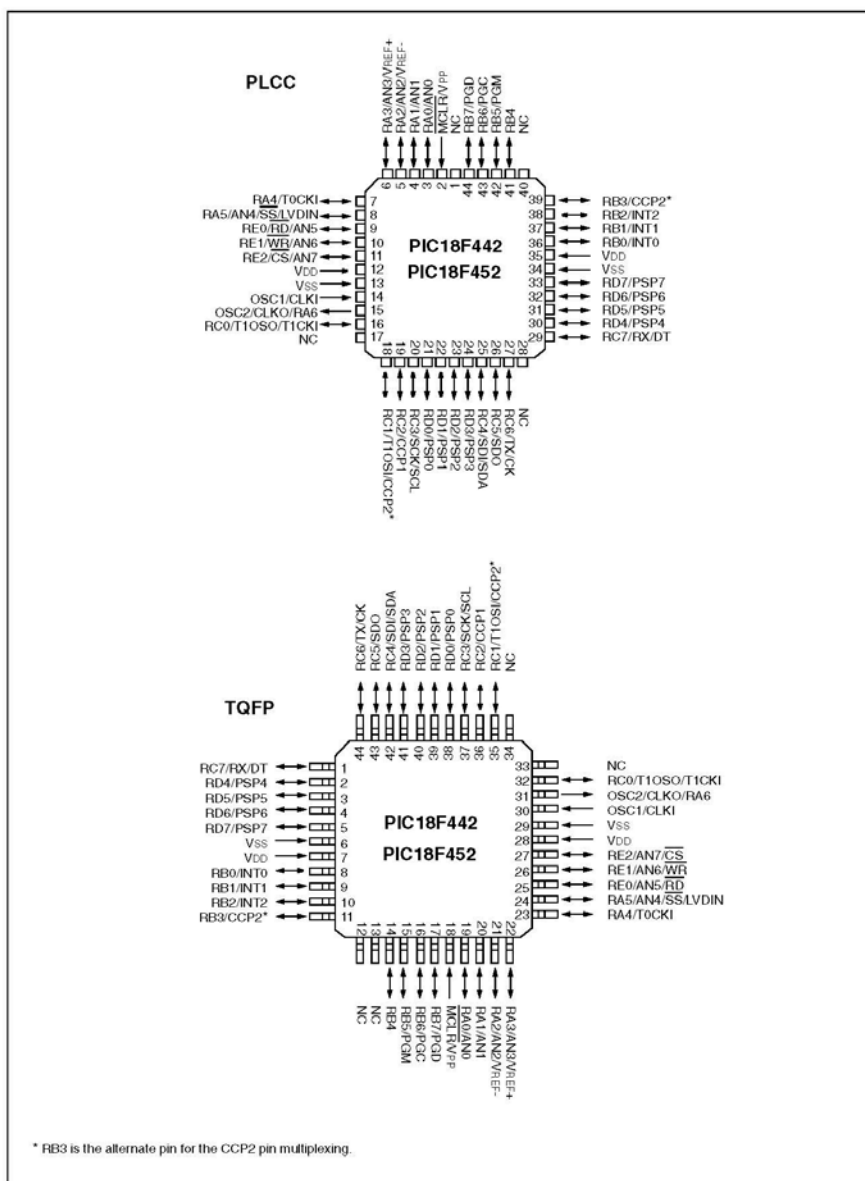
### CMOS Technology:

- Low power, high speed FLASH/EEPROM technology
- Fully static design
- Wide operating voltage range (2.0V to 5.5V)
- Industrial and Extended temperature ranges
- Low power consumption:
  - < 1.6 mA typical @ 5V, 4 MHz
  - 25 μA typical @ 3V, 32 kHz
  - < 0.2 μA typical standby current



# PIC18FXX2

## Pin Diagrams



## PIC18FXX2

### 1.0 DEVICE OVERVIEW

This document contains device specific information for the following devices:

- PIC18F242
- PIC18F252
- PIC18F442
- PIC18F452

These devices come in 28-pin and 40/44-pin packages. The 28-pin devices do not have a Parallel Slave Port (PSP) implemented and the number of Analog-to-Digital (A/D) converter input channels is reduced to 5. An overview of features is shown in Table 1-1.

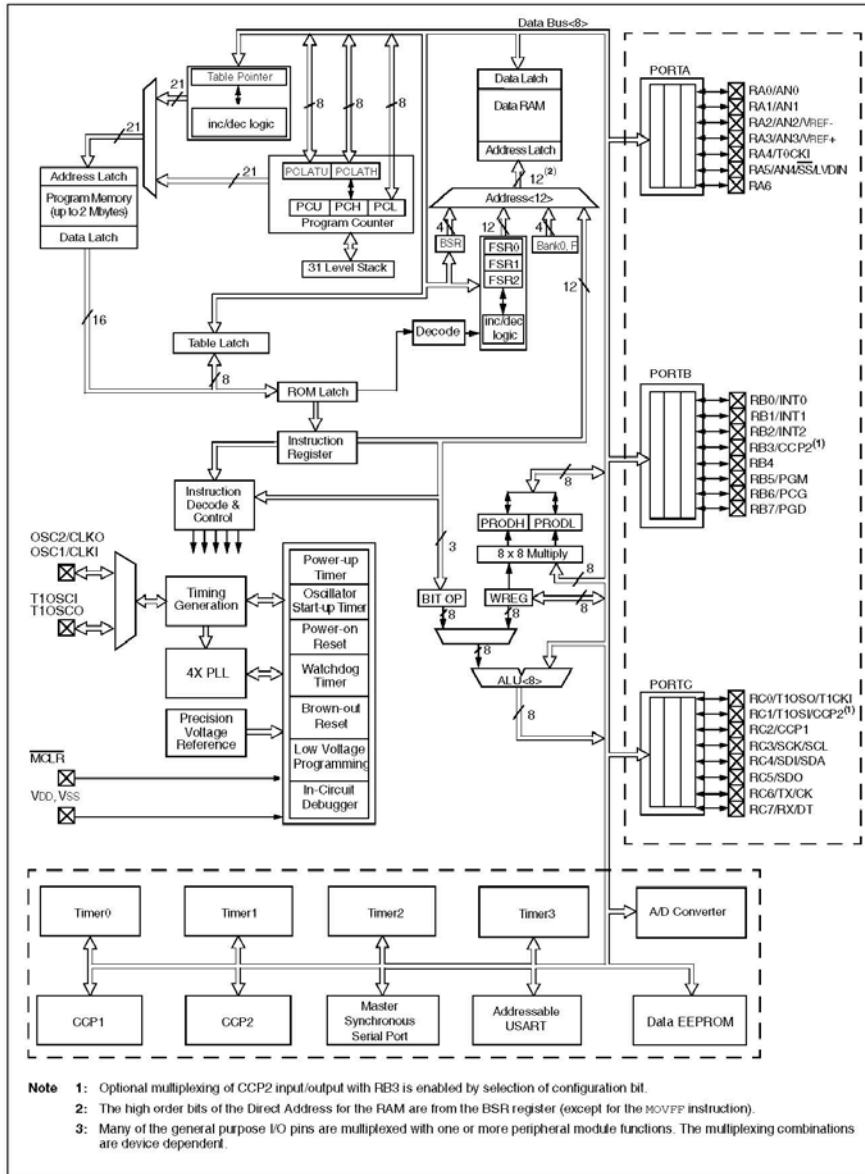
The following two figures are device block diagrams sorted by pin count: 28-pin for Figure 1-1 and 40/44-pin for Figure 1-2. The 28-pin and 40/44-pin pinouts are listed in Table 1-2 and Table 1-3, respectively.

TABLE 1-1: DEVICE FEATURES

Features	PIC18F242	PIC18F252	PIC18F442	PIC18F452
Operating Frequency	DC - 40 MHz	DC - 40 MHz	DC - 40 MHz	DC - 40 MHz
Program Memory (Bytes)	16K	32K	16K	32K
Program Memory (Instructions)	8192	16384	8192	16384
Data Memory (Bytes)	768	1536	768	1536
Data EEPROM Memory (Bytes)	256	256	256	256
Interrupt Sources	17	17	18	18
I/O Ports	Ports A, B, C	Ports A, B, C	Ports A, B, C, D, E	Ports A, B, C, D, E
Timers	4	4	4	4
Capture/Compare/PWM Modules	2	2	2	2
Serial Communications	MSSP, Addressable USART	MSSP, Addressable USART	MSSP, Addressable USART	MSSP, Addressable USART
Parallel Communications	—	—	PSP	PSP
10-bit Analog-to-Digital Module	5 input channels	5 input channels	8 input channels	8 input channels
RESETS (and Delays)	POR, BOR, RESET Instruction, Stack Full, Stack Underflow (PWRT, OST)	POR, BOR, RESET Instruction, Stack Full, Stack Underflow (PWRT, OST)	POR, BOR, RESET Instruction, Stack Full, Stack Underflow (PWRT, OST)	POR, BOR, RESET Instruction, Stack Full, Stack Underflow (PWRT, OST)
Programmable Low Voltage Detect	Yes	Yes	Yes	Yes
Programmable Brown-out Reset	Yes	Yes	Yes	Yes
Instruction Set	75 Instructions	75 Instructions	75 Instructions	75 Instructions
Packages	28-pin DIP 28-pin SOIC	28-pin DIP 28-pin SOIC	40-pin DIP 44-pin PLCC 44-pin TQFP	40-pin DIP 44-pin PLCC 44-pin TQFP

