

ESCUELA POLITÉCNICA NACIONAL

ESCUELA DE FORMACIÓN DE TECNÓLOGOS

CONSTRUCCIÓN DE UN BASTÓN ELECTRÓNICO PARA PERSONAS NO VIDENTES

PROYECTO PREVIO A LA OBTENCIÓN DEL TÍTULO DE TECNÓLOGO EN ELECTRÓNICA Y TELECOMUNICACIONES

ANALUISA YAGUANA ZORAIDA PAULINA (zpaypaulina@yahoo.es)

JARAMILLO POZO EDISSON JAVIER (xa_vas13@yahoo.es)

DIRECTOR: ING. PABLO LÓPEZ

pwlopezm@hotmail.com

QUITO, FEBERO 2011

DECLARACIÓN

Nosotros, Analuisa Yaguana Zoraida Paulina y Jaramillo Pozo Edison Javier declaramos bajo juramento que el trabajo aquí descrito es de nuestra autoría; que no ha sido previamente presentado para ningún grado o calificación profesional, y que hemos consultado las referencias bibliográficas que se incluyen en este documento.

A través de la presente declaración cedemos nuestros derechos de propiedad intelectual correspondiente a este trabajo, a la Escuela Politécnica Nacional, según lo establecido por la Ley de Propiedad Intelectual, por su Reglamento y por la normativa institucional vigente.

Zoraida Analuisa.

Edisson Jaramillo.

CERTIFICACIÓN

Certifico que el presente trabajo fue desarrollado por: Analuisa Yaguana Zoraida Paulina y Jaramillo Pozo Edison Javier, bajo mi supervisión.

Ing. Pablo López
DIRECTOR DEL PROYECTO

AGRADECIMIENTO

A Dios por darme la oportunidad de seguir realizado una de mis metas y permitirme compartir cada alegría con los seres que me rodean.

A mis padres que me han enseñado a no rendirme y siempre dar mi mejor esfuerzo para alcanzar mis sueños, gracias por su confianza y apoyo incondicional a lo largo de mi carrera.

A la Escuela Politécnica Nacional y a todos quienes la conforman ya que gracias a la formación y enseñanza que recibido de cada uno de ellos, hoy tengo la oportunidad de ejercer una profesión.

Al Ing. Pablo López por su acertada dirección y guía continua en el desarrollo del presente proyecto.

A todos mis amigos con los que he compartido gratos momentos y en especial a mi amigo y compañero de tesis Edison que por su constante apoyo durante el desarrollo del presente trabajo he logrado concluir una de mis etapas de estudio.

Paulina

DEDICATORIA

Con mucho cariño para mis padres Marcela y Alberto, quienes han estado a mi lado en todo momento y que con su cariño me han ayudado a llegar a cumplir una de mis metas.

Para mis hermanos quienes siempre me brindan su amistad y cariño en todo momento.

Paulina

DEDICATORIA

El presente proyecto está especialmente dedicado a mis padres Jesús y Judith a quienes amo y admiro porque gracias a sus consejos y apoyo incondicional he tenido la oportunidad de cumplir una de mis metas, ser un profesional.

De igual forma dedico el presente proyecto a mis hermanos Paul y Alan quienes siempre me han brindado su cariño y comprensión, y son una fuente importante de motivación en mi vida.

Edisson

AGRADECIMIENTO

Agradezco a Dios por darme la oportunidad de un nuevo día y darme la oportunidad de cumplir con mis metas.

Agradezco a mi familia que con su ejemplo, dedicación, confianza y apoyo, incondicional me enseñaron a dar mi mejor esfuerzo para alcanzar mis metas.

Agradezco a la Escuela Politécnica Nacional en especial a la carrera de Tecnología en Electrónica y Telecomunicaciones ya que gracias a la formación y enseñanzas que recibí me han permitido desarrollarme como persona y profesional.

Agradezco al Ingeniero Pablo López por su acertada dirección en el presente proyecto de titulación.

Agradezco a mi amiga y compañera por su dedicación y apoyo durante el desarrollo del presente proyecto.

Agradezco a mis amigos que han estado siempre presentes brindándome su amistad sincera y comprensión.

Edisson

ÍNDICE

ÍNDICE.....	VIII
ÍNDICE DE TABLAS.....	XIII
ÍNDICE DE FIGURAS.....	XIV
INTRODUCCIÓN.....	1
RESUMEN.....	2
CAPÍTULO I.....	3
CONCEPTOS GENERALES.....	3
1.1 AYUDAS A LA MOVILIZACIÓN DE LOS NO VIDENTES.....	3
1.1.1 BASTÓN BLANCO.....	4
1.1.2 EL PERRO-GUÍA, EL MEJOR AMIGO DEL INVIDENTE.....	5
Figura 1. 3 Sistema artificial de la visión de Dobbelle.....	6
1.1.3.1 Tecnología.....	7
1.1.3.2 Resultados.....	8
1.1.4 RETINA ARTIFICIAL.....	8
1.1.5 I-STICK: EL BASTÓN PARA CASOS DE EMERGENCIA.....	9
1.1.6 BASTÓN ELECTRÓNICO PARA CIEGOS.....	9
1.1.7 ROBOT PARA CIEGOS.....	12
1.1.8 EL MINI ULTRASÓNICO DE AYUDA DE MOVILIDAD.....	13
1.1.9 GAFAS CON GPS PARA CIEGOS.....	14
1.1.10 EL GUIDECANE (BASTÓN-GUÍA).....	16
1.1.11 TRAJE DE AYUDA PARA INVIDENTES.....	16
1.1.12 UNA VARA PARA CIEGOS.....	17
1.2 MICROCONTROLADORES.....	18
1.2.1 INTRODUCCIÓN ⁶	18
1.2.2 DEFINICIÓN ⁷	19
1.2.3 APLICACIONES DE LOS MICROCONTROLADORES.....	19
1.2.3 RECURSOS COMUNES DE LOS MICROCONTROLADORES ⁶	20

1.2.4 ARQUITECTURA BÁSICA ⁶	20
1.2.5 EL PROCESADOR O CPU ⁶	21
1.2.6 MEMORIA ⁶	22
1.2.7 LÍNEAS DE ENTRADA/SALIDA ⁶	22
1.2.8 RELOJ PRINCIPAL ⁶	23
1.2.8.1 Los microcontroladores admiten cuatro tipos de osciladores:.....	23
1.2.9 RECURSOS ESPECIALES ⁶	24
1.2.9.1 Temporizadores o "Timers" ⁶	24
1.2.9.2 Perro guardián o "Watchdog" ⁶	25
1.2.9.3 Estado de reposo o de bajo consumo ⁶	25
1.2.9.4 Conversor A/D (CAD) ⁶	26
1.2.9.5 Conversor D/A (CDA) ⁶	26
1.2.9.6 Comparador analógico ⁶	26
1.2.9.7 Modulador de ancho de pulso o PWM ⁶	26
1.3 MICROCONTROLADORES AVR.....	27
1.4 SENSORES DE DISTANCIA POR ULTRASONIDO.....	29
1.4.1 FUNCIONAMIENTO BÁSICO	29
1.4.4 SRF08 SENSOR DISTANCIAS POR ULTRASONIDOS	31
1.4.4.2 Control del sensor de distancia por ultrasonido SRF08	31
1.4.4.3 Conexiones.....	32
1.4.4.4 Registros	33
1.4.4.5 Comandos	34
1.4.4.6 Modo de cálculo de distancia con el SRF08	34
1.4.4.7 Modo ANN	35
1.4.4.8 Cómo comprobar que una medición ha finalizado.....	35
1.4.4.9 Cómo cambiar el rango de alcance	36
1.4.4.10 Ganancia analógica	37
1.4.4.11 Medidas	39
1.4.4.12 Sensor de luz	39
1.4.4.13 Led.....	40
1.4.4.14 Cambio de la dirección del bus I2C del SRF08.....	40
1.4.4.15 Consumo de corriente	41
1.4.4.16 Cambio del ángulo de detección.....	42
1.5. LCD.....	43
1.5.1 DEFINICIÓN	43
1.5.2 FUNCIONAMIENTO	45

1.5.3 LCD DE TEXTO	45
1.5.4 LCD DE GRÁFICOS	47
1.6 SECURE DIGITAL	47
1.6.1 MiniSD.....	48
1.6.2 MicroSD	49
1.6.3 MULTI MEDIA CARD.....	50
1.6.1.1 Especificaciones	51
CAPÍTULO II.....	52
CONSTRUCCIÓN DEL HARDWARE Y SOFTWARE	52
2.1 INTRODUCCIÓN.....	52
2.2 ARQUITECTURA DEL BASTÓN	52
2.2.1 FILOSOFÍA DE CONTROL DEL BASTÓN ELECTRÓNICO.....	52
2.2 DIAGRAMA DE BLOQUES DEL BASTÓN ELECTRÓNICO	54
2.3 NOMENCLATURA:	54
2.4 SISTEMA DE DETECCIÓN DE OBSTÁCULOS.....	54
2.4.1 INTRODUCCIÓN	54
2.4.2 ANÁLISIS DEL CIRCUITO	55
2.5 SISTEMA DE CONTROL “TARJETA PRINCIPAL”	56
2.6 FUNDAMENTO DEL SISTEMA ACTUADOR.....	56
2.6.1 SISTEMA DE REPRODUCCIÓN DE MENSAJES DE VOZ	56
2.6.2 INTRODUCCIÓN.	58
2.6.3 DESCRIPCIÓN DEL MÓDULO DE VOZ	58
2.6.3.1 Direccionamiento de la memoria SD	58
2.6.3.2 Amplificación del mensaje de voz.....	60
2.6.4 PROCESO DE OBTENCIÓN Y GRABACIÓN DE LOS MENSAJES	61
2.6.4.1 Obtención de los sonidos	61
2.6.4.1.1 Nomenclatura y contenido de los mensajes de voz.....	62
MENSAJES: SENSOR UNO.....	62
2.7 CONEXIÓN DE LOS DISPOSITIVOS EN LA TARJETA PRINCIPAL.....	63
2.7.1 CONEXIÓN DEL CIRCUITO RESET.....	63

2.7.2 CONEXIÓN DE OSCILADORES.....	65
2.7.3 CONEXIÓN DEL LCD.....	66
2.7.4 CONEXIÓN DE LOS SENSORES.....	67
2.7.5 CONEXIÓN ENTRE LOS DOS MICROPROCESADORES (ATmega 164 y ATmega 644).....	68
2.7.6 CARGADOR DE LA BATERÍA.....	68
2.7.6.1 Descripción	69
2.7.6.2 Montaje.....	70
2.7.7 REGULADOR DE VOLTAJE A 5V	71
2.7.8 DISEÑO DEL CIRCUITO ESQUEMÁTICO	72
2.8 PASOS PARA OBTENER EL CIRCUITO ESQUEMÁTICO	74
2.9 DISEÑO DE LAS PLACAS DEL CIRCUITO IMPRESO (PCB).....	77
2.10 CREACIÓN DE LOS CIRCUITO IMPRESOS	79
2.11 DESCRIPCIÓN DE LAS TARJETAS DEL BASTÓN	83
2.11.1 BASTÓN ELECTRÓNICO	83
2.11.2 TARJETA PRINCIPAL	85
2.11.2 MODULO DE VOZ.....	86
2.12 SOFTWARE DEL BASTÓN	87
2.12.1 DESARROLLO DEL SOFTWARE.....	87
2.12.2 PASOS PARA CREAR UN PROGRAMA.....	87
2.13 LENGUAJE DE PROGRAMACIÓN	88
2.13.1 BASCOM AVR SOFTWARE	88
2.13.2 FUNDAMENTOS DEL LENGUAJE	89
2.13.2.1 Línea de programa de BASCOM.....	90
2.13.2.2 Usando identificadores de línea.....	91
2.13.3 DECLARACIONES DE BASCOM	91
2.13.4 TIPOS DE DATOS	92
2.13.5 VARIABLES	93
2.13.5.1 Nombres de variables	93
2.13.6 EXPRESIONES Y OPERADORES	94
2.13.6.1 Operadores aritméticos.....	95
2.13.6.2 Operadores de relación	95
2.13.6.3 Operadores lógicos	96
2.13.7 OPERADORES FUNCIONALES	96

2.14 DIAGRAMAS DE FLUJO DEL PROGRAMA.....	99
2.11.1 SISTEMA DE CONTROL	99
4.11.2 MODULO DE VOZ	103
2.12 PROGRAMA DE LA TARJETA PRINCIPAL (ATMEGA 164)	104
CAPITULO 3.....	119
PRUEBAS Y RESULTADOS	119
3.1 PRUEBAS	119
3.1.1 FORMATO PARA PRUEBAS DE LOS DIFERENTES SISTEMAS	119
3.1.2 SISTEMA DE DETECCIÓN DE OBSTÁCULOS “SRF08”	120
3.1.3 SISTEMA DE CONTROL “TARJETA PRINCIPAL”	122
3.1.4 SISTEMA DE ACTUADORES “MODULO DE VOZ”	123
3.2 RESULTADOS	123
CAPITULO 4.....	124
CONCLUSIONES Y RECOMENDACIONES.....	124
4.1 CONCLUSIONES	124
4.2 RECOMENDACIONES	126
BIBLIOGRAFIA	128
ANEXOS	130

Índice de tablas

Tabla 1. 1 Característica de algunos microcontroladores utilizados en el medio.	29
Tabla 1. 2 Registros del SRF08	33
Tabla 1. 3 Comandos del SRF08	34
Tabla 1. 4 Modo Ann	35
Tabla 1. 5 Ganancia analógica.....	39
Tabla 1. 6 Dirección del bus I2C del SRF08	41
Tabla 1. 7 Consumo de corriente	42
Tabla 1. 8 Descripción Memorias.....	51
Tabla 2. 1 Nomenclatura y contenido de los mensajes de voz S1	62
Tabla 2. 2 Nomenclatura y contenido de los mensajes de voz S2.	63
Tabla 2. 3 Nomenclatura y contenido de los mensajes de voz.....	63
Tabla 2. 4 Caracteres especiales.....	90
Tabla 2. 5 Operadores de relación.....	96
Tabla 2. 6 Operadores lógicos	96
Tabla 3. 1 Formato de pruebas de los diferentes sistemas de sensores	120