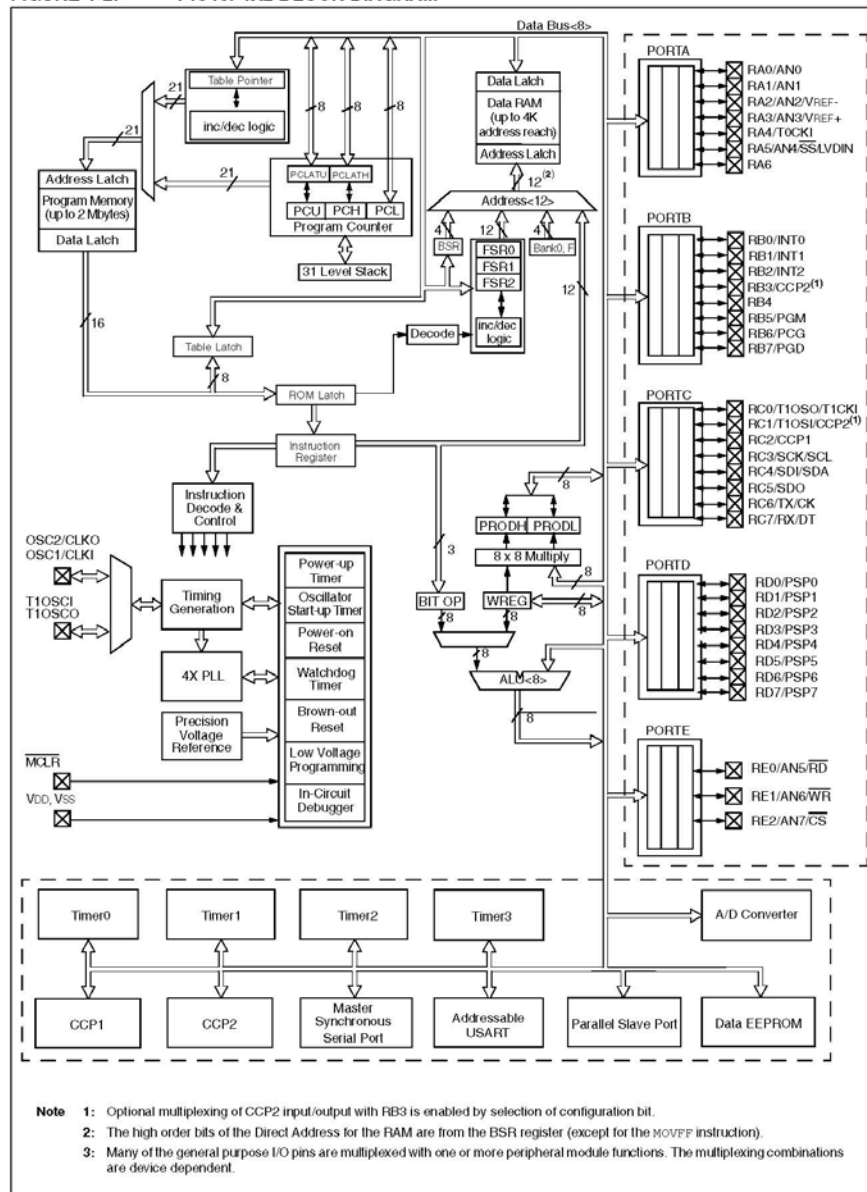
# PIC18FXX2

FIGURE 1-1: PIC18F2X2 BLOCK DIAGRAM



# PIC18FXX2

FIGURE 1-2: PIC18F4X2 BLOCK DIAGRAM



## PIC18FXX2

TABLE 1-2: PIC18F2X2 PINOUT I/O DESCRIPTIONS

Pin Name	Pin Number		Pin Type	Buffer Type	Description
	DIP	SOIC			
MCLR/VPP MCLR VPP	1	1	I I	ST ST	Master Clear (input) or high voltage ICSP programming enable pin. Master Clear (Reset) input. This pin is an active low RESET to the device. High voltage ICSP programming enable pin.
NC	—	—	—	—	These pins should be left unconnected.
OSC1/CLKI OSC1 CLKI	9	9	I I	ST CMOS	Oscillator crystal or external clock input. Oscillator crystal input or external clock source input. ST buffer when configured in RC mode, CMOS otherwise. External clock source input. Always associated with pin function OSC1. (See related OSC1/CLKI, OSC2/CLKO pins.)
OSC2/CLKO/RA6 OSC2 CLKO RA6	10	10	O O I/O	— — TTL	Oscillator crystal or clock output. Oscillator crystal output. Connects to crystal or resonator in Crystal Oscillator mode. In RC mode, OSC2 pin outputs CLKO which has 1/4 the frequency of OSC1, and denotes the instruction cycle rate. General Purpose I/O pin.
RA0/AN0 RA0 AN0 RA1/AN1 RA1 AN1 RA2/AN2/VREF- RA2 AN2 VREF- RA3/AN3/VREF+ RA3 AN3 VREF+ RA4/T0CKI RA4 T0CKI RA5/AN4/SS/LVDIN RA5 AN4 SS LVDIN RA6	2 3 4 5 6 7	2 3 4 5 6 7	I/O I I/O I I/O I I/O I I I I	TTL Analog TTL Analog TTL Analog Analog TTL Analog Analog ST/OD ST TTL Analog ST Analog	PORTA is a bi-directional I/O port. Digital I/O. Analog input 0. Digital I/O. Analog input 1. Digital I/O. Analog input 2. A/D Reference Voltage (Low) input. Digital I/O. Analog input 3. A/D Reference Voltage (High) input. Digital I/O. Open drain when configured as output. Timer0 external clock input. Digital I/O. Analog input 4. SPI Slave Select input. Low Voltage Detect Input. See the OSC2/CLKO/RA6 pin.

Legend: TTL = TTL compatible input  
ST = Schmitt Trigger input with CMOS levels  
O = Output  
OD = Open Drain (no P diode to VDD)

CMOS = CMOS compatible input or output  
I = Input  
P = Power

## ANEXO 3

### PROGRAMAS

#### Control de las Matrices

```

asm
    __CONFIG __CONFIG1H, _OSCS_OFF_1H & _HS_OSC_1H
    __CONFIG __CONFIG2L, _BOR_OFF_2L & _BORV_20_2L & _PWRT_OFF_2L
    __CONFIG __CONFIG2H, _WDT_OFF_2H & _WDTPS_128_2H
endasm

'DEFINE NO_CLRWDT 1

include "modedefs.bas"
DEFINE OSC 20

DEFINE I2C_SCLOUT 1
CPIN VAR portc.1
DPIN VAR portc.0

trisa=0
portb=0

clock var portb.1
dato var portb.0

b0 var portb.2
b1 var portb.3
b2 var portb.4

valor var byte
control var byte
'control=120
control=0
i var byte
i=0
a var byte
b var byte
b0=0
b1=0
b2=0
portb.5=0
portb.6=0

trisd=0
portd=0
trisc=%01110000
portc=0

b3 var portb.5
b4 var portb.6
b5 var portb.7
b6 var portd.0
b7 var portd.1
b8 var portd.2
b9 var portd.3
b10 var portd.4
b11 var portd.5

```

```

b12 var portd.6
b13 var portd.7
b14 var portc.2
b15 var portc.3
dao var word
'dao var byte
dao[0]=%1111111111111111

```

```
coje var bit
```

```

j var byte
h var byte
j=0
h=0

```

```

estado var byte
estado=0

```

```

'===== LETRAS =====0
fila1 var byte[16]
fila2 var byte[16]

```

```

'===== FIN LETRAS =====
'===== NUEVO =====

```

```

contador var byte
cada var byte
cada=0
aux var byte
aux1 var byte
aux2 var byte
aux3 var byte
aux4 var byte
auxs var byte[196]
conteo var byte
conteo=0

```

```

'===== FIN NUEVO=====
'===== LETRAS EJEMPLO ASCII =====
una var byte
letras var byte[196]

```

```

'===== LO NUEVO AQUI =====0
datouni var byte
datocen var byte
datodec var byte
numero var byte
temporal var byte
datos var byte

```

```

conteotiempo var byte
conteotiempo=0
horass var byte[6]
unidad var byte
decena var byte
momentanea var byte
segu var byte

```

```

dias var byte
diam var byte
mes var byte
anio var byte

```

```

hora var byte
minu var byte

```

```

posicion var byte
posiciontemp var byte

```

anterior var byte

horao var byte  
 minuo var byte  
 horao=\$1  
 minuo=\$40  
 validatiempo var byte  
 validatiempo=0

validatemp var byte  
 validatemp=0

analogo var word

DEFINE ADC\_BITS 10  
 DEFINE ADC\_CLOCK 3  
 DEFINE ADC\_SAMPLEUS 50

'=====

serout portc.6,T9600,[" "]  
 serout portc.6,T9600,[" "]  
 serout portc.6,T9600,[" "]

actualizado var bit  
 eeprom 0,[0]

read 0,actualizado

if actualizado=0 then  
 gosub primeralectura  
 gosub recolecciondatos  
 goto recupera  
else  
 serin portc.7,T9600,5000,recupera,datos  
 if datos="a" then  
 gosub primeralectura  
 gosub recolecciondatos  
 goto recupera  
 endif  
endif

'=====

'===== VARIACION DE TIEMPO

'===== FIN DE TIEMPO

'for conteo=0 to 70  
 ' auxs[conteo]=letras[conteo]  
 'next conteo

nueva var byte  
 nueva=4  
 reta var byte  
 reta=0  
 limite var byte  
 limite=0  
 espera var byte

ascii var byte  
 '===== FIN LETRAS EX =====



inicio:

```
for contador=0 to 4
  una=letras[contador]
  if portc.4=1 and portc.5=1 then
    pause 1000
    goto cambiadatos
  endif
  serout portc.6,T9600,[# una]
  pause 1000
  gosub comparador

  dao=fila1[0]
  gosub bucle
  gosub bucle1

  gosub limpia

  dao=fila1[1]
  gosub bucle
  gosub bucle2

  gosub limpia

  dao=fila1[2]
  gosub bucle
  gosub bucle3

  gosub limpia

  dao=fila1[3]
  gosub bucle
  gosub bucle4

  gosub limpia

  dao=fila1[4]
  gosub bucle
  gosub bucle5

  gosub limpia

  dao=fila1[5]
  gosub bucle
  gosub bucle6

  gosub limpia

  dao=fila1[6]
  gosub bucle
  gosub bucle7

  gosub limpia

  dao=fila1[7]
  gosub bucle
  gosub bucle8
  gosub limpia

  dao=fila1[8]
  gosub bucle
  gosub bucle9

  gosub limpia

  dao=fila1[9]
```

```

gosub bucle
gosub bucle10

gosub limpia

dao=fila1[10]
gosub bucle
gosub bucle11

gosub limpia

dao=fila1[11]
gosub bucle
gosub bucle12

gosub limpia

dao=fila1[12]
gosub bucle
gosub bucle13

gosub limpia

dao=fila1[13]
gosub bucle
gosub bucle14

gosub limpia

dao=fila1[14]
gosub bucle
gosub bucle15

gosub limpia

dao=fila1[15]
gosub bucle
gosub bucle16

gosub limpia
j=j+8 'Si aumento se separan las letras
next contador

J=0

pause 5 'Cambio tono de luz

control=control+1
if control>=45 then

if nueva>=numero-1 then
if control>=80 then
nueva=4
for conteo=0 to 10
letras[conteo]=auxs[conteo]
next conteo
if validatiempo=1 then
i2cread DPIN,CPIN,%11010000,0,[segu]
i2cread DPIN,CPIN,%11010000,1,[minu]
i2cread DPIN,CPIN,%11010000,2,[hora]
i2cread DPIN,CPIN,%11010000,3,[dias]
i2cread DPIN,CPIN,%11010000,4,[diam]
i2cread DPIN,CPIN,%11010000,5,[mes]
i2cread DPIN,CPIN,%11010000,6,[anio]

gosub cambiarhora
gosub cambiarminu

```

```

    unidad=a/10
    decena=a mod 10
    momentanea=unidad
    asci=momentanea
    gosub cambiaascci
    letras[posicion]=asci

    momentanea=decena
    asci=momentanea
    gosub cambiaascci
    letras[posicion+1]=asci

    unidad=b/10
    decena=b mod 10
    momentanea=unidad
    asci=momentanea
    gosub cambiaascci
    letras[posicion+3]=asci

    momentanea=decena
    asci=momentanea
    gosub cambiaascci
    letras[posicion+4]=asci
    letras[posicion+2]=58
endif
if validatep=1 then
    trisa=%00000001
    adcon0=%11000001
    adcon1=%00001110
    ADCIN 0, analogo
    pause 300
    analogo = analogo /128
    unidad=analogo/10
    decena=analogo mod 10
    momentanea=unidad
    asci=momentanea
    gosub cambiaascci
    letras[posiciontemp]=asci
    momentanea=decena
    asci=momentanea
    gosub cambiaascci
    letras[posiciontemp+1]=asci
endif

    control=0
endif
else
    control=37
    nueva=nueva+1
    aux=LETRAS[nueva]
    aux1=letras[1]
    aux2=letras[2]
    aux3=letras[3]
    aux4=letras[4]
    letras[0]=aux1
    letras[1]=aux2
    letras[2]=aux3
    letras[3]=aux4
    letras[4]=aux
    pauseUS 500
endif

endif

goto inicio