Índice de Figuras

Figura 1. 1 El perro-guía	5
Figura 1. 2 El perro-guía	5
Figura 1. 3 Sistema artificial de la visión de Dobbie	6
Figura 1. 4 I-STICK	9
Figura 1. 5 Bastón electrónico para ciegos	9
Figura 1. 6 Robot para ciegos.	12
Figura 1. 7 El mini ultrasónico	13
Figura 1. 8 Gafas con GPS para ciegos.....	15
Figura 1. 9 Traje de ayuda para invidentes	17
Figura 1. 10 Una vara para ciegos	17
Figura 1. 11 Una vara para ciegos	18
Figura 1. 12 Arquitectura Von Neumann ⁸	20
Figura 1. 13 Arquitectura Harvard ⁸	21
Figura 1. 14 Funcionamiento de los ultrasonidos	30
Figura 1. 15 Sensor SRF08.....	31
Figura 1. 16 Conexiones del SRF08.	32
Figura 1. 17 Medidas del SRF08.....	39
Figura 1. 18 Ángulo de detección.....	42
Figura 1. 19 Imagen donde se pueden observar los píxeles1	45
Figura 1. 20 Display lcd de texto de 16x2 con controlador hd44780.....	46
Figura 1. 21 Detalle de una pantalla LCD en color.....	47
Figura 1. 22 Memoria SD.	47
Figura 1. 23 Memoria MiniSD y adaptador.....	48
Figura 1. 24 Tarjeta MicroSD (64MB) al lado de una moneda 1€	49
Figura 1. 25 Tarjeta MMC	50
Figura 2. 1 Diagrama de bloques global, “Sistema a lazo abierto”	53
Figura 2. 2 Diagrama de bloques del bastón electrónico.	54
Figura 2. 3 Análisis del circuito.....	55
Figura 2. 4 Diagrama del control y monitoreo del bastón serialmente.....	56

Figura 2. 5 Diagrama circuital del módulo de voz.....	57
Figura 2. 6 Direccionamiento de la memoria SD.....	59
Figura 2. 7 Amplificación del mensaje de voz	60
Figura 2. 8 EXPstudio audio editor.....	61
Figura 2. 9 EXPstudio audio editor formato de los mensajes voz	62
Figura 2. 10 Circuito reset (1).....	64
Figura 2. 11 Circuito reset (2).....	64
Figura 2. 12 Conexión de osciladores.....	65
Figura 2. 13 Conexión del LCD	66
Figura 2. 14 Conexión de los Sensores	67
Figura 2. 15 Conexión entre los dos microprocesadores	68
Figura 2. 16Cargador de batería	68
Figura 2. 17 Placa del cargador de batería.	70
Figura 2. 18 Circuito impreso del cargador de batería.	71
Figura 2. 19 Regulador de voltaje a 5V.....	72
Figura 2. 20 Circuito Esquemático.	73
Figura 2. 21 Primer paso a seguir.	74
Figura 2. 22 Ventana ISIS.....	74
Figura 2. 23 Ventana PICK DEVICES.....	75
Figura 2. 24 Ventana PICK DEVICES con elementos.....	75
Figura 2. 25 Ventana ISIS con elementos.....	76
Figura 2. 26 Circuito Esquemático en ISIS.....	76
Figura 2. 27 Modulo de voz en ares (1).....	77
Figura 2. 28 Modulo de voz en ares (2).....	77
Figura 2. 29 Módulo de voz en ares (3).....	78
Figura 2. 30 Diseño de impresión.....	78
Figura 2. 31 Diagrama del circuito impreso placa principal.....	80
Figura 2. 32 Screen de los elemento placa principal.....	80
Figura 2. 33 Diagrama posicional de los elementos placa principal.....	81
Figura 2. 34 Baño de cloruro férrico.....	82
Figura 2. 35 Tarjeta principal sacada del acido y lavada.....	82
Figura 2. 36 Screen de elementos placa cerebro.....	83
Figura 2. 37 Bastón vista frontal.....	84

Figura 2. 38 Bastón vista lateral.	84
Figura 2. 39 Tarjeta principal.....	85
Figura 2. 40 Parte frontal del modulo de voz.....	86
Figura 2. 41 Parte posterior del modulo de voz.....	86
Figura 2. 42 Diagrama de bloques de pasos para programar un microcontrolador.....	87
Figura 2. 43 Pasos para grabar el programa realizado en el microcontrolador	88
Figura 2. 44 Imagen del icono del programa BASCOM-AVR.....	88
Figura 2. 45 BASCOM.....	89
Figura 2. 46 Diagrama de Flujo del sistema de control	99
Figura 2. 47 Diagrama de Flujo del sistema de control.	100
Figura 2. 48 Diagrama de Flujo del sistema de control.	101
Figura 2. 49 Diagrama de Flujo del sistema de control.	102
Figura 2. 50 Diagrama de Flujo del modulo de voz.	103

INTRODUCCIÓN

El bastón electrónico para personas no videntes consta de un cargador de batería de 12V, una estructura física de madera para su mejor movilidad por su reducido peso y con la ayuda de un soporte de cinco ruedas incorporadas a esta estructura, se colocaron tres sensores de distancias por ultrasonido, uno de los sensores está ubicado en la parte frontal superior del bastón a 65 cm el cual detectará cuando exista la presencia de obstáculos más altos o grandes a una distancia máxima de 2m, el segundo se colocó a una altura de 5.5 cm para detectar gradas y el tercer sensor que se encuentra colocado en la parte baja del bastón a 5 cm de altura, este es utilizado para detectar la presencia de inclinaciones y agujeros si existiese el caso, estas distancias serán visualizadas en un LCD, el cual indicará los valores de los tres sensores, este LCD fue colocado como forma de ayuda para la persona que realice el mantenimiento a este bastón y para visualizar el buen funcionamiento de los sensores.

La información que es recolectada por los tres sensores es almacenada en el microcontrolador AVR ATMEGA 164 de la placa principal, en esta es comparada y dependiendo del valor seleccionado por el programa que en este caso sería, el que tenga un mayor acercamiento entre el obstáculo y la persona no vidente se enviarán instrucciones al microcontrolador AVR ATMEGA 644 que se encuentra en la placa del modulo de voz el cual buscará en la memoria SD los mensajes de aviso que serán reproducidos con la ayuda de audífonos que se colocaron en el bastón.

Los audífonos son para que la persona no vidente use constantemente por el ruido del medio físico el cual evita que la persona no vidente identifique el aviso de peligro, estos mensajes fueron previamente grabados y editados en un computador para posteriormente guardados en una memoria SD esperando ser seleccionados por el microcontrolador de la tarjeta del módulo de voz. Y dependiendo de la distancia estos mensajes son reproducidos avisando a la persona no vidente de la cercanía del obstáculo.

RESUMEN

EL presente trabajo desarrolla el prototipo de un bastón electrónico para personas no videntes mediante el uso de sensores de distancia por ultrasonido.

Para una adecuada comprensión de la función que debe cumplir el dispositivo a desarrollar se buscará conocer sobre algunas de las ayudas existentes para la movilización de las personas no videntes.

Para el desarrollo de este proyecto se procedió a realizar una investigación de las ayudas existentes que estén relacionadas con: ayudas al no vidente mediante sensores de distancia por ultrasonido, los ya existentes que se encontraron tienen costos demasiado elevados y todavía algunos no han salido al mercado.

Se diseñó el diagrama lógico del sistema y se establecieron tres partes: la tarjeta principal, el módulo de voz y el cargador de batería de 12V del bastón electrónico para ciegos, estos circuitos fueron probados en un protoboard después de haber tenido sus respectivos aciertos, dificultades y pruebas realizamos el impreso de las placas, las cuales por el tamaño del bastón y el lugar donde debían ser colocadas las realizamos en forma independiente, en esta etapa existieron dificultades ya que siempre se presentan problemas en el funcionamiento cuando los circuitos se encuentran ya en la placa, también se tuvo dificultad en la ubicación de los sensores los cuales por tener sus respectivos lóbulos de radiación se interferían mutuamente y no permitían dar un valor exacto y por esto la modificación de la parte física del bastón la cual cambió bastante de la idea original. Existieron problemas con la reproducción de sonido el cual está encargado por el módulo de voz que consta de un AVR, los cuales fueron superados después de varias pruebas.

Finalmente se concluyó el proyecto "Bastón electrónico para personas no videntes" con las respectivas pruebas y superando las dificultades descritas las cuales nos permitieron presentar el presente proyecto.

CAPÍTULO I

CONCEPTOS GENERALES

1.1 AYUDAS A LA MOVILIZACIÓN DE LOS NO VIDENTES

Las personas invidentes físicamente no tienen impedimento para caminar, pero al carecer del sentido de la vista sienten inseguridad y temor al movilizarse, esto afecta a su habilidad para establecer relación con los objetos y su capacidad de representación mental del espacio.

Por mucho tiempo se han realizado estudios en diferentes lugares con el objeto específico de dotar al no vidente de instrumentos y técnicas que le permitan facilitar su movilización.

Los diferentes instrumentos y técnicas se basan en proveer información al vidente para que aproveche los sentidos que posee, cada sentido brinda características especiales que pueden ser aprovechadas como por ejemplo:

El sentido auditivo, que proporciona información sobre distancia, orientación, dirección, tamaño y peligro.

El sentido del tacto, que brinda sensaciones de variaciones de presión, temperatura y con adecuada estimulación brindara sensaciones de orientación y memoria muscular.

El sentido del olfato, que permite obtener información sobre distancia, orientación y diferenciación.

Existe una variedad de aparatos de orientación y movilidad para los individuos que padecen de incapacidades de la vista. La mayoría de estos aparatos son "de baja tecnología" porque son aparatos simples, típicamente el bastón. Estos son aparatos que la mayoría de incapacitados de la vista usan por dos buenas razones:

Su precio es relativamente barato y son fáciles de mantener.

Su fácil uso (especialmente el bastón) y adiestrar a los perros guías.

Existen otros aparatos de "alta tecnología" y menos conocidos que existen.

A continuación se describirá algunas de las ayudas existentes para los no videntes desarrolladas en diversos centros de investigación.

1.1.1 BASTÓN BLANCO

El Bastón blanco o "Bastón blanco para ciegos" es símbolo del logro e independencia de las personas discapacitadas visuales que no necesariamente son ciegos totales, e instrumento para su orientación y movilidad.

El bastón blanco, auxiliar principal en la de ambulación, es también un símbolo, un cartel enorme que dice "Soy ciego, cuidado!" En esta doble función, para ver y ser visto se relaciona con la ambivalencia que, como tal, despierta: Indispensable para percibir, "tercer ojo" insustituible. Es más que una prótesis: es una prolongación del yo.

Sobre todo para los que han perdido la visión recientemente, o para los disminuidos visuales no tan severos, esa sensación de dependencia se substituye o se complementa con un profundo rechazo: aún personas cuya visión residual es ya inútil para desplazarse, siguen negándose a usar bastón argumentando que "Eso es para los ciegos" "Yo no lo necesito, yo veo" Esto puede dar lugar a situaciones de peligro, percibidas o no como tales.

Cuando el reconocimiento de la ambivalencia es mayor, y, por lo tanto, menor el nivel de negación, se escuchan frases como: "Yo sé que debería usarlo, por seguridad, pero siempre me olvido" "No sé cómo todavía no me maté" "No me animo a salir solo" "Me van a mirar raro, van a pensar que me hago el ciego". Posibles soluciones, para una aceptación paulatina, es llevarlo, pero doblado, usarlo solo de noche, o usarlo "poniendo cara de ciego", usar un paraguas en vez de bastón... Es común, cuando ya se ha adoptado, escuchar "No sé porqué

no me decidí antes.", "fue un alivio para mí." (Aunque es posible que, después de esa aceptación entusiasta, el pobre bastón sea 'olvidado' la vez siguiente.)

1.1.2 EL PERRO-GUÍA, EL MEJOR AMIGO DEL INVIDENTE



Figura 1. 1 El perro-guía

Popularmente se les conoce como 'lazarillos' y algunas veces les vemos ejerciendo su trabajo en el transporte público o ante un paso de peatones.

Estos perros adquieren, gracias a un esmerado adiestramiento, el papel de ojos del que no puede ver. En su entrenamiento participan especialistas que no sólo le adiestran para su posterior asignación a una persona ciega, sino que también les transmiten valores como el cariño, la comprensión y la ternura.



Figura 1. 2 El perro-guía

El punto de partida de la historia del perro guía puede marcarse en la fundación de un instituto para ciegos en Viena por Herré Johann Whilhelm

Klein en 1819. Las ideas de Kleim fueron extraídas de la vida de un hombre ciego llamado Joseph Resinguer que enseñó a tres perros para que le ayudaran en sus labores cotidianas. Por fin, el proyecto de Klein inspiró, en 1916, al doctor Gerhard Stalling, que estableció una escuela de entrenamiento para perros con el objetivo de adjudicarlos a excombatientes alemanes de la Primera Guerra Mundial que habían perdido su facultad visual.

1.1.2.1 Características del perro lazarillo

Las primeras prácticas en adiestramiento tuvieron como referente al Pastor Alemán, puesto que era la raza más popular en las tierras teutonas. Actualmente se usan mucho, además del Pastor Alemán, el Labrador y el Golden Retriever.

En cuanto al sexo, no hay diferencias entre machos y hembras, siendo ambos aptos para convertirse en el mejor instrumento de movilidad de una persona incapaz.

Los criadores intentan conseguir la genética perfecta por medio de cruces para, después, formar definitivamente al can. Físicamente, los perros-guía deben guardar unas condiciones perfectas, no superando nunca la talla mediana. Rasgos de carácter como la obediencia, la serenidad, sociabilidad y un buen sentido del equilibrio, en otras palabras deben convertirse en el mejor amigo.