```

end

letraa:

```
fila1[0]=%00000000
fila1[1]=%11000011
fila1[2]=%11000011
fila1[3]=%11000011
fila1[4]=%11000011
fila1[5]=%11000011
fila1[6]=%11111111
fila1[7]=%11111111
fila1[8]=%11000011
fila1[9]=%11000011
fila1[10]=%11000011
fila1[11]=%11000011
fila1[12]=%11000011
fila1[13]=%01111110
fila1[14]=%00111100
fila1[15]=%00000000
```

return

letran:

```
fila1[0]=%00000000
fila1[1]=%11000011
fila1[2]=%11000011
fila1[3]=%11000011
fila1[4]=%11000011
fila1[5]=%11000011
fila1[6]=%11000111
fila1[7]=%11001111
fila1[8]=%11011011
fila1[9]=%11110011
fila1[10]=%11100011
fila1[11]=%11000011
fila1[12]=%11000011
fila1[13]=%11000011
fila1[14]=%11000011
fila1[15]=%00000000
```

return

letrag:

```
fila1[0]=%00000000
fila1[1]=%01111100
fila1[2]=%11111110
fila1[3]=%11000011
fila1[4]=%11000011
fila1[5]=%11000011
fila1[6]=%11011110
fila1[7]=%11011100
fila1[8]=%11000000
fila1[9]=%11000000
fila1[10]=%11000000
fila1[11]=%11000000
fila1[12]=%11000011
fila1[13]=%01111110
fila1[14]=%00111100
fila1[15]=%00000000
```

return

letray:

```
fila1[0]=%00000000
fila1[1]=%00011000
fila1[2]=%00011000
fila1[3]=%00011000
fila1[4]=%00011000
fila1[5]=%00011000
fila1[6]=%00011000
fila1[7]=%00011000
```

```

fila1[8]=%00011000
fila1[9]=%00011000
fila1[10]=%00111100
fila1[11]=%01000110
fila1[12]=%11000011
fila1[13]=%11000011
fila1[14]=%11000011
fila1[15]=%00000000
return

```

letraespacio:

```

fila1[0]=%00000000
fila1[1]=%00000000
fila1[2]=%00000000
fila1[3]=%00000000
fila1[4]=%00000000
fila1[5]=%00000000
fila1[6]=%00000000
fila1[7]=%00000000
fila1[8]=%00000000
fila1[9]=%00000000
fila1[10]=%00000000
fila1[11]=%00000000
fila1[12]=%00000000
fila1[13]=%00000000
fila1[14]=%00000000
fila1[15]=%00000000

```

return

letraw:

```

fila1[0]=%00000000
fila1[1]=%00100100
fila1[2]=%01111110
fila1[3]=%11011011
fila1[4]=%11011011
fila1[5]=%11011011
fila1[6]=%11011011
fila1[7]=%11000011
fila1[8]=%11000011
fila1[9]=%11000011
fila1[10]=%11000011
fila1[11]=%11000011
fila1[12]=%11000011
fila1[13]=%11000011
fila1[14]=%11000011
fila1[15]=%00000000

```

return

letrax:

```

fila1[0]=%00000000
fila1[1]=%11000011
fila1[2]=%11000011
fila1[3]=%11000011
fila1[4]=%11000011
fila1[5]=%11000011
fila1[6]=%01100110
fila1[7]=%00111100
fila1[8]=%00011000
fila1[9]=%00111100
fila1[10]=%01100110
fila1[11]=%11000011
fila1[12]=%11000011
fila1[13]=%11000011
fila1[14]=%11000011
fila1[15]=%00000000

```

return

letraz:

```
fila1[0]=%00000000
fila1[1]=%11111111
fila1[2]=%11111111
fila1[3]=%11100000
fila1[4]=%11100000
fila1[5]=%11100000
fila1[6]=%01110000
fila1[7]=%00111000
fila1[8]=%00011100
fila1[9]=%00001110
fila1[10]=%00000111
fila1[11]=%00000111
fila1[12]=%00000111
fila1[13]=%11111111
fila1[14]=%11111111
fila1[15]=%00000000
return
```

letraq:

```
fila1[0]=%00000000
fila1[1]=%00111101
fila1[2]=%01111110
fila1[3]=%11000111
fila1[4]=%11001011
fila1[5]=%11000011
fila1[6]=%11000011
fila1[7]=%11000011
fila1[8]=%11000011
fila1[9]=%11000011
fila1[10]=%11000011
fila1[11]=%11000011
fila1[12]=%11000011
fila1[13]=%01111110
fila1[14]=%00111100
fila1[15]=%00000000
return
```

letrass:

```
fila1[0]=%00000000
fila1[1]=%11111100
fila1[2]=%11111110
fila1[3]=%00000011
fila1[4]=%00000011
fila1[5]=%00000011
fila1[6]=%00000011
fila1[7]=%01111110
fila1[8]=%11111100
fila1[9]=%11000000
fila1[10]=%11000000
fila1[11]=%11000000
fila1[12]=%11000000
fila1[13]=%01111111
fila1[14]=%00111111
fila1[15]=%00000000
return
```

letrat:

```
fila1[0]=%00000000
fila1[1]=%00011000
fila1[2]=%00011000
fila1[3]=%00011000
fila1[4]=%00011000
fila1[5]=%00011000
fila1[6]=%00011000
fila1[7]=%00011000
```

```
fila1[8]=%00011000
fila1[9]=%00011000
fila1[10]=%00011000
fila1[11]=%00011000
fila1[12]=%00011000
fila1[13]=%11111111
fila1[14]=%11111111
fila1[15]=%00000000
return
```

letrau:

```
fila1[0]=%00000000
fila1[1]=%00111100
fila1[2]=%01111110
fila1[3]=%11000011
fila1[4]=%11000011
fila1[5]=%11000011
fila1[6]=%11000011
fila1[7]=%11000011
fila1[8]=%11000011
fila1[9]=%11000011
fila1[10]=%11000011
fila1[11]=%11000011
fila1[12]=%11000011
fila1[13]=%11000011
fila1[14]=%11000011
fila1[15]=%00000000
return
```

letrab:

```
fila1[0]=%00000000
fila1[1]=%11111100
fila1[2]=%11111110
fila1[3]=%11000011
fila1[4]=%11000011
fila1[5]=%11000011
fila1[6]=%11000110
fila1[7]=%11111100
fila1[8]=%11111100
fila1[9]=%11000110
fila1[10]=%11000011
fila1[11]=%11000011
fila1[12]=%11000011
fila1[13]=%11111110
fila1[14]=%11111100
fila1[15]=%00000000
```

return

letrac:

```
fila1[0]=%00000000
fila1[1]=%00111111
fila1[2]=%01111111
fila1[3]=%11100000
fila1[4]=%11000000
fila1[5]=%11000000
fila1[6]=%11000000
fila1[7]=%11000000
fila1[8]=%11000000
fila1[9]=%11000000
fila1[10]=%11000000
fila1[11]=%11000000
fila1[12]=%11100000
fila1[13]=%01111111
fila1[14]=%00111111
fila1[15]=%00000000
return
```

letrad:

```
    fila1[0]=%00000000
    fila1[1]=%11111100
    fila1[2]=%11111110
    fila1[3]=%11000011
    fila1[4]=%11000011
    fila1[5]=%11000011
    fila1[6]=%11000011
    fila1[7]=%11000011
    fila1[8]=%11000011
    fila1[9]=%11000011
    fila1[10]=%11000011
    fila1[11]=%11000011
    fila1[12]=%11000011
    fila1[13]=%11111110
    fila1[14]=%11111100
    fila1[15]=%00000000
return
```

letrae:

```
    fila1[0]=%00000000
    fila1[1]=%11111111
    fila1[2]=%11111111
    fila1[3]=%11000000
    fila1[4]=%11000000
    fila1[5]=%11000000
    fila1[6]=%11000000
    fila1[7]=%11111100
    fila1[8]=%11111100
    fila1[9]=%11000000
    fila1[10]=%11000000
    fila1[11]=%11000000
    fila1[12]=%11000000
    fila1[13]=%11111111
    fila1[14]=%11111111
    fila1[15]=%00000000
return
```

letraf:

```
    fila1[0]=%00000000
    fila1[1]=%11000000
    fila1[2]=%11000000
    fila1[3]=%11000000
    fila1[4]=%11000000
    fila1[5]=%11000000
    fila1[6]=%11000000
    fila1[7]=%111111000
    fila1[8]=%111111000
    fila1[9]=%11000000
    fila1[10]=%11000000
    fila1[11]=%11000000
    fila1[12]=%11000000
    fila1[13]=%11111111
    fila1[14]=%11111111
    fila1[15]=%00000000
return
```

letrah:

```
    fila1[0]=%00000000
    fila1[1]=%11000011
    fila1[2]=%11000011
    fila1[3]=%11000011
    fila1[4]=%11000011
    fila1[5]=%11000011
```



```

fila1[6]=%11000011
fila1[7]=%11111111
fila1[8]=%11111111
fila1[9]=%11000011
fila1[10]=%11000011
fila1[11]=%11000011
fila1[12]=%11000011
fila1[13]=%11000011
fila1[14]=%11000011
fila1[15]=%00000000
return

```

letrai:

```

fila1[0]=%00000000
fila1[1]=%11111111
fila1[2]=%11111111
fila1[3]=%00011000
fila1[4]=%00011000
fila1[5]=%00011000
fila1[6]=%00011000
fila1[7]=%00011000
fila1[8]=%00011000
fila1[9]=%00011000
fila1[10]=%00011000
fila1[11]=%00011000
fila1[12]=%00011000
fila1[13]=%11111111
fila1[14]=%11111111
fila1[15]=%00000000
return

```

letraj:

```

fila1[0]=%00000000
fila1[1]=%01111000
fila1[2]=%11111000
fila1[3]=%11011000
fila1[4]=%00011000
fila1[5]=%00011000
fila1[6]=%00011000
fila1[7]=%00011000
fila1[8]=%00011000
fila1[9]=%00011000
fila1[10]=%00011000
fila1[11]=%00011000
fila1[12]=%00011000
fila1[13]=%11111111
fila1[14]=%11111111
fila1[15]=%00000000
return

```

letrak:

```

fila1[0]=%00000000
fila1[1]=%11000011
fila1[2]=%11000011
fila1[3]=%11000110
fila1[4]=%11001100
fila1[5]=%11011000
fila1[6]=%11110000
fila1[7]=%11100000
fila1[8]=%11100000
fila1[9]=%11110000
fila1[10]=%11011000
fila1[11]=%11001100
fila1[12]=%11000110
fila1[13]=%11000011
fila1[14]=%11000011
fila1[15]=%00000000