1.1.3 OJO DOBELLE

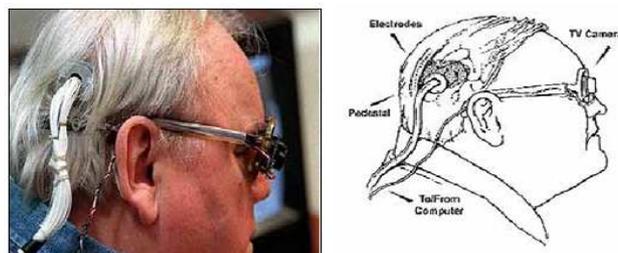


Figura 1. 3 Sistema artificial de la visión de Dobelle

La visión artificial describe un sistema de técnicas diseñadas para restaurar una pérdida severa de la visión, substituyendo un defecto a lo largo del camino visual.

Varios modelos se han propuesto, incluyendo las prótesis secundario-retinianas (diseñadas para sustituir foto receptores en la retina), las prótesis epi-retinianas (diseñadas para comunicarse directamente con el ganglio y las células bipolares), estímulo óptico del nervio, y las prótesis corticales (diseñadas para estimular la corteza visual directamente para los pacientes sin un nervio óptico viable).

1.1.3.1 Tecnología

El dispositivo, en este caso, se implanta en el interior del cráneo, sobre la superficie del lóbulo occipital del cerebro.

Una descripción de las prótesis implantadas describe a un sensor de distancia por ultrasonido que se coloca en un par de gafas que es usada por el paciente. Los sensores están conectados con una computadora miniatura. La computadora procesa las señales del vídeo y de la distancia, reduce el ruido, y acciona un microordenador para transmitir pulsos a un arsenal de electrodo de platino 68 implantado en la superficie de la corteza visual.

El paquete externo de la computadora tiene el tamaño de un diccionario y pesa diez libras, la batería de la computadora cerca de 3 horas. Requiere que se cargue la batería cada día, y el entrenamiento del usuario para el manejo del mismo. El propósito del sistema artificial de la visión de Dobbelle es promover movilidad independiente, pero no para la lectura. Es contraindicado para las personas con infecciones crónicas severas, porque les afecta directamente al cerebro.

1.1.3.2 Resultados

Los resultados de la implantación de este dispositivo fueron descritos en los años sesenta, y se realizó hasta 1972. Describen al paciente con una pérdida de la visión en un ojo a la edad de 22 años y en el otro ojo debido a trauma a la edad de 36 años. No se divulgó ningún problema o infecciones con la implantación. El paciente podría contar los dedos y alcanzó una agudeza 20/1200 visual. El paciente podría caminar por lugares desconocidos.

Este proyecto supone el trabajo de más de 300 científicos, ingenieros y médicos y se espera que el sistema pueda suministrar la cura para la mayoría de los tipos de ceguera en un futuro. El objetivo es el mismo, pero el camino para llegar es diferente.

El equipo de Dobbie implantó el dispositivo, hasta ahora, a ocho pacientes de seis países. En algunos casos, los seguros médicos cubrieron parte del costo de la operación, pero otros tuvieron que desembolsar los 115.000 dólares de su propio bolsillo. Eso sí, los no videntes que se lo implantaron debieron viajar a Portugal para realizar la operación. Por ahora, la Administración de Medicamentos y Alimentos estadounidense (FDA) pone muchísimas restricciones al implante de dispositivos en el cerebro.

1.1.4 RETINA ARTIFICIAL

Es la única que no utiliza ninguna cámara, sino un pequeñísimo chip que se coloca en la retina del no vidente. El microprocesador mide dos milímetros de diámetro (más pequeño que la cabeza de un alfiler) y contiene unas cinco mil pequeñas células solares, cada una de las cuales están unidas a un electrodo minúsculo.

La idea es que la luz que llegue al chip (a través del ojo) active los electrodos y así estimule los nervios ópticos, que luego transportarán la señal al cerebro. Muchos especialistas critican este método porque consideran que las células solares difícilmente generen suficiente energía eléctrica como para activar los nervios.

1.1.5 I-STICK: EL BASTÓN PARA CASOS DE EMERGENCIA

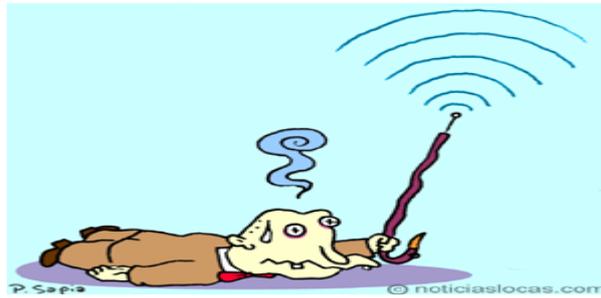


Figura 1. 4 I-STICK

El I-Stick es un bastón que se puede comunicar con un centro de urgencias en caso de detectar peligro después de un accidente.

Siguiendo la onda de iPod y de iTunes, los profesionales del Instituto de Desarrollo Experimental de Software de Fraunhofer inventaron el "I-Stick".

Su funcionamiento es simple: el aparato registra si está en posición horizontal y, en caso contrario, le ordena a su dueño que se levante.

Si éste no puede colocar el aparato en la posición indicada, llama en forma directa a una ambulancia o puede configurarse para que llame a un familiar. El I-Stick es parte de un proyecto que persigue usar la tecnología para ayudar a las personas ancianas. Aunque la personas ancianas no deseen saber nada con la tecnología.

1.1.6 BASTÓN ELECTRÓNICO PARA CIEGOS



Figura 1. 5 Bastón electrónico para ciegos

Mide la distancia de los objetos mediante láser y advierte al usuario con sonidos o vibraciones.

El clásico bastón blanco de los ciegos será sustituido por un bastón electrónico que se apoya en los avances de la ciencia en materia de miniaturización y tecnología láser para mejorar las condiciones de vida de las personas aquejadas de esta discapacidad. El bastón electrónico mide la distancia de los objetos mediante rayos láser y advierte al usuario a través de sonidos y vibraciones. Constituye un eficaz complemento del bastón tradicional, tiene el tamaño de un mando de televisión y pronto se apoyará en satélites para orientar en sus desplazamientos a los invidentes.

Un bastón electrónico para ciegos, que mide la distancia de los objetos mediante rayos láser y transmite la información correspondiente al usuario a través de sonidos o vibraciones emitidos por un pequeño dispositivo del tamaño de un mando de televisión que se lleva adherido a la mano, comienza a implantarse después de haber pasado dos años en fase de experimentación.

El bastón electrónico tiene externamente la misma apariencia que el bastón blanco alargado empleado por los ciegos para orientarse en sus desplazamientos por entornos abiertos, con la diferencia de que cuanto más se aproxima el usuario a un obstáculo, más intensa es la señal emitida por el aparato.

El bastón electrónico no sustituye al tradicional, sino que más bien lo complementa, ya que potencia la capacidad de esta herramienta para evitar los muros, anticipar el comienzo de la acera o descubrir un bache en el firme, e incluso señalar a los videntes la proximidad de una persona ciega.

Hay dos modelos del bastón electrónico: uno denominado Tom Puce (pequeño), que detecta objetos a cuatro metros, alerta mediante vibraciones y cuesta 762 euros.

El otro modelo, más sofisticado y caro (2.286 euros), se llama Tele tacto y describe el espacio con mayor amplitud (hasta 15 metros) y precisión, ya que el barrido del láser permite detectar los perfiles y reconocer las formas.

El bastón ha sido creado por el físico e investigador del CNRS y de la Universidad Paris-Sud/Orsay, René Farsi, quien ha dedicado ocho años de su vida a perfeccionar esta tecnología, probada ya con 60 invidentes franceses, y que continúa desarrollándose con diferentes variables nuevas.

El bastón electrónico no es de fácil manejo, ya que necesita un período de formación que cuesta 1.800 euros, que se suman al precio del producto. Lo más complicado es que los usuarios aprendan a interpretar las señales emitidas por el dispositivo y a construir referencias a partir de estos símbolos.

La eficacia comprobada en el período experimental es del 60%, pero el hecho de que el 40% de los que lo prueban no consigan orientarse no deja de ser un obstáculo para su plena implantación social. De ahí que los objetivos de sus creadores sean todavía modestos: en Francia hay 300.000 ciegos de los que 50.000 utilizan bastón.

De ellos, 10.000 personas no pueden prescindir de este apoyo para su vida cotidiana y lo que pretenden los inventores del bastón electrónico es que en el plazo de diez años 5.000 ciegos se beneficien de esta tecnología.

La mayoría de los usuarios prefiere el modelo más simple y económico, pero los más capaces (8 de los 60) encuentran un completo soporte a su movilidad: consiguen que el oído supla a la visión en la orientación sobre el entorno, permitiendo al usuario deducir los comportamientos de locomoción más adecuados.

El proyecto sin embargo es más ambicioso, ya que pretende incorporar al bastón electrónico el soporte de localización geográfica vía satélite (GPS), con la finalidad de que el usuario pueda tener todas las referencias de un entorno

para orientarse, de la misma forma que hoy lo consigue un conductor para llegar a un destino desconocido.

1.1.7 ROBOT PARA CIEGOS



Figura 1. 6 Robot para ciegos.

Maggie es un "humanoide" capaz de sentir cosquillas, de bailar o de dar las últimas noticias. La presentó su creador, el catedrático de Ingeniería de Sistemas y Automática de la Universidad Carlos III de Madrid, Miguel Ángel Salichs, durante el II Congreso Internacional sobre Domótica, Robótica y Teleasistencia para todos.

Según sus creadores, es el primer prototipo europeo capaz de reconocer la voz, de hablar o de moverse, gracias a sus sensores de movimiento.

Tiene además una pantalla táctil desde la que se le pueden dar órdenes, y es capaz, a través de su conexión a internet, de informar sobre el tiempo o distinguir, mediante un lector de infrarrojos, los medicamentos, su composición y uso.

Posee cámaras de TV y tecnología láser. "Aunque en el futuro los hombres contaremos con la ayuda de robots similares a los de la ciencia ficción, se trata de una realidad que está todavía demasiado lejana", dijo Salichs en el

congreso organizado por la Fundación Once, dedicada a personas con discapacidad.

El profesor considera que los hogares, entre otras cosas, se podrán automatizar. Pero también hay dificultades en la aplicación de la robótica: la manipulación de objetos, la percepción, el tacto, la locomoción y la autonomía, que no sobrepasa una hora. Entre las ventajas estarían la ayuda a personas mayores, a discapacitados y a la industria.

1.1.8 EL MINI ULTRASÓNICO DE AYUDA DE MOVILIDAD



Figura 1. 7 El mini ultrasónico

El MINI es un aparato que se utiliza para el desplazamiento de las personas no videntes con una versatilidad en su forma de uso, debido a la facilidad de direccionamiento ya sea hacia el piso, al frente, izquierda, derecha.

El MINI es sólo como un accesorio a la más tradicional de las ayudas como el bastón blanco y el perro guía. El MINI no es un sustituto de estas ayudas primarias, que sólo debería usarse como una ayuda.

El MINI usa el eco mediante ultrasonidos para detectar objetos. Vibra La ayuda para indicar la distancia a los objetos.

Es más rápido de la vibración cuando esta cerca del objeto. También hay una toma de auriculares que puede utilizarse para proporcionar información.

La ayuda utiliza la última tecnología de la electrónica para proporcionar una excelente robustez, fiabilidad y duración de la batería. La batería suele durar muchos meses.

Un único botón se utiliza para cambiar de la ayuda o desactivar y también cambiar la configuración. La ayuda tiene varios modos y opciones. Las principales modalidades son:

- 4 metros
- 2 metros
- 1 metro
- medio metro
- 8 metros

Objetos grandes sólo pueden ser detectados a 4 metros o más allá, por ejemplo, cercas, paredes.

Los usuarios de perros guía han encontrado que el MINI ha ayudado:

- Evitando los obstáculos, tales como vehículos, postes y mobiliario urbano.
- Detectar los obstáculos que domina como las ramas de los árboles.
- La localización de los empleados y el fin de las colas.
- La localización de las puertas, determinar si se encuentran abiertas las puertas de ascensor.
- Navegando alrededor de mesas, sillas y entornos de oficina.

1.1.9 GAFAS CON GPS PARA CIEGOS

Aunque parece un casco de albañil del futuro, se trata de un invento que va a permitir mayor independencia y movilidad a los ciegos. Desde la Universidad de Valencia nos llega este avanzado aparato que dispone de rayos láser, video y hasta de GPS para situar y ubicar a los invidentes mediante sonidos que le descubren donde existen obstáculos. Y de paso, pueden entrar en una obra sin infringir la ley de prevención de riesgos laborales.



Figura 1. 8 Gafas con GPS para ciegos

La UPV (Universidad Politécnica de Valencia) presenta un prototipo que permitirán a los ciegos sortear obstáculos. Se basa en visión artificial, análisis acústico, sensores y sistemas GPS.

La UVP (Universidad Politécnica de Valencia) ha coordinado el desarrollo de sendos prototipos que utilizan metodologías basadas en visión artificial, análisis acústico, sensores de información 2D y sistemas de posicionamiento GPS que permitirán a los ciegos sortear obstáculos y moverse tanto en espacios abiertos como cerrados.

Concretamente se han creado dos prototipos, ya patentados, uno de ellos unas gafas que emiten un rayo láser que permite localizar objetos por sonidos a una distancia de cinco metros, y el otro, con un alcance de quince metros, basado en un sistema de estereovisión con dos cámaras ubicadas en un casco.

En este proyecto, llamado CASLiP (Sistema de Ayuda Cognitiva para Invidentes, en sus siglas en inglés), participan desde hace tres años siete centros europeos orientados al desarrollo de ayudas cognitivas para personas con discapacidad.

CASLiP "nació con el objetivo de diseñar una herramienta tecnológica que permita a las personas invidentes la integración en la vida social mejorando su

calidad de vida y aportándoles herramientas adicionales a las que ya poseen, como son los bastones y perros", explicó Guillermo Peris, investigador de la UPV

1.1.10 EL GUIDECANE (BASTÓN-GUÍA)

Las personas ciegas también están recibiendo ayuda de ciertos sensores que son cada vez más pequeños y también de fiar. En la Universidad de Michigan, los sensores que miden la distancia del objeto más próximo a una persona superan en rendimiento al tradicional bastón blanco.

El Guide Cané (bastón-guía), que pesa unos tres kilos y medio, tiene un pie motorizado y maniobrable con sensores ultrasónicos en cada lado. Los usuarios oprimen un pequeño joystick colocado cerca del mango para avisarle al bastón hacia dónde desean ir. El bastón obedece, guiando al usuario y esquivando los obstáculos.

"Uno tiene la sensación de estar paseando el perro", dice Johann Borenstein, un científico que forma parte del equipo de investigación y que dirige el Laboratorio de Robótica Móvil, Ingeniería Mecánica y Mecánica Aplicada. "Si el perro tira para un lado, uno no puede evitar seguirlo. El Guide Cane lleva tres años en proceso experimental, pero los sensores aún no son lo suficientemente confiables para uso general.

1.1.11 TRAJE DE AYUDA PARA INVIDENTES

El laboratorio Ishikawa-Namiki Komuro está trabajando en un traje que permitiría "ver" a los invidentes.

El proyecto (llamado "radar táctil/piel extendida") consiste en el desarrollo de un traje con una serie de sensores que detectan los objetos del entorno. Esta información se transmite mediante vibraciones en la superficie del traje, que el usuario recibe e interpreta. Algo similar, según la página del proyecto, al sistema que utilizan las antenas de los insectos o los bigotes de los gatos.

El proyecto no sólo va dirigido a invidentes, sino también a aquellos que trabajan en entornos peligrosos y como un detector de obstáculos para conductores (en este caso los sensores cubrirían la superficie del vehículo).

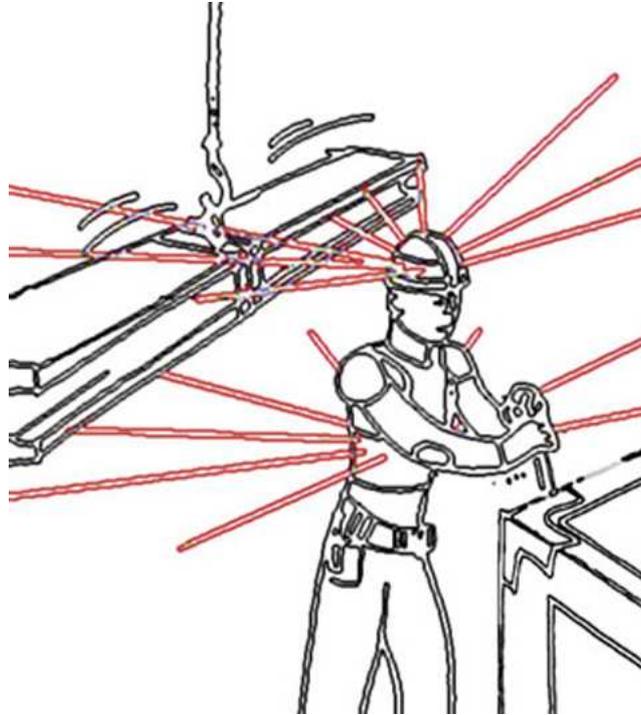


Figura 1. 9 Traje de ayuda para invidentes

1.1.12 UNA VARA PARA CIEGOS



Figura 1. 10 Una vara para ciegos

Cuando la ciencia hace aportes a enfermedades o padecimientos de algún tipo es una doble celebración, y en este caso tenemos a la tecnología ayudando a los ciegos a moverse con más facilidad y evitando los obstáculos.

Diseñado por Woo Jin Han esta vara para ciegos hace cierta referencia al clásico bastón blanco, sólo que un poco de tecnología que permite detectar los obstáculos sin necesidad de golpearlos.

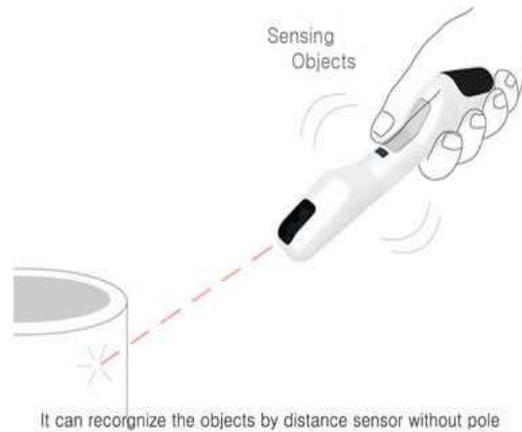


Figura 1. 11 Una vara para ciegos

El tubo tiene sensores incorporados que miden la distancia de los objetos y avisan por medio de vibraciones al portador de su cercanía, para que los evite.

1.2 MICROCONTROLADORES

1.2.1 INTRODUCCIÓN ⁶

Los microcontroladores están presentes en nuestro trabajo, en nuestra casa y en nuestra vida, en general. Se pueden encontrar controlando el funcionamiento de los mouse y teclados de los computadores, en los teléfonos, en los hornos microondas, los televisores y en general en los electrodomésticos de nuestro hogar. Pero en la actualidad esto acaba de comenzar y el nacimiento del siglo XXI será testigo de la conquista masiva de estos diminutos computadores, que gobernarán la mayor parte de los aparatos que fabricaremos y usamos.

⁶ www.monografias.com/trabajos12/microco/microco.shtml

1.2.2 DEFINICIÓN⁷

Un microcontrolador es un circuito integrado o chip que incluye en su interior las tres unidades funcionales de un ordenador: CPU, memoria y unidades de E/S, es decir, se trata de un computador completo en un solo circuito integrado. Aunque sus prestaciones son limitadas, además de dicha integración, su característica principal es su alto nivel de especialización. Aunque los hay del tamaño de un sello de correos, lo normal es que sean incluso más pequeños, ya que, lógicamente, forman parte del dispositivo que controlan.

1.2.3 APLICACIONES DE LOS MICROCONTROLADORES

Cada vez existen más productos que incorporan un microcontrolador con el fin de aumentar sustancialmente sus prestaciones, reducir su tamaño y costo, mejorar su fiabilidad y disminuir el consumo.

Algunos fabricantes de microcontroladores superan el millón de unidades de un modelo determinado producidas en una semana. Este dato puede dar una idea de la utilización masiva de estos componentes.

Los microcontroladores están siendo empleados en multitud de sistemas presentes en nuestra vida diaria, como pueden ser juguetes, horno microondas, frigoríficos, televisores, computadoras, impresoras, módems, el sistema de arranque de vehículos, etc. Y otras aplicaciones con las que seguramente no estaremos tan familiarizados como instrumentación electrónica, control de sistemas en una nave espacial, etc.

Una aplicación típica podría emplear varios microcontroladores para controlar pequeñas partes del sistema.

⁷ <http://es.wikipedia.org/wiki/Microcontrolador>

1.2.3 RECURSOS COMUNES DE LOS MICROCONTROLADORES⁶

Al estar todos los microcontroladores integrados en un chip, su estructura fundamental y sus características básicas son muy parecidas. Todos deben disponer de los bloques esenciales procesador, memoria de datos y de instrucciones, líneas de E/S, oscilador externo y módulos controladores de periféricos. Sin embargo, cada fabricante intenta enfatizar los recursos más idóneos para las aplicaciones a las que se destinan.

A continuación se detallan todos los recursos que se hallan en todos los microcontroladores describiendo las diversas alternativas y opciones que pueden encontrarse según el modelo seleccionado.

1.2.4 ARQUITECTURA BÁSICA⁶

Aunque inicialmente todos los microcontroladores adoptaron la arquitectura clásica de Von Neumann, actualmente se impone la arquitectura Harvard.

La arquitectura de Von Neumann Figura 1.12, se caracteriza por disponer de una sola memoria principal donde se almacenan datos e instrucciones de forma indistinta. A dicha memoria se accede a través de un sistema de buses único (direcciones, datos y control). Los microcontroladores de Intel 8751 usan esta arquitectura.

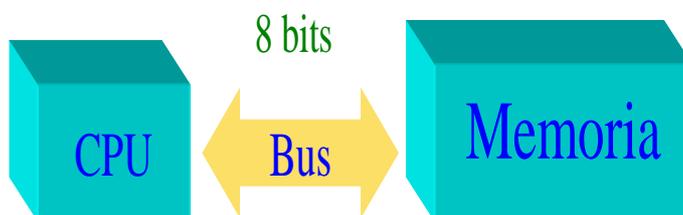


Figura 1. 12 Arquitectura Von Neumm⁸

⁶ www.monografias.com/trabajos12/microco/microco.shtml

⁸ COSTALES, Alcívar, Apuntes de microcontroladores

La arquitectura Harvard Figura 1.13, dispone de dos memorias independientes una que contiene sólo instrucciones y otra sólo datos. Ambas disponen de sus respectivos sistemas de buses de acceso y es posible realizar operaciones de acceso (lectura o escritura) simultáneamente en ambas memorias.

Los microcontroladores PIC (Peripheral Interface Controller) responden a la arquitectura Harvard.

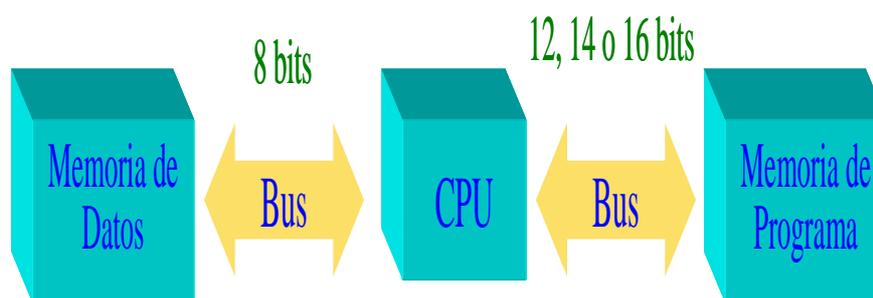


Figura 1. 13 Arquitectura Harvard ⁸

En las dos configuraciones, la CPU contiene a la ALU (Unidad Aritmética Lógica), que realiza las operaciones lógicas y aritméticas.

1.2.5 EL PROCESADOR O CPU⁶

Es el elemento más importante del microcontrolador y determina sus principales características, tanto a nivel hardware como software. Se encarga de direccionar la memoria de instrucciones, recibir el código de la instrucción en curso, su decodificación y la ejecución de la operación que implica la instrucción, así como la búsqueda de los operandos y el almacenamiento del resultado.

Otra característica importante es que el procesador es de tipo RISC (Juego Reducido de Instrucciones para Computadores). En estos procesadores el repertorio de instrucciones máquina es muy reducido y las instrucciones son simples y generalmente se ejecutan en un ciclo de reloj. La sencillez y rapidez

de las instrucciones permiten optimizar el hardware y el software del procesador.

Además se introduce una segmentación en el procesador, dividiendo la ejecución de una instrucción en varias etapas. De esta forma se puede trabajar sobre varias instrucciones simultáneamente cada una en una etapa distinta.

1.2.6 MEMORIA⁶

Memoria de programa: Es una memoria de almacenamiento no volátil (ROM, EPROM, OTP, EEPROM, FLASH), en la que se almacena el programa que gobierna la aplicación a la que está destinado el microcontrolador. No existen dispositivos de almacenamiento masivo por lo que todo el código debe estar almacenado en esta memoria. Por otro lado, al ser un circuito dedicado a una sola tarea debe almacenar un único programa.

En general tiene un tamaño muy reducido desde 512 a 16K palabras en la familia de los microcontroladores PIC. Además al tener una memoria específica de programa se puede adecuar el tamaño de las instrucciones y los buses al más adecuado para cada aplicación.

Memoria de datos: La memoria para almacenar datos debe ser de lectura y escritura, por lo general se usa memoria SRAM, aunque algunos microcontroladores llevan memoria EEPROM para evitar la pérdida de los datos en caso de corte en el suministro de energía. Los tamaños son mucho más reducidos que la memoria de programa.

1.2.7 LÍNEAS DE ENTRADA/SALIDA ⁶

La principal utilidad de los pines que posee la cápsula que contiene un microcontrolador es soportar las líneas de E/S que comunican al computador interno con los periféricos exteriores.

Según los controladores de periféricos que posea cada modelo de microcontrolador, las líneas de E/S se destinan a proporcionar el soporte a las

señales de entrada, salida y control. Manejan la información en paralelo y se agrupan en conjuntos que reciben el nombre de puertos o pódicos. Los pines de los pódicos pueden ser todos configurados de acuerdo a la necesidad de la aplicación, es decir, que los pines de un mismo puerto pueden ser usados unos como entradas y otros como salidas.

Además, algunos pines E/S de los puertos son multiplexados a una función alternativa de características periféricas. En general, cuando una función de estas es habilitada, ese pin tal vez no pueda ser usado como un pin de E/S.

1.2.8 RELOJ PRINCIPAL⁶

Todos los microcontroladores disponen de un circuito oscilador que genera una onda cuadrada de alta frecuencia, que configura los impulsos de reloj usados en la sincronización de todas las operaciones del sistema.

Generalmente, el circuito de reloj está incorporado en el microcontrolador y sólo se necesitan unos pocos componentes exteriores para seleccionar y estabilizar la frecuencia de trabajo.

1.2.8.1 Los microcontroladores admiten cuatro tipos de osciladores:

- Oscilador RC: oscilador de bajo costo formado por una resistencia y un condensador, cuyos valores determinan la frecuencia de oscilación. Proporciona una estabilidad mediocre.
- Oscilador HS: basado en un cristal de cuarzo, alcanza una velocidad entre 4 y 12 MHz
- Oscilador XT: oscilador de cristal o resonador para frecuencias entre 100 MHz y 4 MHz
- Oscilador LP: oscilador de bajo consumo con cristal o resonador para frecuencias entre 35 y 200 KHz.

Aumentar la frecuencia de reloj supone disminuir el tiempo en que se ejecutan las instrucciones pero incrementa el consumo de energía.

1.2.9 RECURSOS ESPECIALES⁶

Cada fabricante oferta numerosas versiones de una arquitectura básica de microcontrolador. En algunas amplía las capacidades de las memorias, en otras incorpora nuevos recursos, en otras reduce las prestaciones al mínimo para aplicaciones muy simples, etc. La labor del diseñador es encontrar el modelo mínimo que satisfaga todos los requerimientos de su aplicación. De esta forma, minimizará el costo, el hardware y el software.