```

return

letram:

```
fila1[0]=%00000000
fila1[1]=%11000011
fila1[2]=%11000011
fila1[3]=%11000011
fila1[4]=%11000011
fila1[5]=%11000011
fila1[6]=%11000011
fila1[7]=%11000011
fila1[8]=%11000011
fila1[9]=%11011011
fila1[10]=%11111111
fila1[11]=%11111111
fila1[12]=%11100111
fila1[13]=%11000011
fila1[14]=%11000011
fila1[15]=%00000000
return
```

letrao:

```
fila1[0]=%00000000
fila1[1]=%00111100
fila1[2]=%01111110
fila1[3]=%11000011
fila1[4]=%11000011
fila1[5]=%11000011
fila1[6]=%11000011
fila1[7]=%11000011
fila1[8]=%11000011
fila1[9]=%11000011
fila1[10]=%11000011
fila1[11]=%11000011
fila1[12]=%11000011
fila1[13]=%01111110
fila1[14]=%00111100
fila1[15]=%00000000
```

return

letrap:

```
fila1[0]=%00000000
fila1[1]=%11000000
fila1[2]=%11000000
fila1[3]=%11000000
fila1[4]=%11000000
fila1[5]=%11000000
fila1[6]=%11000000
fila1[7]=%11111100
fila1[8]=%11111110
fila1[9]=%11000011
fila1[10]=%11000011
fila1[11]=%11000011
fila1[12]=%11000011
fila1[13]=%11111110
fila1[14]=%11111100
fila1[15]=%00000000
return
```

letrar:

```
fila1[0]=%00000000
fila1[1]=%11000011
fila1[2]=%11000011
fila1[3]=%11000110
fila1[4]=%11001100
fila1[5]=%11011000
```

```
fila1[6]=%11110000
fila1[7]=%11100000
fila1[8]=%11111100
fila1[9]=%11111110
fila1[10]=%11000011
fila1[11]=%11000011
fila1[12]=%11000011
fila1[13]=%11111110
fila1[14]=%11111100
fila1[15]=%00000000
return
```

letral:

```
fila1[0]=%00000000
fila1[1]=%11111110
fila1[2]=%11111110
fila1[3]=%11000000
fila1[4]=%11000000
fila1[5]=%11000000
fila1[6]=%11000000
fila1[7]=%11000000
fila1[8]=%11000000
fila1[9]=%11000000
fila1[10]=%11000000
fila1[11]=%11000000
fila1[12]=%11000000
fila1[13]=%11000000
fila1[14]=%11000000
fila1[15]=%00000000
return
```

letrav:

```
fila1[0]=%00000000
fila1[1]=%00011000
fila1[2]=%00111100
fila1[3]=%01100110
fila1[4]=%11000011
fila1[5]=%11000011
fila1[6]=%11000011
fila1[7]=%11000011
fila1[8]=%11000011
fila1[9]=%11000011
fila1[10]=%11000011
fila1[11]=%11000011
fila1[12]=%11000011
fila1[13]=%11000011
fila1[14]=%11000011
fila1[15]=%00000000
return
```

uno:

```
fila1[0]=%00000000
fila1[1]=%01111110
fila1[2]=%01111110
fila1[3]=%00011000
fila1[4]=%00011000
fila1[5]=%00011000
fila1[6]=%00011000
fila1[7]=%00011000
fila1[8]=%00011000
fila1[9]=%00011000
fila1[10]=%10011000
fila1[11]=%11011000
fila1[12]=%01111000
fila1[13]=%00111000
fila1[14]=%00011000
fila1[15]=%00000000
```

return

dos:

```
fila1[0]=%00000000
fila1[1]=%11111111
fila1[2]=%11111111
fila1[3]=%11000000
fila1[4]=%01100000
fila1[5]=%00110000
fila1[6]=%00011000
fila1[7]=%00001100
fila1[8]=%00000110
fila1[9]=%00000011
fila1[10]=%00000011
fila1[11]=%11000011
fila1[12]=%11000011
fila1[13]=%01111110
fila1[14]=%00111100
fila1[15]=%00000000
return
```

tres:

```
fila1[0]=%00000000
fila1[1]=%00111100
fila1[2]=%01111110
fila1[3]=%11000011
fila1[4]=%00000011
fila1[5]=%00000011
fila1[6]=%00000110
fila1[7]=%00111100
fila1[8]=%00111100
fila1[9]=%00000110
fila1[10]=%00000010
fila1[11]=%00000011
fila1[12]=%11000011
fila1[13]=%01111110
fila1[14]=%00111100
fila1[15]=%00000000
```

return

cuatro:

```
fila1[0]=%00000000
fila1[1]=%00000011
fila1[2]=%00000011
fila1[3]=%00000011
fila1[4]=%00000011
fila1[5]=%00000011
fila1[6]=%00000011
fila1[7]=%11111111
fila1[8]=%11111111
fila1[9]=%11000011
fila1[10]=%11000011
fila1[11]=%11000011
fila1[12]=%11000011
fila1[13]=%11000011
fila1[14]=%11000011
fila1[15]=%00000000
return
```

cinco:

```
fila1[0]=%00000000
fila1[1]=%11111110
fila1[2]=%11111111
fila1[3]=%00000011
fila1[4]=%00000011
```

```

fila1[5]=%00000011
fila1[6]=%00000011
fila1[7]=%11111111
fila1[8]=%11111110
fila1[9]=%11000000
fila1[10]=%11000000
fila1[11]=%11000000
fila1[12]=%11000000
fila1[13]=%11111111
fila1[14]=%01111111
fila1[15]=%00000000
return

```

seis:

```

fila1[0]=%00000000
fila1[1]=%00111100
fila1[2]=%01111110
fila1[3]=%11000011
fila1[4]=%10000011
fila1[5]=%11000011
fila1[6]=%11111110
fila1[7]=%11111100
fila1[8]=%11000000
fila1[9]=%01100000
fila1[10]=%00110000
fila1[11]=%00011000
fila1[12]=%00001100
fila1[13]=%00000110
fila1[14]=%00000011
fila1[15]=%00000000
return

```

siete:

```

fila1[0]=%00000000
fila1[1]=%11000000
fila1[2]=%11000000
fila1[3]=%11000000
fila1[4]=%11000000
fila1[5]=%11000000
fila1[6]=%11000000
fila1[7]=%01100000
fila1[8]=%00110000
fila1[9]=%00011000
fila1[10]=%00001100
fila1[11]=%00000110
fila1[12]=%00000011
fila1[13]=%11111111
fila1[14]=%11111111
fila1[15]=%00000000
return

```

ocho:

```

fila1[0]=%00000000
fila1[1]=%00111100
fila1[2]=%01111110
fila1[3]=%11000011
fila1[4]=%11000011
fila1[5]=%11000011
fila1[6]=%01000010
fila1[7]=%00111100
fila1[8]=%00111100
fila1[9]=%01000010
fila1[10]=%11000011
fila1[11]=%11000011
fila1[12]=%11000011
fila1[13]=%01111110
fila1[14]=%00111100

```

```

fila1[15]=%00000000
return

```

nueve:

```

fila1[0]=%00000000
fila1[1]=%11111100
fila1[2]=%11111110
fila1[3]=%00000011
fila1[4]=%00000011
fila1[5]=%00000011
fila1[6]=%00000011
fila1[7]=%00111111
fila1[8]=%01111111
fila1[9]=%11000011
fila1[10]=%11000011
fila1[11]=%11000011
fila1[12]=%11000011
fila1[13]=%01111110
fila1[14]=%00111100
fila1[15]=%00000000
return

```

cero:

```

fila1[0]=%00000000
fila1[1]=%00111100
fila1[2]=%01111110
fila1[3]=%11000011
fila1[4]=%11000011
fila1[5]=%11000011
fila1[6]=%11000011
fila1[7]=%11000011
fila1[8]=%11000011
fila1[9]=%11000011
fila1[10]=%11000011
fila1[11]=%11000011
fila1[12]=%11000011
fila1[13]=%01111110
fila1[14]=%00111100
fila1[15]=%00000000
return

```

puntos:

```

    fila1[0]=%00000000
fila1[1]=%00000000
fila1[2]=%00000000
fila1[3]=%00011000
fila1[4]=%00111100
fila1[5]=%00111100
fila1[6]=%00011000
fila1[7]=%00000000
fila1[8]=%00000000
fila1[9]=%00011000
fila1[10]=%00111100
fila1[11]=%00111100
fila1[12]=%00011000
fila1[13]=%00000000
fila1[14]=%00000000
fila1[15]=%00000000

```

return

letrama:

```

fila1[0]=%00000000
fila1[1]=%01111111
fila1[2]=%11111111
fila1[3]=%11000011
fila1[4]=%11000011

```

```
fila1[5]=%01111111
fila1[6]=%00111111
fila1[7]=%00000011
fila1[8]=%11000011
fila1[9]=%01111111
fila1[10]=%00111110
fila1[11]=%00000000
fila1[12]=%00000000
fila1[13]=%00000000
fila1[14]=%00000000
fila1[15]=%00000000
return
```

```
letramb:
fila1[0]=%00000000
fila1[1]=%11111100
fila1[2]=%11111110
fila1[3]=%11000011
fila1[4]=%11000011
fila1[5]=%11000011
fila1[6]=%11000011
fila1[7]=%11000011
fila1[8]=%11111110
fila1[9]=%11111100
fila1[10]=%11000000
fila1[11]=%11000000
fila1[12]=%11000000
fila1[13]=%11000000
fila1[14]=%11000000
fila1[15]=%00000000
return
```

```
letramc:
fila1[0]=%00000000
fila1[1]=%00111100
fila1[2]=%01111110
fila1[3]=%11000000
fila1[4]=%11000000
fila1[5]=%11000000
fila1[6]=%11000000
fila1[7]=%11000000
fila1[8]=%11000000
fila1[9]=%01111110
fila1[10]=%00111100
fila1[11]=%00000000
fila1[12]=%00000000
fila1[13]=%00000000
fila1[14]=%00000000
fila1[15]=%00000000
return
```

```
letramd:
fila1[0]=%00000000
fila1[1]=%00111100
fila1[2]=%01111110
fila1[3]=%11000011
fila1[4]=%11000011
fila1[5]=%11000011
fila1[6]=%11000011
fila1[7]=%11000011
fila1[8]=%01111111
fila1[9]=%00111111
fila1[10]=%00000011
fila1[11]=%00000011
fila1[12]=%00000011
fila1[13]=%00000011
fila1[14]=%00000011
```

```
fila1[15]=%00000000  
return
```

```
letrame:  
fila1[0]=%00000000  
fila1[1]=%00111100  
fila1[2]=%01111110  
fila1[3]=%11000011  
fila1[4]=%11000000  
fila1[5]=%11111100  
fila1[6]=%11111110  
fila1[7]=%11000011  
fila1[8]=%11000011  
fila1[9]=%01111110  
fila1[10]=%00111100  
fila1[11]=%00000000  
fila1[12]=%00000000  
fila1[13]=%00000000  
fila1[14]=%00000000  
fila1[15]=%00000000  
return
```

```
letramf:  
fila1[0]=%00000000  
fila1[1]=%11000000  
fila1[2]=%11000000  
fila1[3]=%11000000  
fila1[4]=%11110000  
fila1[5]=%11110000  
fila1[6]=%11000000  
fila1[7]=%11000000  
fila1[8]=%11000011  
fila1[9]=%01111110  
fila1[10]=%00111100  
fila1[11]=%00000000  
fila1[12]=%00000000  
fila1[13]=%00000000  
fila1[14]=%00000000  
fila1[15]=%00000000  
return
```

```
letramg:  
fila1[0]=%00000000  
fila1[1]=%01111110  
fila1[2]=%11111111  
fila1[3]=%11000011  
fila1[4]=%00000011  
fila1[5]=%00111111  
fila1[6]=%01111111  
fila1[7]=%11000011  
fila1[8]=%11000011  
fila1[9]=%01111110  
fila1[10]=%00111100  
fila1[11]=%00000000  
fila1[12]=%00000000  
fila1[13]=%00000000  
fila1[14]=%00000000  
fila1[15]=%00000000  
return
```

```
letramh:  
fila1[0]=%00000000  
fila1[1]=%11000011  
fila1[2]=%11000011  
fila1[3]=%11000011  
fila1[4]=%11000011  
fila1[5]=%11000011
```



```
fila1[6]=%11000011
fila1[7]=%11000011
fila1[8]=%11111111
fila1[9]=%11111110
fila1[10]=%11000000
fila1[11]=%11000000
fila1[12]=%11000000
fila1[13]=%11000000
fila1[14]=%11000000
fila1[15]=%00000000
return
```

letrami:

```
fila1[0]=%00000000
fila1[1]=%00111100
fila1[2]=%00111100
fila1[3]=%00011000
fila1[4]=%00011000
fila1[5]=%00011000
fila1[6]=%00011000
fila1[7]=%00011000
fila1[8]=%00000000
fila1[9]=%00011000
fila1[10]=%00011000
fila1[11]=%00000000
fila1[12]=%00000000
fila1[13]=%00000000
fila1[14]=%00000000
fila1[15]=%00000000
return
```

letramj:

```
fila1[0]=%00000000
fila1[1]=%01110000
fila1[2]=%11111000
fila1[3]=%11011000
fila1[4]=%00011000
fila1[5]=%00011000
fila1[6]=%00011000
fila1[7]=%00011000
fila1[8]=%00000000
fila1[9]=%00011000
fila1[10]=%00011000
fila1[11]=%00000000
fila1[12]=%00000000
fila1[13]=%00000000
fila1[14]=%00000000
fila1[15]=%00000000
return
```

letramk:

```
fila1[0]=%00000000
fila1[1]=%11000110
fila1[2]=%11001100
fila1[3]=%11011000
fila1[4]=%11110000
fila1[5]=%11110000
fila1[6]=%11011000
fila1[7]=%11001100
fila1[8]=%11000110
fila1[9]=%11000000
fila1[10]=%11000000
fila1[11]=%11000000
fila1[12]=%11000000
fila1[13]=%11000000
fila1[14]=%11000000
fila1[15]=%00000000
```

return

letraml:

```
fila1[0]=%00000000
fila1[1]=%00011000
fila1[2]=%00011000
fila1[3]=%00011000
fila1[4]=%00011000
fila1[5]=%00011000
fila1[6]=%00011000
fila1[7]=%00011000
fila1[8]=%00011000
fila1[9]=%00011000
fila1[10]=%00011000
fila1[11]=%00011000
fila1[12]=%00011000
fila1[13]=%00011000
fila1[14]=%00011000
fila1[15]=%00000000
return
```

letramm:

```
fila1[0]=%00000000
fila1[1]=%11000011
fila1[2]=%11000011
fila1[3]=%11000011
fila1[4]=%11000011
fila1[5]=%11000011
fila1[6]=%11011011
fila1[7]=%11011011
fila1[8]=%11011011
fila1[9]=%11111111
fila1[10]=%01100110
fila1[11]=%00000000
fila1[12]=%00000000
fila1[13]=%00000000
fila1[14]=%00000000
fila1[15]=%00000000
return
```

letramn:

```
fila1[0]=%00000000
fila1[1]=%11000011
fila1[2]=%11000011
fila1[3]=%11000011
fila1[4]=%11000011
fila1[5]=%11000011
fila1[6]=%11000011
fila1[7]=%11000011
fila1[8]=%11000011
fila1[9]=%11111110
fila1[10]=%10111100
fila1[11]=%00000000
fila1[12]=%00000000
fila1[13]=%00000000
fila1[14]=%00000000
fila1[15]=%00000000
return
```

letramo:

```
fila1[0]=%00000000
fila1[1]=%00111100
fila1[2]=%01111110
fila1[3]=%11000011
fila1[4]=%11000011
fila1[5]=%11000011
fila1[6]=%11000011
```

```
fila1[7]=%11000011
fila1[8]=%11000011
fila1[9]=%01111110
fila1[10]=%00111100
fila1[11]=%00000000
fila1[12]=%00000000
fila1[13]=%00000000
fila1[14]=%00000000
fila1[15]=%00000000
return
```

```
letramp:
fila1[0]=%00000000
fila1[1]=%11000000
fila1[2]=%11000000
fila1[3]=%11000000
fila1[4]=%11111100
fila1[5]=%11111110
fila1[6]=%11000011
fila1[7]=%11000011
fila1[8]=%11000011
fila1[9]=%01111110
fila1[10]=%00111100
fila1[11]=%00000000
fila1[12]=%00000000
fila1[13]=%00000000
fila1[14]=%00000000
fila1[15]=%00000000
return
```

```
letramq:
fila1[0]=%00000000
fila1[1]=%00000011
fila1[2]=%00000011
fila1[3]=%00000011
fila1[4]=%00000011
fila1[5]=%00111111
fila1[6]=%01111111
fila1[7]=%11000011
fila1[8]=%11000011
fila1[9]=%01111110
fila1[10]=%00111100
fila1[11]=%00000000
fila1[12]=%00000000
fila1[13]=%00000000
fila1[14]=%00000000
fila1[15]=%00000000
return
```

```
letramr:
fila1[0]=%00000000
fila1[1]=%11000000
fila1[2]=%11000000
fila1[3]=%11000000
fila1[4]=%11000000
fila1[5]=%11000000
fila1[6]=%11100000
fila1[7]=%11111110
fila1[8]=%11011100
fila1[9]=%11000000
fila1[10]=%11000000
fila1[11]=%00000000
fila1[12]=%00000000
fila1[13]=%00000000
fila1[14]=%00000000
fila1[15]=%00000000
return
```

```

letrams:
fila1[0]=%00000000
fila1[1]=%00111110
fila1[2]=%01111111
fila1[3]=%11000011
fila1[4]=%00000011
fila1[5]=%00111110
fila1[6]=%01111100
fila1[7]=%11000000
fila1[8]=%11000011
fila1[9]=%01111110
fila1[10]=%00111100
fila1[11]=%00000000
fila1[12]=%00000000
fila1[13]=%00000000
fila1[14]=%00000000
fila1[15]=%00000000
return

```

```

letramt:
fila1[0]=%00000000
fila1[1]=%00111100
fila1[2]=%01111110
fila1[3]=%01100011
fila1[4]=%01100000
fila1[5]=%01100000
fila1[6]=%01100000
fila1[7]=%01100000
fila1[8]=%01100000
fila1[9]=%11110000
fila1[10]=%11110000
fila1[11]=%01100000
fila1[12]=%01100000
fila1[13]=%01100000
fila1[14]=%01100000
fila1[15]=%00000000
return

```

```

letramu:
fila1[0]=%00000000
fila1[1]=%00111011
fila1[2]=%01111111
fila1[3]=%11000011
fila1[4]=%11000011
fila1[5]=%11000011
fila1[6]=%11000011
fila1[7]=%11000011
fila1[8]=%11000011
fila1[9]=%11000011
fila1[10]=%11000011
fila1[11]=%00000000
fila1[12]=%00000000
fila1[13]=%00000000
fila1[14]=%00000000
fila1[15]=%00000000
return

```

```

letramv:
fila1[0]=%00000000
fila1[1]=%00011000
fila1[2]=%00111100
fila1[3]=%01100110
fila1[4]=%01100110
fila1[5]=%01100110
fila1[6]=%01100110
fila1[7]=%11000011

```

```
fila1[8]=%11000011
fila1[9]=%11000011
fila1[10]=%11000011
fila1[11]=%00000000
fila1[12]=%00000000
fila1[13]=%00000000
fila1[14]=%00000000
fila1[15]=%00000000
return
```

letramw:

```
fila1[0]=%00000000
fila1[1]=%01100110
fila1[2]=%11111111
fila1[3]=%11011011
fila1[4]=%11011011
fila1[5]=%11000011
fila1[6]=%11000011
fila1[7]=%11000011
fila1[8]=%11000011
fila1[9]=%11000011
fila1[10]=%11000011
fila1[11]=%00000000
fila1[12]=%00000000
fila1[13]=%00000000
fila1[14]=%00000000
fila1[15]=%00000000
return
```

letramx:

```
fila1[0]=%00000000
fila1[1]=%11000011
fila1[2]=%11000011
fila1[3]=%11000011
fila1[4]=%01100110
fila1[5]=%00111100
fila1[6]=%00011000
fila1[7]=%00111100
fila1[8]=%01100110
fila1[9]=%11000011
fila1[10]=%11000011
fila1[11]=%00000000
fila1[12]=%00000000
fila1[13]=%00000000
fila1[14]=%00000000
fila1[15]=%00000000
return
```

letramy:

```
fila1[0]=%00000000
fila1[1]=%01111110
fila1[2]=%11111111
fila1[3]=%11000011
fila1[4]=%00000011
fila1[5]=%00111111
fila1[6]=%01111111
fila1[7]=%11000011
fila1[8]=%11000011
fila1[9]=%11000011
fila1[10]=%11000011
fila1[11]=%00000000
fila1[12]=%00000000
fila1[13]=%00000000
fila1[14]=%00000000
fila1[15]=%00000000
return
```

```
letramz:  
fila1[0]=%00000000  
fila1[1]=%11111111  
fila1[2]=%11111111  
fila1[3]=%01100000  
fila1[4]=%00110000  
fila1[5]=%01111110  
fila1[6]=%00111111  
fila1[7]=%00000110  
fila1[8]=%00000011  
fila1[9]=%11111111  
fila1[10]=%11111111  
fila1[11]=%00000000  
fila1[12]=%00000000  
fila1[13]=%00000000  
fila1[14]=%00000000  
fila1[15]=%00000000  
return
```

```
grados:  
fila1[0]=%00000000  
fila1[1]=%00000000  
fila1[2]=%00000000  
fila1[3]=%00000000  
fila1[4]=%00000000  
fila1[5]=%00000000  
fila1[6]=%00000000  
fila1[7]=%00000000  
fila1[8]=%00000000  
fila1[9]=%00000000  
fila1[10]=%00111100  
fila1[11]=%01111110  
fila1[12]=%01100110  
fila1[13]=%01111110  
fila1[14]=%00111100  
fila1[15]=%00000000  
return
```

```
espe1:  
fila1[0]=%00000000  
fila1[1]=%00000000  
fila1[2]=%00011000  
fila1[3]=%00111100  
fila1[4]=%00011000  
fila1[5]=%00000000  
fila1[6]=%00011000  
fila1[7]=%00011000  
fila1[8]=%00011000  
fila1[9]=%00011000  
fila1[10]=%00011000  
fila1[11]=%00011000  
fila1[12]=%00011000  
fila1[13]=%00011000  
fila1[14]=%00011000  
fila1[15]=%00000000  
return
```

```
espe2:  
fila1[0]=%00000000  
fila1[1]=%00000000  
fila1[2]=%00000000  
fila1[3]=%00000000  
fila1[4]=%00000000  
fila1[5]=%00000000  
fila1[6]=%00000000  
fila1[7]=%00000000  
fila1[8]=%00000000
```

```
fila1[9]=%00000000
fila1[10]=%01100110
fila1[11]=%01100110
fila1[12]=%01100110
fila1[13]=%01100110
fila1[14]=%01100110
fila1[15]=%00000000
return
```

espe3:

```
fila1[0]=%00000000
fila1[1]=%01100110
fila1[2]=%01100110
fila1[3]=%01100110
fila1[4]=%01100110
fila1[5]=%11111111
fila1[6]=%11111111
fila1[7]=%01100110
fila1[8]=%01100110
fila1[9]=%11111111
fila1[10]=%11111111
fila1[11]=%01100110
fila1[12]=%01100110
fila1[13]=%01100110
fila1[14]=%01100110
fila1[15]=%00000000
return
```

espe4:

```
fila1[0]=%00011000
fila1[1]=%00111100
fila1[2]=%01111110
fila1[3]=%11011011
fila1[4]=%00011011
fila1[5]=%00011011
fila1[6]=%00111110
fila1[7]=%01111100
fila1[8]=%11011000
fila1[9]=%11011000
fila1[10]=%11011000
fila1[11]=%11011000
fila1[12]=%11011011
fila1[13]=%01111110
fila1[14]=%00111100
fila1[15]=%00011000
return
```