Principales recursos específicos incorporados en los microcontroladores:

- Temporizadores o "Timers".
- Perro guardián o "Watchdog".
- Protección ante fallo de alimentación o "Brownout".
- Estado de reposo o de bajo consumo.
- Conversor A/D.
- Conversor D/A.
- Comparador analógico.
- Modulador de ancho de pulso o PWM.
- Puertos de E/S digitales.
- Puertos de comunicación.

1.2.9.1 Temporizadores o "Timers"⁶

Se emplean para controlar periodos de tiempo (temporizadores) y para llevar la cuenta de acontecimientos que suceden en el exterior (contadores).

Para la medida de tiempos se carga un registro con el valor adecuado y a continuación dicho valor se va incrementando o decrementando al ritmo de los impulsos de reloj o algún múltiplo hasta que se desborde y llegue a 0, momento en el que se produce un aviso. Cuando se desean contar acontecimientos que se materializan por cambios de nivel o flancos en alguno de los pines del

⁶ www.monografias.com/trabajos12/microco/microco.shtml

microcontrolador, el mencionado registro se va incrementando o decrementando al ritmo de dichos impulsos.

1.2.9.2 Perro guardián o "Watchdog"⁶

El Perro guardián consiste en un temporizador que cuando se desborda y pasa por 0, provoca un reset automáticamente en el sistema. Se debe diseñar el programa de trabajo que controla la tarea de forma que refresque o inicialice al perro guardián antes de que provoque el reset. Si falla el programa o se bloquea, no se refrescará al perro guardián y al completar su temporización, "ladrará y ladrará" hasta provocar el reset.

Protección ante fallo de alimentación o "Brownout"⁶

Se trata de un circuito que resetea al microcontrolador cuando el voltaje de alimentación (V_{DD}) es inferior a un voltaje mínimo ("brownout"). Mientras el voltaje de alimentación sea inferior al de brownout el dispositivo se mantiene reseteado, comenzando a funcionar normalmente cuando sobrepasa dicho valor.

1.2.9.3 Estado de reposo o de bajo consumo⁶

Son abundantes las situaciones reales de trabajo en que el microcontrolador debe esperar, sin hacer nada, a que se produzca algún acontecimiento externo que le ponga de nuevo en funcionamiento.

Para ahorrar energía (factor clave en los aparatos portátiles) los microcontroladores disponen de una instrucción especial (SLEEP en los PIC), que les pasa al estado de reposo o de bajo consumo, en el cual los requerimientos de potencia son mínimos.

En dicho estado se detiene el reloj principal y se "congelan" sus circuitos asociados, quedando el microcontrolador sumido en un profundo "sueño". Al activarse una interrupción ocasionada por el acontecimiento esperado, el microcontrolador se despierta y reanuda su trabajo.

⁶ www.monografias.com/trabajos12/microco/microco.shtml

1.2.9.4 Conversor A/D (CAD)⁶

Los microcontroladores que incorporan un conversor A/D (Analógico/Digital) pueden procesar señales analógicas, tan abundantes en las aplicaciones. Suelen disponer de un multiplexor que permite aplicar a la entrada del CAD diversas señales analógicas desde los pines del circuito integrado.

1.2.9.5 Conversor D/A (CDA)⁶

Transforma los datos digitales obtenidos del procesamiento de un dispositivo en su correspondiente señal analógica que se envía al exterior por uno de los pines de la cápsula.

1.2.9.6 Comparador analógico⁶

Algunos modelos de microcontroladores disponen internamente de un amplificador operacional que actúa como comparador entre una señal fija de referencia y otra variable que se aplica por uno de los pines de la cápsula.

La salida del comparador proporciona un nivel lógico 1 o 0 según una señal sea mayor o menor que la otra. También hay modelos de microcontroladores con un módulo de tensión de referencia que proporciona diversas tensiones que se pueden aplicar en los comparadores.

1.2.9.7 Modulador de ancho de pulso o PWM⁶

Son circuitos que proporcionan en su salida pulsos de ancho variable, que se ofrecen al exterior a través de los pines del encapsulado.

Puertos de comunicación⁶

Con objeto de dotar al microcontrolador de la posibilidad de comunicarse con otros dispositivos externos, otros buses de microprocesadores, buses de sistemas, buses de redes y poder adaptarlos con otros elementos bajo otras

⁶ www.monografias.com/trabajos12/microco/microco.shtml

⁶ www.monografias.com/trabajos12/microco/microco.shtml

normas y protocolos, algunos modelos disponen de recursos que permiten esta tarea, entre los que destacan:

UART: adaptador de comunicación serie asincrónica.

USART: adaptador de comunicación serie sincrónica y asincrónica. Puerta paralela esclava para poder conectarse con los buses de otros microprocesadores.

USB (Universal Serial **Bus**): moderno bus serie para los PC.

Bus I₂C: interfaz serie de dos hilos desarrollado por Philips.

CAN (Controller Área Network): permite la adaptación con redes de conectedo multiplexado, desarrollado conjuntamente por Bosch e Intel para el cableado de dispositivos en automóviles.

1.3 MICROCONTROLADORES AVR

Los AVR son una familia de microcontroladores RISC de Atmel. La arquitectura de los AVR fue concebida por dos estudiantes en el Norwegian Institute of Technology, y posteriormente refinada y desarrollada en Atmel Norway, la empresa subsidiaria de Atmel, fundada por los dos arquitectos del chip.

El AVR es una CPU de arquitectura Harvard. Tiene 32 registros de 8 bits. Algunas instrucciones sólo operan en un subconjunto de estos registros. La concatenación de los 32 registros, los registros de entrada/salida y la memoria de datos conforman un espacio de direcciones unificado, al cual se accede a través de operaciones de carga/almacenamiento. A diferencia de los microcontroladores PIC, el stack se ubica en este espacio de memoria unificado, y no está limitado a un tamaño fijo.

El AVR fue diseñado desde un comienzo para la ejecución eficiente de código C compilado. Como este lenguaje utiliza profundamente punteros para el manejo de variables en memoria, los tres últimos pares de registros internos del procesador, son usados como punteros de 16 bits al espacio de memoria externa, bajo los nombres X, Y y Z. Esto es un compromiso que se hace en arquitecturas de ocho bits desde los tiempos de Intel 8008, ya que su tamaño

de palabra original de 8 bits (256 localidades accedidas) es pobre para direccionar. Por otro lado, hacer que todo el banco superior de 16 registros de 8 bits tenga un comportamiento alterno como un banco de 8 registros de 16 bits, complicaría mucho el diseño, violando la premisa original de su simplicidad. Además, algunas instrucciones tales como 'suma inmediata' faltan, ya que la instrucción 'resta inmediata' con el complemento dos puede ser usada como alternativa.

El set de instrucciones AVR está implementado físicamente y disponible en el mercado en diferentes dispositivos, que comparten el mismo núcleo AVR pero tienen distintos periféricos y cantidades de RAM y ROM: desde el microcontrolador de la familia Tiny AVR ATtiny11 con 1KB de memoria flash y sin RAM (sólo los 32 registros), y 8 pines, hasta el microcontrolador de la familia Mega AVR ATMEGA 2560 con 256KB de memoria flash, 8KB de memoria RAM, 4KB de memoria EEPROM, conversor análogo digital de 10 bits y 16 canales, temporizadores, comparador analógico, JTAG, etc. La compatibilidad entre los distintos modelos es preservada en un grado razonable.

Los microcontroladores AVR tienen una cañería ('pipeline' en inglés) con dos etapas (cargar y ejecutar), que les permite ejecutar la mayoría en un ciclo de reloj, lo que los hace relativamente rápidos entre los microcontroladores de 8-bit.

El set de instrucciones de los AVR es más regular que la de la mayoría de los microcontroladores de 8-bit (por ejemplo, los PIC). Sin embargo, no es completamente ortogonal:

- Los registros punteros X, Y y Z tienen capacidades de direccionamiento diferentes entre sí.
- Los registros 0 al 15 tienen diferentes capacidades de direccionamiento que los registros 16 al 31.
- Las registros de I/O 0 al 31 tienen distintas características que las posiciones 32 al 63.

- La instrucción CLR afecta los 'flag', mientras que la instrucción SER no lo hace, a pesar de que parecen ser instrucciones complementarias (dejar todos los bits en 1, y dejar todos los bits en 0 respectivamente).
- Los códigos de operación 0x95C8 y 0x9004 hacen exactamente lo mismo (LPM).

Como los PIC, tiene una comunidad de seguidores (ejemplificadas por el foro de internet AVRFreaks), principalmente debido a la existencia de herramientas de desarrollo gratuitas o de bajo costo. Estos microcontroladores están soportados por tarjetas de desarrollo de costo razonable, capaces de descargar el código al microcontrolador, y por una versión de las herramientas GNU. Esto último es posible por su uniformidad en el acceso al espacio de memoria, propiedad de la que carecen los procesadores de memoria segmentada o por bancos, como el PIC o el 8051 y sus derivados.

	<i>AT89S51</i>	<i>PIC16F877</i>	<i>ATMEGA16</i>
I/O	32	33	32
FLASH	4K	8K	16 K
RAM	128 BYTE	368 BYTE	1 KBYTE
EEPROM	ND	256 BYTE	512 BYTE
PWM	ND	2	4
ADC	ND	8 10BITS	8 10 BITS
TIMER 8BITS	ND	2 PRESCALER	2
PRESCALER			
TIMER 16BITS	2	1 PRESCALER	1 PRESCALER
IN CIRCUIT	SI	SI	SI
UART	1	1	1
SPI	ND	1	1
I2C	ND	1	1
CM	F/12	F/4	F
1K\$	1	4	

Tabla 1. 1 Característica de algunos microcontroladores utilizados en el medio.

¹⁰ GALARZA JUAN, Apuntes de microcontroladores AVR

1.4 SENSORES DE DISTANCIA POR ULTRASONIDO

1.4.1 FUNCIONAMIENTO BÁSICO

Los sensores de distancia por ultrasonido conocidos como ultrasonidos, son antes que nada sonido, exactamente igual que los que oímos normalmente, salvo que tienen una frecuencia mayor que la máxima audible por el oído humano. Ésta comienza desde unos 16 Hz y tiene un límite superior de

aproximadamente 20 KHz, mientras que nosotros vamos a utilizar sonido con una frecuencia de 40 KHz. A este tipo de sonidos es a lo que llamamos Ultrasonidos.

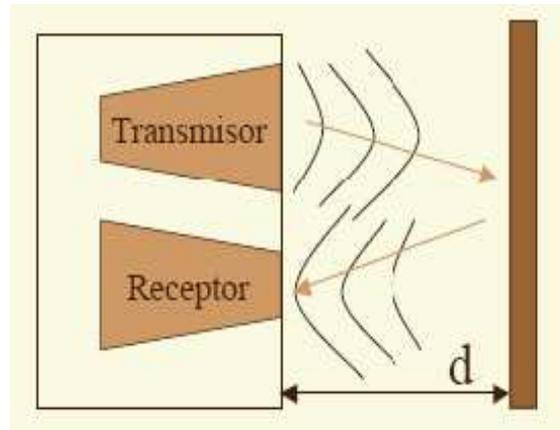


Figura 1. 14 Funcionamiento de los ultrasonidos

El funcionamiento básico de los ultrasonidos como medidores de distancia se muestra de una manera muy clara en el siguiente esquema, donde se tiene un transmisor que emite un pulso de ultrasonido que rebota sobre un determinado objeto y la reflexión de ese pulso es detectada por un receptor de ultrasonidos.

La mayoría de los sensores de ultrasonido de bajo costo se basan en la emisión de un pulso de ultrasonido cuyo lóbulo, o campo de acción, es de forma cónica. Midiendo el tiempo que transcurre entre la emisión del sonido y la percepción del eco se puede establecer la distancia a la que se encuentra el obstáculo que ha producido la reflexión de la onda sonora, mediante la fórmula:

$$d = \frac{1}{2} V \cdot t$$

Donde V es la velocidad del sonido en el aire y t es el tiempo transcurrido entre la emisión y recepción del pulso.

1.4.4 SRF08 SENSOR DISTANCIAS POR ULTRASONIDOS

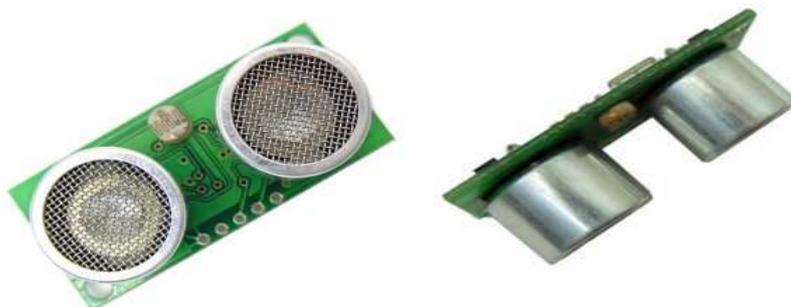


Figura 1. 15 Sensor SRF08

SRF08 es un medidor ultrasónico de distancias para robots que representa la última generación en sistemas de medidas de distancias por sonar, consiguiendo niveles de precisión y alcance únicos e impensables hasta ahora con esta tecnología. El sensor es capaz de detectar objetos a una distancia de 6 m con facilidad además de conectarse al microcontrolador mediante un bus I2C, por lo que se pueden conectar cuantos sensores sean necesarios en el mismo bus. Con una alimentación única de 5V, solo requiere 15 mA, para funcionar y 3mA mientras esta en reposo, lo que representa una gran ventaja para robots alimentados por pilas. El sensor SRF08 Incluye además un sensor de luz que permite conocer el nivel de luminosidad usando igualmente el bus I2C y sin necesidad de recursos adicionales.

1.4.4.2 Control del sensor de distancia por ultrasonido SRF08

La comunicación con el sensor ultrasónico SRF08 se realiza a través del bus I2C. Este está disponible en la mayoría de los controladores del mercado como BasicX-24, OOPic y Basic Stamp 2P, así como en una amplia gama de microcontroladores. Para el programador, el sensor SRF08 se comporta de la misma manera que las EEPROM de las series 24xx, con la excepción de que la dirección I2C es diferente. La dirección por defecto de fábrica del SRF08 es 0xE0. El usuario puede cambiar esta dirección con 16 direcciones diferentes: E0, E2, E4, E6, E8, EA, EC, EE, F0, F2, F4, F6, F8, FA, FC o FE, por lo que es posible utilizar hasta 16 sensores sobre un mismo bus I2C. Además de las direcciones anteriores, todos los sonares conectados al bus I2C responderán a

la dirección 0 -al ser la dirección de atención general. Esto significa que escribir un comando de medición de la distancia para la dirección 0 de I2C (0x00) iniciará las mediciones en todos los sensores al mismo tiempo. Esto debería ser útil en el modo ANN (Véase a continuación). Los resultados deben leerse de manera individual de cada uno de las direcciones reales de los sensores. Disponemos de ejemplos del uso de un módulo SRF08 con una amplia gama de controladores del mercado.

1.4.4.3 Conexiones

El pin señalado como "Do Not Connect" (No conectar) debería permanecer sin conexión. En realidad, se trata de la línea MCLR de la CPU y se utiliza solamente en la fábrica para programar el PIC16F872 después del montaje, dispone de una resistencia interna de tipo pull-up. Las líneas SCL y SDA deberían tener cada una de ellas una resistencia pull-up de +5v en el bus I2C. Sólo necesita un par de resistencias en todo el bus, no un par por cada módulo o circuito conectado al bus I2C. Normalmente se ubican en el bus maestro en vez de en los buses esclavos. El sensor SRF08 es siempre un bus esclavo - y nunca un bus maestro. Un valor apropiado sería el de 1,8 K en caso de que las necesitase. Algunos módulos como el OOPic ya disponen de resistencias pull-up por lo que no es necesario añadir ninguna más.

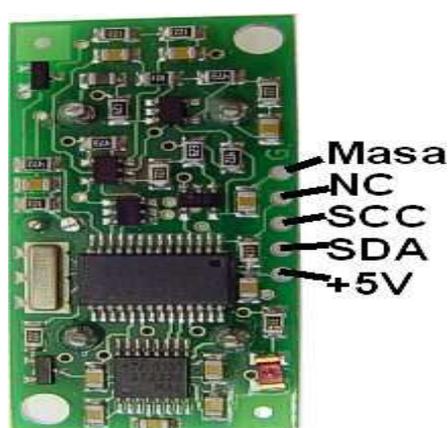


Figura 1. 16 Conexiones del SRF08.

1.4.4.4 Registros

El sensor SRF08 tiene un conjunto de 36 registros.

Ubicación	Lectura	Escritura
0	Revisión de Software	Registro de comando
1	Sensor de luz	Registro de ganancia máx. (por defecto 31)
2	Byte alto de 1º eco	Registro de alcance de distancia (por defecto 255)
3	Byte alto de 2º eco	No disponible
----	----	----
34	Byte alto de 17º eco	No disponible
35	Byte bajo de 17º eco	No disponible

Tabla 1. 2 Registros del SRF08

Solamente se puede escribir en las ubicaciones 0, 1 y 2. La ubicación 0 es el registro de comandos y se utiliza para iniciar la sesión de cálculo de la distancia. No puede leerse. La lectura de la ubicación da como resultado la revisión del software de SRF08. Por defecto, la medición dura 65mS, aunque puede cambiarse modificando el registro de alcance de la ubicación 2. Si lo hace, tendrá que cambiar la ganancia analógica en la ubicación 1. Consulte las secciones siguientes relacionadas con el cambio de medición y ganancia analógica.

La ubicación 1 es el sensor de luz en placa. Este dato se actualiza cada vez que se ejecuta un comando de medición de distancia y se puede leer cuando se leen los datos de la medición. Las dos ubicaciones siguientes, 2 y 3, son resultados sin signo de 16 bits de la última medición - el nivel lógico alto en primer lugar. El significado de este valor depende del comando utilizado, y puede estar expresado en pulgadas, o en centímetros, o bien el tiempo de vuelo del ping expresado en us. Un valor cero indica que no se ha detectado objeto alguno. Hay hasta 16 resultados adicionales que indican los ecos de objetos más lejanos.

1.4.4.5 Comandos

Existen tres comandos para iniciar una medición de distancia (desde 80 hasta 82), que devuelve el resultado en pulgadas, centímetros o microsegundos. Asimismo, también existe un modo ANN (Artificial Neural Network) que se describe a continuación y un grupo de comandos para modificar la dirección de I2C del srf08.

COMANDOS		ACCIÓN
Decimal	Hexadecimal	
80	0X50	Modo cálculo distancia - Resultado en pulgadas
81	0X51	Modo cálculo distancia - Resultado en centímetros
82	0X52	Modo cálculo distancia - Resultado en microsegundos
83	0X53	Modo ANN - Resultado en pulgadas
84	0X54	Modo ANN - Resultado en centímetros
85	0X55	Modo ANN - Resultado en micro-segundos
160	0XA0	1º en la secuencia para cambiar la dirección I2C
165	0XA5	3º en la secuencia para cambiar la dirección I2C
170	0XAA	2º en la secuencia para cambiar la dirección I2C

Tabla 1. 3 Comandos del SRF08

1.4.4.6 Modo de cálculo de distancia con el SRF08

Para iniciar la medición de la distancia, deberá escribir uno de los comandos anteriores en el registro de comando (registro 0) y esperar el tiempo necesario para la ejecución de la operación. A continuación, deberá leer el resultado en el formato que desee (pulgadas, centímetros, etc.). El búfer de eco se pone a cero al comienzo de cada medición. La primera medición del eco se coloca en las ubicaciones 2 y 3, la segunda en 4 y 5, etc. Si una ubicación (niveles altos o bajos de bytes) es 0, entonces no se encontrará ningún otro valor en el resto de los registros. El tiempo recomendado y establecido por defecto para realizar la operación es de 65mS, sin embargo es posible acortar este periodo escribiendo

en el registro de alcance antes de lanzar el comando de medición. Los datos del sensor de luz de la ubicación 1 se actualizarán también después del comando de medición.

1.4.4.7 Modo ANN

El modo ANN (Artificial Neural Network) ha sido diseñado para proporcionar datos múltiples de un modo en el que es más fácil entrar en una red neural, o al menos eso es lo que se pretende - aunque aún no se ha hecho. El modo ANN ofrece un búfer de 32 bytes (ubicaciones de 4 a 35 inclusive) en el que cada byte representa el tiempo máximo de vuelo 65536uS dividido por 32 tramos de 2048uS cada uno - equivalente a aproximadamente 352mm de alcance. Si se recibe un eco en uno de los espacios de tiempo de bytes, a continuación se fijará en un valor diferente a cero, para que no sea cero. Por lo tanto si se recibe un eco desde los primeros 352mm, la ubicación 4 será diferente a cero. Si se detecta un objeto a 3 metros de distancia, la ubicación 12 será diferentes de cero ($3000/352 = 8$) ($8+4=12$). Organizar los datos de esta manera sería mejor para una red neural que para otros formatos. La entrada a su red debería ser 0 si el byte es cero y 1 si es diferente de cero. En el futuro, se pretende organizar un mapa SOFM (Self Organizing Feature Map) para la red neural, aunque se espera que sea aplicable para cualquier tipo de red.

Ubicación 4	Ubicación 5	Ubicación 6	Ubicación 7	Ubicación 8
0-352mm	353-705mm	706-1057mm	1058-1410mm	En adelante

Tabla 1. 4 Modo Ann

1.4.4.8 Cómo comprobar que una medición ha finalizado

No es necesario utilizar un temporizador en su propio controlador para saber que la medición ha terminado. Puede aprovechar la ventaja que le ofrece el hecho de que el sensor SRF08 no responde a ninguna otra actividad I2C mientras está realizando la medición. Por lo tanto, si intenta leer el valor en el

sensor SRF08 (utilizamos el número de revisión de software en la ubicación 0) por lo que recibirá 255 (0xFF) durante la medición. Esto se debe a que la línea de datos I2C (SDA) se eleva si nada lo está controlando. Tan pronto como finaliza la medición el sensor SRF08 responderá de nuevo al bus I2C, por lo que deberá esperar a que desaparezca el valor 255 (0xFF) en el registro. A continuación, podrá leer los datos del sensor. El controlador puede aprovechar esta ventaja para realizar otras tareas mientras el SRF08 está realizando la medición.

1.4.4.9 Cómo cambiar el rango de alcance

El alcance máximo del sensor SRF08 está controlado por el temporizador interno. Por defecto, este es 65mS o el equivalente a 11 metros de alcance. Esto supera los 6 metros de los que el SRF08 es realmente capaz de ofrecer. Es posible reducir el tiempo que espera el sensor SRF08 a escuchar un eco, y por lo tanto el alcance, modificando el registro range en la ubicación 2. El alcance puede regularse en pasos de aproximadamente 43mm (0,043 metros o 1,68 pulgadas) hasta llegar a los 11 metros. El alcance es ((Range Register x 43mm) + 43mm) por lo que fijar este registro (Range Register) en el valor 0 (0x00) ofrece un alcance máximo de 43mm. Fijar el registro Range Register en el valor 1 (0x01) ofrece un alcance máximo de 86mm. En un ejemplo más útil, el valor 24 (0x18) ofrece un alcance de 1 metro mientras que el valor 140 (0x8C) da 6 metros. El valor 255 (0xFF) ofrece los 11 metros originales (255 x 43 + 43 es 11008mm).

Existen dos razones por las que es positivo reducir el tiempo de medición.

1. Para obtener la información sobre el alcance en menos tiempo.
2. Para poder realizar mediciones con el sensor SRF08 a una tasa más rápida.

Si lo único que desee es recibir en menos tiempo, la información sobre el alcance y pretende realizar las mediciones a una tasa de 65ms o más lento, todo funcionará de manera correcta. Sin embargo, si desea lanzar el sensor

SRF08 a una tasa ligeramente más alta de 65mS, deberá reducir la ganancia - consulte la siguiente sección.

El alcance está fijado en el valor máximo cada vez que se pone en marcha el sensor SRF08. Si necesita un alcance diferente, cámbielo al principio como parte del código de iniciación del sistema.

1.4.4.10 Ganancia analógica

En el registro de la ganancia analógica, se configura la ganancia máxima de las etapas analógicas. Para configurar la ganancia máxima del srf08, simplemente deberá escribir uno de estos valores en el registro de ganancia de la ubicación 1. Durante la medición, la ganancia analógica empieza con su valor mínimo de 94. Este valor se incrementa en intervalos de aproximadamente 70uS hasta llegar al valor de ganancia máxima, configurada en el registro 1. La ganancia máxima posible se alcanza después de aproximadamente 390mm de alcance.

La finalidad de poder limitar la ganancia máxima es permitirle iniciar mediciones a una frecuencia mayor de 65mS. Dado que la medición puede ser muy corta, es posible iniciar una nueva medición tan pronto como se hayan leído los datos de la medición previa. Un riesgo potencial de esto es que la segunda medición podría captar un retorno de un eco distante del "ping" anterior, dando un resultado falso referente a un objeto cercano cuando en realidad no hay ninguno. Para reducir esta posibilidad, la ganancia máxima puede reducirse para limitar la sensibilidad de los módulos al eco distante más débil, mientras que al mismo tiempo sigue siendo capaz de detectar la proximidad de objetos.

La configuración de la ganancia máxima se almacena sólo en la memoria RAM del CPU y se inicia con el encendido del equipo, por lo que si sólo desea realizar las mediciones cada 65mS, o más, puede ignorar los registros Range y Gain.

Nota - Es efectivo sólo en Modo de cálculo de distancia, en el Modo ANN, la ganancia se controla automáticamente.

Registro de ganancia		Ganancia analógica máxima
Decimal	Hexadecimal	
0	0X00	Fija la ganancia analógica máxima en 94
1	0X01	Fija la ganancia analógica máxima en 97
2	0X02	Fija la ganancia analógica máxima en 100
3	0X03	Fija la ganancia analógica máxima en 103
4	0X04	Fija la ganancia analógica máxima en 107
5	0X05	Fija la ganancia analógica máxima en 110
6	0X06	Fija la ganancia analógica máxima en 114
7	0X07	Fija la ganancia analógica máxima en 118
8	0X08	Fija la ganancia analógica máxima en 123
9	0X09	Fija la ganancia analógica máxima en 128
10	0X10	Fija la ganancia analógica máxima en 133
11	0X11	Fija la ganancia analógica máxima en 139
12	0X12	Fija la ganancia analógica máxima en 145
13	0X13	Fija la ganancia analógica máxima en 152
14	0X14	Fija la ganancia analógica máxima en 159
15	0X15	Fija la ganancia analógica máxima en 168
16	0X16	Fija la ganancia analógica máxima en 177
17	0X17	Fija la ganancia analógica máxima en 187
18	0X18	Fija la ganancia analógica máxima en 199
19	0X19	Fija la ganancia analógica máxima en 212
20	0X20	Fija la ganancia analógica máxima en 227
21	0X21	Fija la ganancia analógica máxima en 245
22	0X22	Fija la ganancia analógica máxima en 265
23	0X23	Fija la ganancia analógica máxima en 288
24	0X24	Fija la ganancia analógica máxima en 317
25	0X25	Fija la ganancia analógica máxima en 352
26	0X26	Fija la ganancia analógica máxima en 395
27	0X27	Fija la ganancia analógica máxima en 450
28	0X28	Fija la ganancia analógica máxima en 524
29	0X29	Fija la ganancia analógica máxima en 626

30	0X30	Fija la ganancia analógica máxima en 777
31	0X31	Fija la ganancia analógica máxima en 1025

Tabla 1. 5 Ganancia analógica

Tenga en cuenta que la relación entre el registro de ganancia y la ganancia real no es una relación lineal. No existe una fórmula mágica que diga "si utiliza este valor de ganancia, el alcance será exactamente este". Depende del tamaño, forma, y material del objeto y de los elementos restantes de la habitación. Lo recomendable es experimentar con diferentes valores hasta obtener los resultados deseados. Si obtiene lecturas falsas, puede que sean los ecos de los "pings" anteriores, vuelva a lanzar el sensor SRF08 cada 65mS o más (menos tasa).