espe5:

```
fila1[0]=%00000000
fila1[1]=%00000000
fila1[2]=%00000110
fila1[3]=%10001111
fila1[4]=%11001111
fila1[5]=%01100110
fila1[6]=%00110000
fila1[7]=%00011000
fila1[8]=%00001100
fila1[9]=%00000110
fila1[10]=%01100011
fila1[11]=%11110001
fila1[12]=%11110000
fila1[13]=%01100000
fila1[14]=%00000000
fila1[15]=%00000000
return
```

espe6:

```
fila1[0]=%00000000
fila1[1]=%00000000
fila1[2]=%00000000
fila1[3]=%00000000
fila1[4]=%00000000
fila1[5]=%00000000
fila1[6]=%00000000
fila1[7]=%00000000
fila1[8]=%00000000
fila1[9]=%00000000
fila1[10]=%00000000
fila1[11]=%00011000
fila1[12]=%00011000
fila1[13]=%00011000
fila1[14]=%00011000
fila1[15]=%00000000
return
```

espe7:

```
fila1[0]=%00000000
fila1[1]=%00000110
fila1[2]=%00001100
fila1[3]=%00011000
fila1[4]=%00110000
fila1[5]=%01100000
fila1[6]=%01100000
fila1[7]=%01100000
fila1[8]=%01100000
fila1[9]=%01100000
fila1[10]=%01100000
fila1[11]=%00110000
fila1[12]=%00011000
fila1[13]=%00001100
fila1[14]=%00000110
fila1[15]=%00000000
return
```

espe8:

```
fila1[0]=%00000000
fila1[1]=%01100000
fila1[2]=%00110000
fila1[3]=%00011000
fila1[4]=%00001100
fila1[5]=%00000110
fila1[6]=%00000110
fila1[7]=%00000110
fila1[8]=%00000110
fila1[9]=%00000110
fila1[10]=%00000110
fila1[11]=%00001100
fila1[12]=%00011000
fila1[13]=%01110000
fila1[14]=%01100000
fila1[15]=%00000000
return
```

espe9:

```
fila1[0]=%00000000
fila1[1]=%11000000
fila1[2]=%11000000
fila1[3]=%01100000
fila1[4]=%01100000
fila1[5]=%00110000
fila1[6]=%00110000
fila1[7]=%00011000
fila1[8]=%00011000
fila1[9]=%00001100
```



```
fila1[10]=%00001100
fila1[11]=%00000110
fila1[12]=%00000110
fila1[13]=%00000011
fila1[14]=%00000011
fila1[15]=%00000000
return
```

comparador:

```
if una="A" TheN
  gosub letraa
endif
if una="N" TheN
  gosub lettran
endif
if una="G" TheN
  gosub letrag
endif
if una="Y" TheN
  gosub letray
endif
if una=" " TheN
  gosub letraespacio
endif
if una="H" TheN
  gosub letrah
endif
if una="O" TheN
  gosub letrao
endif
if una="L" TheN
  gosub letral
endif
if una="B" TheN
  gosub letrab
endif
if una="C" TheN
  gosub letrac
endif
if una="D" TheN
  gosub letrad
endif

if una="E" TheN
  gosub letrae
endif

if una="F" TheN
  gosub letraf
endif

if una="I" TheN
  gosub letrai
endif

if una="J" TheN
  gosub letraj
endif

if una="K" TheN
  gosub letrak
endif

if una="M" TheN
  gosub letram
```

```
endif
if una="P" TheN
  gosub letrap
endif
if una="R" TheN
  gosub letrar
endif
if una="V" TheN
  gosub letrav
endif
if una="Q" TheN
  gosub letraq
endif

if una="S" TheN
  gosub letrass
endif
if una="T" TheN
  gosub letrat
endif
if una="U" TheN
  gosub letrau
endif

if una="W" TheN
  gosub letraw
endif

if una="X" TheN
  gosub letrax
endif

if una="Z" TheN
  gosub letraz
endif

if una="0" TheN
  gosub cero
endif

if una="1" TheN
  gosub uno
endif

if una="2" TheN
  gosub dos
endif

if una="3" TheN
  gosub tres
endif

if una="4" TheN
  gosub cuatro
endif

if una="5" TheN
  gosub cinco
endif

if una="6" TheN
  gosub seis
endif

if una="7" TheN
  gosub siete
endif
```

```
if una="8" TheN
  gosub ocho
endif

if una="9" TheN
  gosub nueve
endif
if una=":" TheN
  gosub puntos
endif
if una="a" TheN
  gosub letrama
endif
if una="b" TheN
  gosub letramb
endif
if una="c" TheN
  gosub letramc
endif
if una="d" TheN
  gosub letramd
endif
if una="e" TheN
  gosub letrame
endif
if una="f" TheN
  gosub letramf
endif
if una="g" TheN
  gosub letramg
endif
if una="h" TheN
  gosub letramh
endif
if una="i" TheN
  gosub letrami
endif
if una="j" TheN
  gosub letramj
endif
if una="k" TheN
  gosub letramk
endif
if una="l" TheN
  gosub letraml
endif
if una="m" TheN
  gosub letramm
endif
if una="n" TheN
  gosub letramn
endif
if una="o" TheN
  gosub letramo
endif
if una="p" TheN
  gosub letramp
endif
if una="q" TheN
  gosub letramq
endif
if una="r" TheN
  gosub letramr
endif
if una="s" TheN
  gosub letrams
```

```
endif
if una="t" TheN
  gosub letramt
endif
if una="u" TheN
  gosub letramu
endif
if una="v" TheN
  gosub letramv
endif
if una="w" TheN
  gosub letramw
endif
if una="x" TheN
  gosub letramx
endif
if una="y" TheN
  gosub letramy
endif
if una="z" TheN
  gosub letramz
endif
if una="}" TheN
  gosub grados
endif
if una="!" TheN
  gosub espe1
endif
if una=34 TheN
  gosub espe2
endif
if una="#" TheN
  gosub espe3
endif
if una="$" TheN
  gosub espe4
endif
if una="%" TheN
  gosub espe5
endif
if una=""" TheN
  gosub espe6
endif
if una="(" TheN
  gosub espe7
endif
if una=")" TheN
  gosub espe8
endif
if una="/" TheN
  gosub espe9
endif

return

bucle1:
  b0=1
  pauseus 120
  b0=0
  pauseus 3
return

bucle2:
  b1=1
  pauseus 120
  b1=0
  pauseus 3
```

return

```
bucle3:  
  b2=1  
  pauseus 120  
  b2=0  
  pauseus 3  
return
```

```
bucle4:  
  b3=1  
  pauseus 120  
  b3=0  
  pauseus 3  
return
```

```
bucle5:  
  b4=1  
  pauseus 120  
  b4=0  
  pauseus 3  
return
```

```
bucle6:  
  b5=1  
  pauseus 120  
  b5=0  
  pauseus 3  
return
```

```
bucle7:  
  b6=1  
  pauseus 120  
  b6=0  
  pauseus 3  
return
```

```
bucle8:  
  b7=1  
  pauseus 120  
  b7=0  
  pauseus 3  
return
```

```
bucle9:  
  b8=1  
  pauseus 120  
  b8=0  
  pauseus 3  
return
```

```
bucle10:  
  b9=1  
  pauseus 120  
  b9=0  
  pauseus 3  
return
```

```
bucle11:  
  b10=1  
  pauseus 120  
  b10=0  
  pauseus 3  
return
```

```
bucle12:  
  b11=1
```

```
    pauseus 120
    b11=0
    pauseus 3
return

bucle13:
    b12=1
    pauseus 120
    b12=0
    pauseus 3
return

bucle14:
    b13=1
    pauseus 120
    b13=0
    pauseus 3
return

bucle15:
    b14=1
    pauseus 120
    b14=0
    pauseus 3
return

bucle16:
    b15=1
    pauseus 120
    b15=0
    pauseus 3
return

limpia:
    for i=0 to 40
        dato=0
        clock=0
        clock=1
    next i
return

bucle:
for i=j to control

    if dao.7=0 then
'   if coje=0 then
'       dato=0
'       pauseus 1
'       clock=0
'       pauseus 1
'       clock=1
'       pauseus 1
    else
'       dato=1
'       pauseus 1
'       clock=0
'       pauseus 1
'       clock=1
'       pauseus 1
    endif
    dao=dao<<1

next
'pauseus 800
return
```

```
'=====
```

primeralectura:

```

pause 500
' serout portc.6,T9600,["#INGRESE LA CANTIDAD DE CARACTERES#"]
serout portc.6,T9600,["#"]
' serin portc.7,T9600,5000,recupera,datos
serin portc.7,T9600,3000,primeralectura,datouni
pause 300
serin portc.7,T9600,3000,primeralectura,datodec
pause 300
serin portc.7,T9600,3000,primeralectura,datocen
pause 300
temporal=datouni
gosub cambiobdato
datouni=temporal
temporal=datodec
gosub cambiobdato
datodec=temporal
temporal=datocen
gosub cambiobdato
datocen=temporal
datocen=datocen*100
datodec=datodec*10

numero=datouni+datodec+datocen
write 0,1
write 1,numero

```

return

recolecciondatos:

```

' serout portc.6,T9600,["#INGRESE SUS CARACTERES#"]
serout portc.6,T9600,["@"]
for i=0 to numero-1
  serin portc.7,T9600,datos
  ' serin portc.7,T9600,3000,inicio,datos
  if validatiempo=1 then
    horass[conteotiempos]=datos
    conteotiempos=conteotiempos+1
  endif

  if datos="]" then
    datos=" "
    validatiempo=1
    posicion=i+1
    write 255,1
    write 254,posicion
  endif
  "Temperatura
  if datos="{" then
    datos=" "
    validatemp=1
    posiciontemp=i+1
    write 253,1
    write 252,posiciontemp
  endif

  write i+2,datos
next i

if validatiempo=1 then
temporal=horass[0]
gosub cambiobdato
horass[0]=temporal

```

```

temporal=horass[1]
gosub cambiodato
horass[1]=temporal
temporal=horass[3]
gosub cambiodato
horass[3]=temporal
temporal=horass[4]
gosub cambiodato
horass[4]=temporal

horass[0]=horass[0]*10
anterior=horass[0]+horass[1]

gosub inverso
horao=nueva
horass[3]=horass[3]*10
anterior=horass[3]+horass[4]
gosub inversom
minuo=nueva
gosub grabarrtc
else
  validatiempo=0
  write 255,0

endif

if validatemp=1 then

else
  validatemp=0
  write 253,0
endif

' serout portc.6,T9600,["#CARACTERES INGRESADOS CORRECTAMENTE#"]
serout portc.6,T9600,["*"]

return

cambiodato:

if temporal=48 then temporal=0
if temporal=49 then temporal=1
if temporal=50 then temporal=2
if temporal=51 then temporal=3
if temporal=52 then temporal=4
if temporal=53 then temporal=5
if temporal=54 then temporal=6
if temporal=55 then temporal=7
if temporal=56 then temporal=8
if temporal=57 then temporal=9

return

recupera:

read 255,validatiempo
read 254,posicion
read 253,validatemp
read 252,posiciontemp

if validatemp=1 then
  trisa=%00000001
  adcon0=%11000001
  adcon1=%00001110
  ADCIN 0, analogo

```



```

    pause 300
    analogo = analogo /128
    unidad=analogo/10
    decena=analogo mod 10
    momentanea=unidad
    asci=momentanea
    gosub cambiaascci
    letras[posiciontemp]=asci
    momentanea=decena
    asci=momentanea
    gosub cambiaascci
    letras[posiciontemp+1]=asci
endif

if validatiempo=1 then
    i2cread DPIN,CPIN,%11010000,0,[segu]
    i2cread DPIN,CPIN,%11010000,1,[minu]
    i2cread DPIN,CPIN,%11010000,2,[hora]
    i2cread DPIN,CPIN,%11010000,3,[dias]
    i2cread DPIN,CPIN,%11010000,4,[diam]
    i2cread DPIN,CPIN,%11010000,5,[mes]
    i2cread DPIN,CPIN,%11010000,6,[anio]

    gosub cambiarhora
    gosub cambiarminu

    unidad=a/10
    decena=a mod 10
    momentanea=unidad
    asci=momentanea
    gosub cambiaascci
    letras[posicion]=asci

    momentanea=decena
    asci=momentanea
    gosub cambiaascci
    letras[posicion+1]=asci

    unidad=b/10
    decena=b mod 10
    momentanea=unidad
    asci=momentanea
    gosub cambiaascci
    letras[posicion+3]=asci

    momentanea=decena
    asci=momentanea
    gosub cambiaascci
    letras[posicion+4]=asci
    letras[posicion+2]=58
endif

read 1,numero

for i=0 to numero-1
    read i+2,letras[i]
    auxs[i]=letras[i]
next i
goto inicio

inverso:
if anterior=1 then nueva=$1
if anterior=2 then nueva=$2
if anterior=3 then nueva=$3

```

```
if anterior=4 then nueva=$4
if anterior=5 then nueva=$5
if anterior=6 then nueva=$6
if anterior=7 then nueva=$7
if anterior=8 then nueva=$8
if anterior=9 then nueva=$9
if anterior=10 then nueva=$10
if anterior=11 then nueva=$11
if anterior=12 then nueva=$12
if anterior=13 then nueva=$13
if anterior=14 then nueva=$14
if anterior=15 then nueva=$15
if anterior=16 then nueva=$16
if anterior=17 then nueva=$17
if anterior=18 then nueva=$18
if anterior=19 then nueva=$19
if anterior=20 then nueva=$20
if anterior=21 then nueva=$21
if anterior=22 then nueva=$22
if anterior=23 then nueva=$23
if anterior=0 then nueva=$24
return
```

inversom:

```
if anterior=1 then nueva=$1
if anterior=2 then nueva=$2
if anterior=3 then nueva=$3
if anterior=4 then nueva=$4
if anterior=5 then nueva=$5
if anterior=6 then nueva=$6
if anterior=7 then nueva=$7
if anterior=8 then nueva=$8
if anterior=9 then nueva=$9
if anterior=10 then nueva=$10
if anterior=11 then nueva=$11
if anterior=12 then nueva=$12
if anterior=13 then nueva=$13
if anterior=14 then nueva=$14
if anterior=15 then nueva=$15
if anterior=16 then nueva=$16
if anterior=17 then nueva=$17
if anterior=18 then nueva=$18
if anterior=19 then nueva=$19
if anterior=20 then nueva=$20
if anterior=21 then nueva=$21
if anterior=22 then nueva=$22
if anterior=23 then nueva=$23
if anterior=24 then nueva=$24
if anterior=25 then nueva=$25
if anterior=26 then nueva=$26
if anterior=27 then nueva=$27
if anterior=28 then nueva=$28
if anterior=29 then nueva=$29
if anterior=30 then nueva=$30
if anterior=31 then nueva=$31
if anterior=32 then nueva=$32
if anterior=33 then nueva=$33
if anterior=34 then nueva=$34
if anterior=35 then nueva=$35
if anterior=36 then nueva=$36
if anterior=37 then nueva=$37
if anterior=38 then nueva=$38
if anterior=39 then nueva=$39
if anterior=40 then nueva=$40
if anterior=41 then nueva=$41
if anterior=42 then nueva=$42
if anterior=43 then nueva=$43
```

```

if anterior=44 then nueva=$44
if anterior=45 then nueva=$45
if anterior=46 then nueva=$46
if anterior=47 then nueva=$47
if anterior=48 then nueva=$48
if anterior=49 then nueva=$49
if anterior=50 then nueva=$50
if anterior=51 then nueva=$51
if anterior=52 then nueva=$52
if anterior=53 then nueva=$53
if anterior=54 then nueva=$54
if anterior=55 then nueva=$55
if anterior=56 then nueva=$56
if anterior=57 then nueva=$57
if anterior=58 then nueva=$58
if anterior=59 then nueva=$59
if anterior=0 then nueva=$0

```

```
return
```

grabarRTC:

```

pause 2000
i2cwrite DPIN,CPIN,%11010000,0,[$00]
pause 10
i2cwrite DPIN,CPIN,%11010000,1,[minuo]
pause 10
i2cwrite DPIN,CPIN,%11010000,2,[horao]
pause 10
i2cwrite DPIN,CPIN,%11010000,3,[$01]
pause 10
i2cwrite DPIN,CPIN,%11010000,4,[$27]
pause 10
i2cwrite DPIN,CPIN,%11010000,5,[$9]
pause 10
i2cwrite DPIN,CPIN,%11010000,6,[$04]
pause 10

```

```
PAUSE 100
```

```
return
```

cambiarhora:

```

if hora=$1 then a=1
if hora=$2 then a=2
if hora=$3 then a=3
if hora=$4 then a=4
if hora=$5 then a=5
if hora=$6 then a=6
if hora=$7 then a=7
if hora=$8 then a=8
if hora=$9 then a=9
if hora=$10 then a=10
if hora=$11 then a=11
if hora=$12 then a=12
if hora=$13 then a=13
if hora=$14 then a=14
if hora=$15 then a=15
if hora=$16 then a=16
if hora=$17 then a=17
if hora=$18 then a=18
if hora=$19 then a=19
if hora=$20 then a=20
if hora=$21 then a=21
if hora=$22 then a=22
if hora=$23 then a=23
if hora=$0 then a=24
return

```

cambiarminu:

```
if minu=$1 then b=1
if minu=$2 then b=2
if minu=$3 then b=3
if minu=$4 then b=4
if minu=$5 then b=5
if minu=$6 then b=6
if minu=$7 then b=7
if minu=$8 then b=8
if minu=$9 then b=9
if minu=$10 then b=10
if minu=$11 then b=11
if minu=$12 then b=12
if minu=$13 then b=13
if minu=$14 then b=14
if minu=$15 then b=15
if minu=$16 then b=16
if minu=$17 then b=17
if minu=$18 then b=18
if minu=$19 then b=19
if minu=$20 then b=20
if minu=$21 then b=21
if minu=$22 then b=22
if minu=$23 then b=23
if minu=$24 then b=24
if minu=$25 then b=25
if minu=$26 then b=26
if minu=$27 then b=27
if minu=$28 then b=28
if minu=$29 then b=29
if minu=$30 then b=30
if minu=$31 then b=31
if minu=$32 then b=32
if minu=$33 then b=33
if minu=$34 then b=34
if minu=$35 then b=35
if minu=$36 then b=36
if minu=$37 then b=37
if minu=$38 then b=38
if minu=$39 then b=39
if minu=$40 then b=40
if minu=$41 then b=41
if minu=$42 then b=42
if minu=$43 then b=43
if minu=$44 then b=44
if minu=$45 then b=45
if minu=$46 then b=46
if minu=$47 then b=47
if minu=$48 then b=48
if minu=$49 then b=49
if minu=$50 then b=50
if minu=$51 then b=51
if minu=$52 then b=52
if minu=$53 then b=53
if minu=$54 then b=54
if minu=$55 then b=55
if minu=$56 then b=56
if minu=$57 then b=57
if minu=$58 then b=58
if minu=$59 then b=59
if minu=$0 then b=0
```

return

cambiaascci:

```

if asci=0 then asci=48
if asci=1 then asci=49
if asci=2 then asci=50
if asci=3 then asci=51
if asci=4 then asci=52
if asci=5 then asci=53
if asci=6 then asci=54
if asci=7 then asci=55
if asci=8 then asci=56
if asci=9 then asci=57
return
'=====

```

```

cambiadatos:
  gosub primeralectura
  gosub recolecciondatos
  goto recupera

goto inicio

```

## Control Mensaje

```

Asd
program FINAL

'0986234697
dim j as byte
dim ok as string [3]
dim sigue as byte
dim b as byte
dim compara1 as string[4]
dim mensaje as string[200]
dim dato as string[4]
dim tomar as byte
dim k as byte
dim envia as byte
dim h as byte
dim espera as string[4]
dim tmp as byte
dim uno as byte
dim dos as byte
dim tres as byte
dim i as byte
dim musica as string[2]
dim conteomusic as byte
dim dato1 as byte
dim dato2 as byte
dim compara as string[9]
dim momen as byte
dim timeout as word
dim clave as string[8]
dim verificador as byte
dim paso as byte
dim nuevo as byte
dim prueba as string[4]
dim finatime as word
dim mensa1 as string[50]
dim mensa2 as string[50]
dim mensa3 as string[50]
dim mensa4 as string[50]
dim auto as byte

sub procedure inimp3

Soft_Uart_Write(0x7E)

```

```

Soft_Uart_Write(0x07)
Soft_Uart_Write(0xA0)
Soft_Uart_Write(0x30)
Soft_Uart_Write(0x30)
Soft_Uart_Write(0x30)
Soft_Uart_Write(0x30)

end sub

sub procedure ini2mp3

Soft_Uart_Write(0x7E)
Soft_Uart_Write(0x07)
Soft_Uart_Write(0xA0)
Soft_Uart_Write(0x30)
Soft_Uart_Write(0x30)
Soft_Uart_Write(0x30)

end sub

sub procedure finmp3

Soft_Uart_Write(0x7E)

end sub

sub procedure inicio

portc.4=0
portc.5=0
portb.3=1
while h<1
  if Usart_Data_Ready() = 1 then
    dato[0]=Usart_Read
    if (dato[0]=espera[0]) then
      h=h+1
    end if
  end if
wend
portb.3=0
delay_ms(1000)
Usart_Write(5)
delay_ms(1000)
Usart_Write(0)
delay_ms(1000)
Usart_Write(0)
delay_ms(1000)
portb.3=1
h=0
while h<1
  if Usart_Data_Ready() = 1 then
    dato[0]=Usart_Read
    if (dato[0]=espera[1]) then
      h=h+1
    end if
  end if
wend

portc.4=0
portc.5=0

portb.3=0
Usart_Write("T")
delay_ms(1000)
Usart_Write("e")
delay_ms(1000)

```

```

Usart_Write("s")
delay_ms(1000)
Usart_Write("T")
delay_ms(1000)
Usart_Write(" ")

```

```

h=0
'while h<1
' if Usart_Data_Ready() = 1 then
'   dato[0]=Usart_Read
'   if (dato[0]=espera[2]) then
'     h=h+1
'   end if
' end if
'wend

```

```

portb.1=1
portb.2=1
portb.3=1
delay_ms(3000)
portb.1=1
portb.2=0
portb.3=0
portb.4=0
portb.5=0

```

```
end sub
```

```
sub procedure hacer
```

```

portb.3=1
h=0
while h<1
  if Usart_Data_Ready() = 1 then
    dato[0]=Usart_Read
    if (dato[0]=espera[0]) then
      h=h+1
    end if
  end if