Si tiene alguna duda acerca de los registros Range y Gain, recuerde que en el sensor SRF08 se fijan los valores por defecto automáticamente cuando se inicia el sistema. Es más, puede olvidarse de esta configuración y utilizar los valores por defecto y el sensor funcionará correctamente, detectando objetos a 6 metros cada 65mS o menos.

1.4.4.11 Medidas

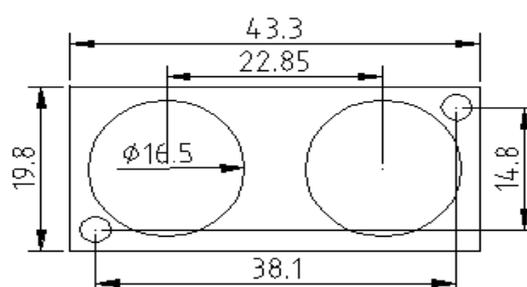


Figura 1. 17 Medidas del SRF08

1.4.4.12 Sensor de luz

El medidor ultrasónico SRF08 dispone de un sensor fotoeléctrico en la propia placa. Este medidor realiza una lectura de la intensidad de la luz cada vez que se calcula la distancia en los modos Ranging o ANN (La conversión

analógica/digital se realiza realmente justo antes de que se lance el "ping" mientras el generador de 10v +/- se encuentra en fase de estabilización). EL valor de la lectura va aumentando a medida que aumenta la intensidad de la luz, por lo que valor máximo lo obtendrá con una luz brillante y el valor mínimo en total oscuridad. La lectura debería acercarse a 2-3 en total oscuridad y aproximadamente a 248 (0xF8) en luz diurna. La intensidad de la luz puede leerse en el registro del sensor de luz en la ubicación 1 al mismo tiempo que puede leer los datos del alcance.

1.4.4.13 Led

EL indicador Led rojo se utiliza para indicar el código de la dirección I2C del sensor en el encendido (ver abajo). Así mismo, también emite un breve destello durante el "ping" en el cálculo de la distancia.

1.4.4.14 Cambio de la dirección del bus I2C del SRF08

Para modificar la dirección I2C del sensor SRF08 sólo podrá tener un sensor conectado al bus. Escriba los 3 comandos de secuencias en el orden correcto seguidos de la dirección. Ejemplo; para cambiar la dirección de un sensor que tiene actualmente la dirección 0xE0 (la dirección de fábrica por defecto) a la dirección 0xF2, escriba lo siguiente en la dirección 0xE0; (0xA0, 0xAA, 0xA5, 0xF2). Se deberían enviar estos comandos con el orden secuencial correcto para modificar la dirección I2C. Además, no es posible emitir cualquier otro comando en medio de la secuencia. La secuencia debe enviarse al registro de comandos de la ubicación 0, lo que implica que se escribirán 4 transacciones independientes en el bus I2C. Una vez realizado todo esto, deberá etiquetar el sensor con su dirección. No obstante, si olvida hacerlo, cuando lo encienda, no se enviará ningún comando.

El sensor SRF08 indicará su dirección mediante el LED. Un destello largo seguido de un número de destellos cortos indicará la dirección. Los destellos terminarán inmediatamente después de enviar un comando al sensor SRF08.

Dirección			
Decimal	Hexadecimal	Destello Largo	Destellos cortos
224	E0	1	0
226	E2	1	1
228	E4	1	2
230	E6	1	3
232	E8	1	4
234	EA	1	5
236	EC	1	6
238	EE	1	7
240	F0	1	8
242	F2	1	9
244	F4	1	10
246	F6	1	11
248	F8	1	12
250	FA	1	13
252	FC	1	14
254	FE	1	15

Tabla 1. 6 Dirección del bus I2C del SRF08

Asegúrese de no configurar más de un sensor con la misma dirección, ya que se produciría una colisión en el bus, con resultados totalmente imprevisibles.

1.4.4.15 Consumo de corriente

El consumo medio de corriente se calcula que es aproximadamente 12mA durante el cálculo de la distancia, y 3mA en modo de espera. El módulo entrará automáticamente en modo de espera después de terminar la medición, mientras espera al siguiente comando del bus I2C. El perfil real de consumo de corriente del SRF08 es el siguiente:

Tipo de operación realizada	Corriente	Duración
Comando de medición de la distancia recibido –Encendido	275mA	3uS
Estabilización del generador de +/- 10v	25mA	600uS
8 ciclos de "ping" 40kHz	40mA	200uS
Medición	1mA	65mS máx.
Modo de espera (Stand-by)	3mA	Indefinido

Tabla 1. 7 Consumo de corriente

Los valores de la tabla anterior se ofrecen sólo a modo orientativos, no se han comprobado en unidades de producción.

1.4.4.16 Cambio del ángulo de detección

El ángulo de detección no se puede cambiar. Esta es una pregunta que se hace muy frecuentemente y cuya respuesta es que no se puede alterar. El foco de trabajo del SRF08 es un cono cuyo ancho depende del propio traductor y esta es fija. La forma del área de trabajo del traductor ultrasónico empleado en el SRF08 es la de la siguiente figura, tomada de la hoja de características del fabricante.

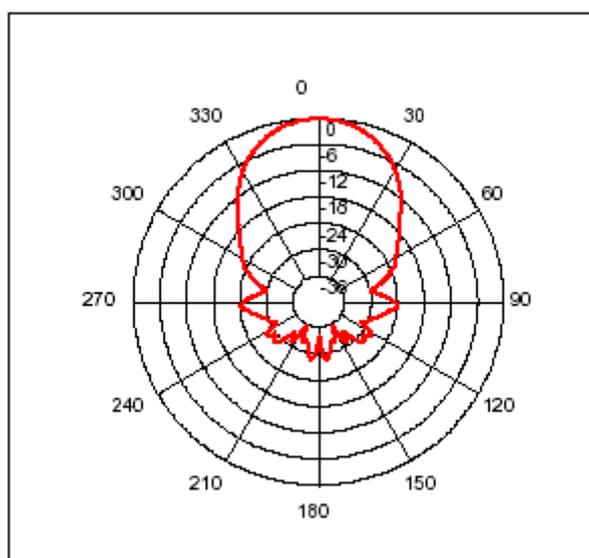


Figura 1. 18 Ángulo de detección.

1.5. LCD

1.5.1 DEFINICIÓN

LCD (Liquid Cristal Display) son las siglas en inglés de Pantalla de Cristal Líquido, dispositivo inventado por Jack Janning, quien fue empleado de NCR.

Se trata de un sistema eléctrico de presentación de datos formado por 2 capas conductoras transparentes y en medio un material especial cristalino (cristal líquido) que tienen la capacidad de orientar la luz a su paso.

Cuando la corriente circula entre los electrodos transparentes con la forma a representar (por ejemplo, un segmento de un número) el material cristalino se reorienta alterando su transparencia.

El material base de un LCD lo constituye el cristal líquido, el cual exhibe un comportamiento similar al de los líquidos y unas propiedades físicas anisótropas similares a las de los sólidos cristalinos. Las moléculas de cristal líquido poseen una forma alargada y son más o menos paralelas entre sí en la fase cristalina.

Según la disposición molecular y su ordenamiento, se clasifican en tres tipos: nemáticos, esméticos y colestéricos. La mayoría de cristales responden con facilidad a los campos eléctricos, exhibiendo distintas propiedades ópticas en presencia o ausencia del campo. El tipo más común de visualizador LCD es, con mucho, el denominado nemático de torsión, término que indica que sus moléculas en su estado desactivado presentan una disposición en espiral. La polarización o no de la luz que circula por el interior de la estructura, mediante la aplicación o no de un campo eléctrico exterior, permite la activación de una serie de segmentos transparentes, los cuales rodean al cristal líquido. Según sus características ópticas, pueden también clasificarse como: reflectivos, transmisivos y transreflectivos.

Las pantallas LCD se encuentran en multitud de dispositivos industriales y de consumo: máquinas expendedoras, electrodomésticos, equipos de

telecomunicaciones, computadoras, etc. Todos estos dispositivos utilizan pantallas fabricadas por terceros de una manera más o menos estandarizada.

Cada LCD se compone de una pequeña placa integrada que consta de:

- La propia pantalla LCD.
- Un microchip controlador (Circuito integrado).
- Una pequeña memoria que contiene una tabla de caracteres.
- Un interfaz de contactos eléctricos, para conexión externa.
- Opcionalmente, una luz trasera para iluminar la pantalla.

El controlador simplifica el uso del LCD proporcionando una serie de funciones básicas que se invocan mediante el interfaz eléctrico, destacando:

- La escritura de caracteres en la pantalla.
- El posicionado de un cursor parpadeante, si se desea.
- El desplazamiento horizontal de los caracteres de la pantalla (scrolling).
- Etc.

La memoria implementa un mapa de bits para cada carácter de un juego de caracteres, es decir, cada octeto de esta memoria describe los puntitos o píxeles que deben iluminarse para representar un carácter en la pantalla. Generalmente, se pueden definir caracteres a medida modificando el contenido de esta memoria. Así, es posible mostrar símbolos que no están originalmente contemplados en el juego de caracteres.

El interfaz de contactos eléctricos suele ser de tipo paralelo, donde varias señales eléctricas simultáneas indican la función que debe ejecutar el controlador junto con sus parámetros. Por tanto, se requiere cierta sincronización entre estas señales eléctricas.

La luz trasera facilita la lectura de la pantalla LCD en cualquier condición de iluminación ambiental.

Existen dos tipos de pantallas LCD en el mercado: pantallas de texto y pantallas gráficas.

1.5.2 FUNCIONAMIENTO

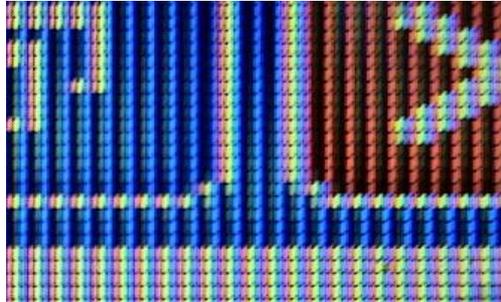


Figura 1. 19 Imagen donde se pueden observar los píxeles1

El funcionamiento de estas pantallas se fundamenta en sustancias que comparten las propiedades de sólidos y líquidos a la vez. Cuando un rayo de luz atraviesa una partícula de estas sustancias tiene necesariamente que seguir el espacio vacío que hay entre sus moléculas como lo haría atravesar un cristal sólido pero a cada una de estas partículas se le puede aplicar una corriente eléctrica que cambie su polarización dejando pasar a la luz o no.

Una pantalla LCD está formada por 2 filtros polarizados colocados perpendicularmente de manera que al aplicar una corriente eléctrica al segundo de ellos dejaremos pasar o no la luz que ha atravesado el primero de ellos. Para conseguir el color es necesario aplicar tres filtros más para cada uno de los colores básicos rojo, verde y azul y para la reproducción de varias tonalidades de color se deben aplicar diferentes niveles de brillo intermedios entre luz y no luz lo, cual consigue con variaciones en el voltaje que se aplicaba los filtros.

1.5.3 LCD DE TEXTO

Solamente permiten visualizar mensajes cortos de texto. Existen algunos modelos estandarizados en la industria, en función de su tamaño medido en número de líneas y columnas de texto. Existen modelos de una, dos y cuatro

filas únicamente. El número de columnas típico es de ocho, dieciséis, veinte y cuarenta caracteres.



Figura 1. 20 Display lcd de texto de 16x2 con controlador hd44780.

El controlador Hitachi HD44780 se ha convertido en un estándar de industria cuyas especificaciones funcionales son imitadas por la mayoría de los fabricantes.

Este controlador cuenta con los siguientes interfaces eléctricos:

- D0-D7: ocho señales eléctricas que componen un bus de datos.
- R/W: una señal que indica si se desea leer o escribir en la pantalla (generalmente solamente se escribe).
- RS: una señal que indica si los datos presentes en D0-D7 corresponden bien a una instrucción, bien a sus parámetros.
- E: una señal para activar o desactivar la pantalla.
- V0: señal eléctrica para determinar el contraste de la pantalla. Generalmente en el rango de cero a cinco voltios. Cuando el voltaje es de cero voltios se obtienen los puntos más oscuros.
- Vss y Vdd: señales de alimentación. Generalmente a cinco voltios.

Estas señales son fácilmente controladas desde un ordenador a través de un interfaz paralelo, típicamente a través del interfaz IEEE 1284, también conocido como "Centronics". El mismo que se utiliza para conectar impresoras

1.5.4 LCD DE GRÁFICOS

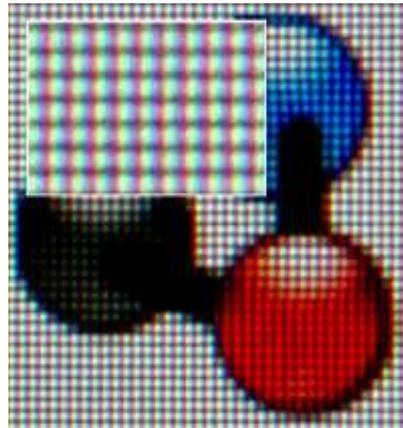


Figura 1. 21 Detalle de una pantalla LCD en color.

Las pantallas LCD gráficas permiten encender y apagar individualmente píxeles de la pantalla. De esta manera es posible mostrar gráficos en blanco y negro, no solamente texto. Los controladores más populares son el Hitachi HD61202 y el Samsung KS0108. Los tamaños también están estandarizados y se miden en filas y columnas de píxeles. Algunos tamaños típicos son 128x64 y 96x60.

Naturalmente algunos controladores también permiten la escritura de texto de manera sencilla.

1.6 SECURE DIGITAL

Secure Digital (SD) es un formato de tarjeta de memoria flash. Se utiliza en dispositivos portátiles tales como cámaras fotográficas digitales, PDAs, teléfonos móviles e incluso videoconsolas (tanto de sobremesa como la Wii como de mano como la GP2X), entre otros.



Figura 1. 22 Memoria SD.

Estas tarjetas tienen unas dimensiones de 32 mm x 24 mm x 2'1 mm. Existen dos tipos: unos que funcionan a velocidades normales, y otros de alta velocidad que tienen tasas de transferencia de datos más altas. Algunas cámaras fotográficas digitales requieren tarjetas de alta velocidad para poder grabar vídeo con fluidez o para capturar múltiples fotografías en una sucesión rápida.