```

```
end if
```

```
if finatime>10000 then
```

```

  uno=0
  dos=0
  tres=0
  finatime=0
  h=1
end if

```

```
wend
```

```

portb.3=0
delay_ms(1000)
'Usart_Write(5)

```

```
if (uno=9) then
```

```

  uno=0
  dos=dos+1
else
  uno=uno+1
end if

```

```

Usart_Write(uno)
'Usart_Write(2)
delay_ms(1000)
Usart_Write(dos)
'Usart_Write(8)
delay_ms(1000)

```

```

'Usart_Write(0)
Usart_Write(tres)
delay_ms(1000)
portb.3=1
h=0
while h<1
  if Usart_Data_Ready() = 1 then
    dato[0]=Usart_Read
    if (dato[0]=espera[1]) then
      h=h+1
    end if
  end if

  if finatime>10000 then
    finatime=0
    uno=0
    dos=0
    tres=0
    h=1
  end if
wend

portc.4=0
portc.5=0

portb.3=0
'Usart_Write("f")
'delay_ms(2000)
'Usart_Write("r")
'delay_ms(2000)
'Usart_Write("a")
'delay_ms(2000)
'Usart_Write("n")
'delay_ms(2000)
'Usart_Write(" ")

if (uno=0) then
  uno=8
  dos=dos-1
else
  if (uno=0) and (dos=0) and (tres=1) then
    uno=8
    dos=9
    tres=0
  else
    if (uno=0) and (dos=0) and (tres=2) then
      uno=8
      dos=9
      tres=1
    else
      if (uno=1) and (dos=0) and (tres=1) then
        uno=9
        dos=9
        tres=0
      else
        if (uno=1) and (dos=0) and (tres=2) then
          uno=9
          dos=9
          tres=1
        else
          uno=uno-2
        end if
      end if
    end if
  end if
end if
end if
end if

```



```

tres=tres*100
dos=dos*10
uno=tres+dos+uno

'uno=80
for i=0 to uno
Usart_Write(mensaje[i])
portb.2=1
delay_ms(1000)
portb.2=0
next i

Usart_Write(" ")
delay_ms(1000)
'Usart_Write(" ")
'delay_ms(1000)

h=0
while h<1
  if Usart_Data_Ready() = 1 then
    dato[0]=Usart_Read
    if (dato[0]=espera[2]) then
      h=h+1
    end if
  end if

  if (finatime>10000) then
    uno=0
    dos=0
    tres=0
    h=1
    finatime=0
  end if

wend

portb.1=1
portb.2=1
portb.3=1
delay_ms(3000)
portb.1=1
portb.2=0
portb.3=0

end sub

sub procedure cambia1
if (compara[0]=musica[0]) then
  dato1=0x30
end if
if (compara[1]=musica[0]) then
  dato1=0x31
end if
if (compara[2]=musica[0]) then
  dato1=0x32
end if
if (compara[3]=musica[0]) then
  dato1=0x33
end if
if (compara[4]=musica[0]) then
  dato1=0x34
end if
if (compara[5]=musica[0]) then
  dato1=0x35

```

```
end if
if (compara[6]=musica[0]) then
  dato1=0x36
end if
if (compara[7]=musica[0]) then
  dato1=0x37
end if
if (compara[8]=musica[0]) then
  dato1=0x38
end if
if (compara[9]=musica[0]) then
  dato1=0x39
end if
```

```
end sub
```

```
sub procedure cambia2
```

```
if (compara[0]=musica[1]) then
  dato2=0x30
end if
if (compara[1]=musica[1]) then
  dato2=0x31
end if
if (compara[2]=musica[1]) then
  dato2=0x32
end if
if (compara[3]=musica[1]) then
  dato2=0x33
end if
if (compara[4]=musica[1]) then
  dato2=0x34
end if
if (compara[5]=musica[1]) then
  dato2=0x35
end if
if (compara[6]=musica[1]) then
  dato2=0x36
end if
if (compara[7]=musica[1]) then
  dato2=0x37
end if
if (compara[8]=musica[1]) then
  dato2=0x38
end if
if (compara[9]=musica[1]) then
  dato2=0x39
end if
```

```
end sub
```

```
main:
```

```
auto=0
```

```
compara[0]="0"
compara[1]="1"
compara[2]="2"
compara[3]="3"
compara[4]="4"
compara[5]="5"
```

```

compara[6]="6"
compara[7]="7"
compara[8]="8"
compara[9]="9"

clave[0]="3"
clave[1]="9"
clave[2]="5"
clave[3]="2" "Normal"
clave[4]="6" "MENSAJE 1"
clave[5]="7" "MENSAJE 2"
clave[6]="8" "MENSAJE 3"
clave[7]="9" "MENSAJE 4"

mensa1="Inicio de Matriculas" '20
mensa2="Inicio de Clases" '16
mensa3="Pago de Pensiones" '17
mensa4="Inicio de Examenes" '18

conteomusic=0

sigue=0
trisb=0
portb=0
b=0
trisc=%10001100
portc=0
Usart_Init(9600)

Soft_Uart_Init(PORTB, 6, 0, 9600, 0)

j=0

compara1[0]="h"
compara1[1]="o"
compara1[2]="l"
compara1[3]="a"

espera[0]="#"
espera[1]="@"
espera[2]="*"

ok[0]="@"
ok[1]="*"
ok[2]="#"
ok[3]="%"

portc.4=0
portc.5=0

envia=0
h=0

tmp = Eeprom_Read(0)

uno=0
dos=0
tres=0

' portb.4=1
' portb.5=1
' inicio
' Eeprom_Write(0,10)
' delay_ms(1000)
' portb.4=0
' portb.5=0

```

```

if (tmp=10) then

else
  portb.4=1
  portb.5=1
  inicio
  Eeprom_Write(0,10)
  delay_ms(1000)
  portb.4=0
  portb.5=0
end if
delay_ms(3000)

Usart_Write_Text("AT")
Usart_Write(13)
delay_ms(1000)
Usart_Write_Text("at+cnmi=1,2,0,0,0")
Usart_Write(13)
delay_ms(1000)
Usart_Write_Text("at+cmgf=1")
Usart_Write(13)
delay_ms(1000)
Usart_Write_Text("ATE0")
Usart_Write(13)
delay_ms(1000)
tomar=0

'inimp3

'Soft_Uart_Write(0x31)

'finmp3

portb.1=1

portb.4=0

portb.5=0
j=0
uno=0
k=0
timeout=0

verificador=0

paso=0

nuevo=0

finatime=0

while(1)

while j<205

  envia=0
  if Usart_Data_Ready() = 1 then
    portb.3=1
    dato[0]= Usart_Read
    if ((dato[0]=ok[2])and(tomar=2)) then
      timeout=0
      tomar=0
      j=230
    end if
  end if
end while
end while

```

```

if tomar=1 then
  timeout=0
  mensaje[j]= dato[0]'Usart_Read
  k=k+1
  j=j+1
end if

if (tomar=2) then
  timeout=0
  musica[conteomusic]=dato[0]
  conteomusic=conteomusic+1
end if

if (dato[0]=clave[0]) and (nuevo=1) then
  portb.2=1
  delay_ms(2000)
  timeout=0
  verificador=verificador+1
  nuevo=0
end if
if (dato[0]=clave[1]) and (verificador=1) then
  portb.2=1
  delay_ms(2000)
  timeout=0
  verificador=verificador+1
  nuevo=0
end if
if (dato[0]=clave[2]) and (verificador=2) then
  portb.2=1
  delay_ms(2000)
  timeout=0
  verificador=verificador+1
  nuevo=0
end if
if (dato[0]=clave[3]) and (verificador=3) then
  portb.2=1
  delay_ms(2000)
  timeout=0
  verificador=verificador+1
  tomar=1
  nuevo=0
end if

if (dato[0]=clave[4]) and (verificador=3) then

  mensaje=mensa1
  k=20
  j=20
  tomar=1
  conteomusic=1
  musica[0]="1"

end if

if (dato[0]=clave[5]) and (verificador=3) then

  mensaje=mensa2
  k=16
  j=16
  tomar=1
  conteomusic=1
  musica[0]="2"

end if

if (dato[0]=clave[6]) and (verificador=3) then

```

```

    mensaje=mensa3
    k=17
    j=17
    tomar=1
    conteomusic=1
    musica[0]="3"
end if

if (dato[0]=clave[7]) and (verificador=3) then
    mensaje=mensa4
    k=18
    j=18
    tomar=1
    conteomusic=1
    musica[0]="4"
end if

if ((dato[0]=ok[1])and(tomar=1)) then
'tomar=0
    j=100
    timeout=0
    tomar=2
end if
if (dato[0]=ok[0]) then
'tomar=1
    portb.3=1
    delay_ms(2000)
    timeout=0
    nuevo=1
end if
else
    timeout=timeout+1
end if

if timeout>250 then
    nuevo=0
    mensaje=""
    dato=""
    j=0
    timeout=0
    verificador=0
end if

    portb.3=0
wend

j=0
portb.2=1
delay_ms(500)
portb.2=0

'if ((mensaje[0]=compara1[0])and (envia=0)) then
if (envia=0) then

    k=k-1

    'if (conteomusic=1) then
    ' cambia1
    ' inimp3
    ' Soft_Uart_Write(dato1)
    ' finmp3
    '
    ' cambia1
    ' inimp3
    ' Soft_Uart_Write(dato1)
    ' finmp3

```

```

'
'   cambia1
'   inimp3
'   Soft_Uart_Write(dato1)
'   finmp3
'
' end if
'
' if (conteomusic=2) then
'   cambia1
'   cambia2
'   ini2mp3
'   Soft_Uart_Write(dato1)
'   Soft_Uart_Write(dato2)
'   finmp3
'
'   ini2mp3
'   Soft_Uart_Write(dato1)
'   Soft_Uart_Write(dato2)
'   finmp3
'
'   ini2mp3
'   Soft_Uart_Write(dato1)
'   Soft_Uart_Write(dato2)
'   finmp3
' end if

delay_ms(2000)

envia=1
portb.4=1
portb.5=1
delay_ms(3000)
portc.4=1
portc.5=1

if (k>0) and (k<10) then
  uno=k
  dos=0
  tres=0
end if

if (k>9) and (k<100) then
  uno=k mod 10
  dos=k/10
  tres=0
end if

if (k>99) and (k<250) then
  uno=k mod 10
  dos=k-uno
  momen=dos / 10
  dos=momen mod 10
  tres=momen /10
end if

' mensaje="fran "
' hacer
' portc.4=0
' portc.5=0

if (conteomusic=1) then
  cambia1
  inimp3
  Soft_Uart_Write(dato1)
  finmp3

```

```
delay_ms(1000)

cambia1
inimp3
Soft_Uart_Write(dato1)
finmp3

delay_ms(1000)

cambia1
inimp3
Soft_Uart_Write(dato1)
finmp3

end if

if (conteomusic=2) then
cambia1
cambia2
ini2mp3
Soft_Uart_Write(dato1)
Soft_Uart_Write(dato2)
finmp3

delay_ms(1000)

ini2mp3
Soft_Uart_Write(dato1)
Soft_Uart_Write(dato2)
finmp3

delay_ms(1000)

ini2mp3
Soft_Uart_Write(dato1)
Soft_Uart_Write(dato2)
finmp3
end if

delay_ms(5000)
portb.4=0
delay_ms(1000)
portb.5=0
delay_ms(2000)
end if

dato=""
mensaje=""
j=0
k=0
tomar=0
dato1=0
dato2=0
conteomusic=0
musica=""
paso=0
timeout=0
verificador=0
auto=0

wend

end.
```