Los dispositivos con ranuras SD pueden utilizar tarjetas MMC, que son más finas, pero las tarjetas SD no caben en las ranuras MMC. Asimismo, se pueden utilizar directamente en las ranuras de CompactFlash o de PC Card con un adaptador.

Sus variantes MiniSD y MicroSD se pueden utilizar, también directamente, en ranuras SD mediante un adaptador. Hay algunas tarjetas SD que tienen un conector USB integrado con un doble propósito, y hay lectores que permiten que las tarjetas SD sean accesibles por medio de muchos puertos de conectividad como USB, FireWire y el puerto paralelo común. Las tarjetas SD también son accesibles mediante una disquetera usando un adaptador FlashPath.

1.6.1 MiniSD



Figura 1. 23Memoria MiniSD y adaptador.

MiniSD es un formato de tarjeta de memoria flash. Presentada por primera vez por SanDisk en CeBIT 2003, el miniSD se unió a la Memory Stick Duo y XD-Picture Card en cuanto a dispositivos pequeños.

La tarjeta miniSD fue adoptada en 2003 por la Asociación de Tarjetas SD (SD Card Association) como una extensión de tamaño ultra-pequeño para el estándar de tarjeta SD.

Dado que las nuevas tarjetas se diseñaron especialmente para ser usadas en teléfonos móviles, están envueltas por un adaptador miniSD que permite la compatibilidad con todos los dispositivos equipados con una ranura de tarjeta SD.

1.6.2 MicroSD

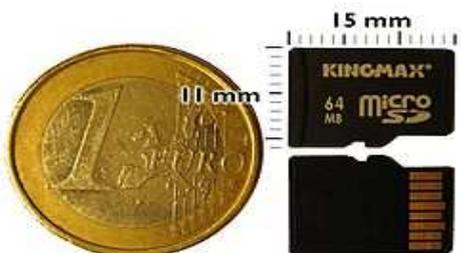


Figura 1. 24 Tarjeta MicroSD (64MB) al lado de una moneda 1€

Las tarjetas MicroSD o Transflash corresponden a un formato de tarjeta de memoria flash más pequeña que la MiniSD, desarrollada por SanDisk; adoptada por la Asociación de Tarjetas SD (SD Card Association) bajo el nombre de «MicroSD» en julio de 2005. Mide tan solo 15 × 11 × 0,7 milímetros, lo cual le da un área de 165 mm². Esto es tres y media veces más pequeña que la miniSD, que era hasta ahora el formato más pequeño de tarjetas SD, y es alrededor de un décimo del volumen de una SD card. Sus tasas de Transferencia no son muy altas, sin embargo, empresas como SanDisk han trabajado en ello, llegando a versiones que soportan velocidades de lectura de hasta 10 Mbit/s.

Debido a que su costo duplica al de una Secure Digital, su uso se ciñe a aplicaciones donde el tamaño es crítico, como los teléfonos móviles, sistemas GPS o tarjetas Flash para consolas de mano (como la GameBoy Advance o la Nintendo DS)

1.6.3 MULTI MEDIA CARD



Figura 1. 25 Tarjeta MMC

MultiMediaCard, o MMC, es un estándar de tarjeta de memoria flash. Prácticamente igual a la SD, carece de la pestaña de seguridad que evita sobrescribir la información grabada en ella.

Su forma está inspirada en el aspecto de los antiguos disquetes de 3,5 pulgadas. Actualmente ofrece una capacidad máxima de 4 GB.

Presentada en 1997 por Siemens AG y SanDisk, se basa en la memoria flash de Toshiba base NAND, y por ello es más pequeña que sistemas anteriores basados en memorias flash de Intel base NOR, tal como la CompactFlash. MMC tiene el tamaño de un sello de correos: 24 mm x 32 mm x 1,4 mm.

Originalmente usaba un interfaz serie de 1-bit, pero versiones recientes de la especificación permite transferencias de 4 o a veces incluso 8 bits de una vez. Han sido más o menos suplantadas por las Secure Digital (SD), pero siguen teniendo un uso importante porque las MMCs pueden usarse en la mayoría de aparatos que soportan tarjetas SD (son prácticamente iguales), pudiendo retirarse fácilmente para leerse en un PC.

Las MMCs están actualmente disponibles en tamaños de hasta 4GB con modelos de 8GB anunciados, aún no disponibles. Se usan en casi cualquier contexto donde se usen tarjetas de memoria, como teléfonos móviles, reproductores de audio digital, cámaras digitales y PDAs.

Desde la introducción de la tarjeta Secure Digital y la ranura SDIO (Secure Digital Input/Output), pocas compañías fabrican ranuras MMC en sus dispositivos, pero las MMCs, ligeramente más delgadas y de pines compatibles, pueden usarse en casi cualquier dispositivo que soporte tarjetas SD si lo hace su software/firmware.

1.6.1.1 Especificaciones

	Tarjeta miniSD	Tarjeta SD	Tarjeta MicroSD
Ancho	20mm	24mm	11mm
Largo	21,5mm	32mm	15mm
Grosor	1,4mm	2,1mm	1mm
Volumen de la tarjeta	589 mm ³	1,596 mm ³	165mm ³
Peso	1g aprox.	2g aprox.	< 1g aprox.
Voltaje de funcionamiento	2,7 - 3,6V	2,7 - 3,6V	2,7 - 3,6V
Interruptor de protección contra escritura	No	Sí	No
Protectores de terminal	No	Sí	No
Número de pines	11 pines	9 pines	8 pines

Tabla 1. 8 Descripción Memorias

CAPÍTULO II

CONSTRUCCIÓN DEL HARDWARE Y SOFTWARE

2.1 INTRODUCCIÓN

En el capítulo anterior se dio una descripción de los diferentes elementos que se utilizó en el proyecto, dando conceptos generales de los elementos principales, en este capítulo se detalla la construcción del hardware del proyecto siendo el objetivo principal que detecte los obstáculos frontales y huecos, cada sistema sensor debe de comunicarse tanto con el sistema de actuadores como también con el sistema de control, y hacer que todos los sistemas trabajen en conjunto.

2.2 ARQUITECTURA DEL BASTÓN

Conociendo de la necesidad las personas no videntes, realizamos una breve investigación de los sistemas de ayuda existentes en el mercado, con lo cual se decidió la construcción de un bastón electrónico por medio de sensores de ultrasonido por las características propias de este elemento.

2.2.1 FILOSOFÍA DE CONTROL DEL BASTÓN ELECTRÓNICO

La idea principal del proyecto de titulación es construir un bastón que detecte un objeto u obstáculo frontalmente y un agujero en el suelo, al detectar el obstáculo frontalmente enviara un dato al microcontrolador el cual procesará y se comunicará con el módulo de voz reproduciendo un mensaje de voz que dirá a qué distancia se encuentra el objeto, en caso de ser un agujero reproduce un mensaje de alerta.

Su estructura está formada de madera, ya que como elemento principal para el montaje mecánico y soporte de los circuitos impresos se utilizará un soporte en forma de bastón.

La tarjeta principal del bastón es la encargada de inicializar los sensores de ultrasonido, enviar una señal la cual al detectar un objeto refleja dicha señal en el receptor de ultrasonido, enviando un dato al microcontrolador ATMEGA 164 el cuál por medio de software se conecta con el módulo de voz, indicando que mensaje se tiene que reproducir. Se utilizó el microprocesador de la familia ATMEL, debido a que este elemento soporta todos los recursos necesarios para el desarrollo ó implementación de la misma, con una bondad que se debe recalcar, en estos chips se puede implementar software con bastante facilidad, teniendo en cuenta que no se necesita de muchos elementos conectados a este para que pueda funcionar óptimamente y son de fácil adquisición.

El microcontrolador ATMEGA 164, ubicado en la tarjeta principal, funciona como el “Cerebro” del bastón electrónico, pues controla todo el flujo de datos que circula a través de los circuitos de control, en el se han conectado todos los sensores, en función de este microcontrolador, el bastón detectará que obstáculo se encuentra frontal o perpendicular, advirtiéndolo a la persona por medio de mensajes de voz que existe un obstáculo en esa dirección.

De manera global, el bastón está formado por un Cerebro (Sistema de Control), Sistema Sensorial (Sensores de ultrasonido), y los Actuadores (Tarjeta de módulo de voz) como podemos ver en la Figura 2.1

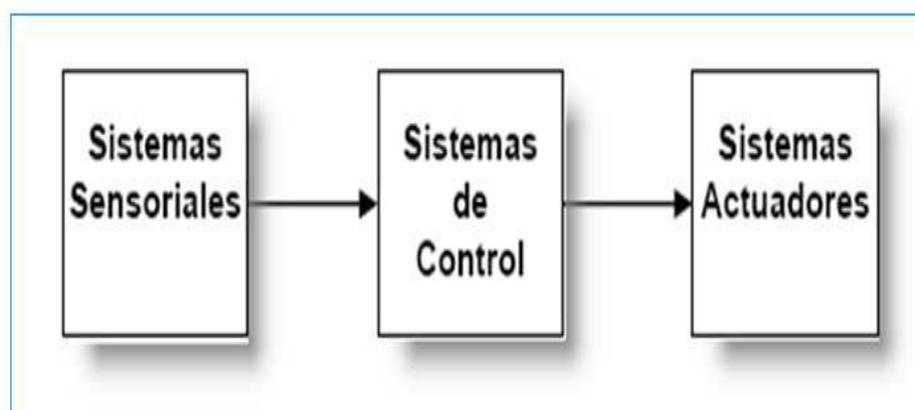


Figura 2. 1 Diagrama de bloques global, “Sistema a lazo abierto”

2.2 DIAGRAMA DE BLOQUES DEL BASTÓN ELECTRÓNICO

Se explicarán de manera muy general todos los sistemas de los cuales está compuesto el Bastón electrónico, el mismo que se encuentra compuesto de tres partes fundamentales: su Sistema de Control (Tarjeta Principal), su Sistema Actuador (Módulo de Voz) y su sistema de detección de obstáculos (SRF08) (Figura 2.2).



Figura 2. 2 Diagrama de bloques del bastón electrónico.

2.3 NOMENCLATURA:



SISTEMA DE DETECCIÓN DE OBSTÁCULOS (SFR08)



SISTEMA ACTUADOR (MÓDULO DE VOZ)



SISTEMA DE CONTROL (TARJETA PRINCIPAL)

2.4 SISTEMA DE DETECCIÓN DE OBSTÁCULOS

2.4.1 INTRODUCCIÓN

Uno de los sistemas que le ayudará en forma muy considerable al Bastón Electrónico, es el sistema de detección de obstáculos. Este sistema le brinda al

usuario del Bastón Electrónico la posibilidad de desplazarse con mayor facilidad.

Hemos visto conveniente utilizar los sensores SRF08 de ultrasonido que representa la última generación en sistemas de medidas de distancias por sonar, consiguiendo niveles de precisión y alcance únicos e impensables hasta ahora con esta tecnología. El sensor es capaz de detectar objetos a una distancia de 6 m, con la facilidad además de conectarse al microcontrolador mediante un bus I2C, por lo que se pueden conectar cuantos sensores sean necesarios en el mismo bus. Con una alimentación única de 5V, solo requiere 15 mA, para funcionar y 3mA mientras esta en reposo, lo que representa una gran ventaja para robots alimentados por pilas. El sensor SRF08 incluye además un sensor de luz que permite conocer el nivel de luminosidad usando igualmente el bus I2C y sin necesidad de recursos adicionales.

2.4.2 ANÁLISIS DEL CIRCUITO

Dispone de un microcontrolador AVR ATMEGA 164 el cual está programado para hacer funcionar los tres sensores SRF08 de ultrasonido, S1 y S2 los cuales se encuentran ubicados en la parte frontal del bastón con ángulo de 90° , S3 se encuentra en la parte frontal inferior del bastón con un ángulo de -90° , como se muestra en la figura 2.3.

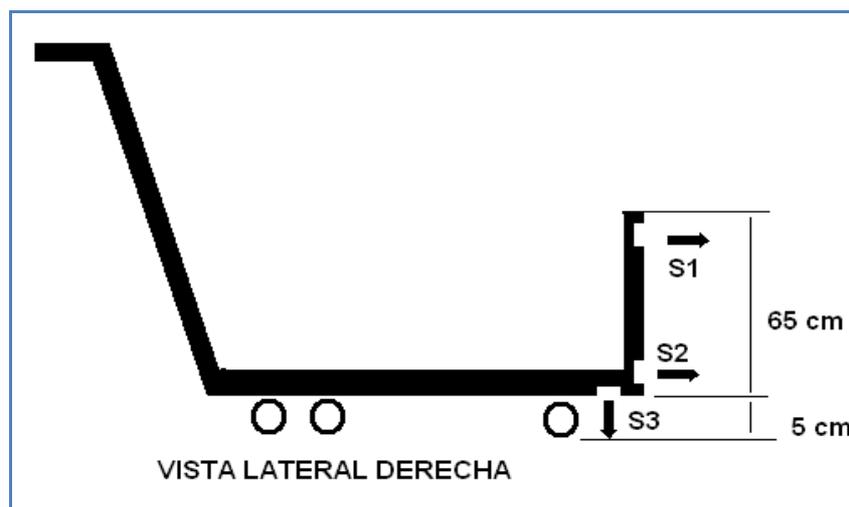


Figura 2. 3 Análisis del circuito.

2.5 SISTEMA DE CONTROL “TARJETA PRINCIPAL”

Este es el sistema más importante del bastón electrónico, pues al no disponer de un administrador o controlador de acciones (Tarjeta principal), existiría un gran conflicto entre cada uno de los circuitos y los resultados no serían nada agradables. El software del “Tarjeta principal” como ya se mencionó está implementado en un microcontrolador ATMEGA 164.

- **Sistema concurrente:** Aunque el proceso de ejecución y control de acciones, como también del monitoreo de cada uno de los sensores es serial (Figura 2.4); es decir ejecuta una sola acción a la vez (en tiempo del orden de microsegundos), para la percepción humana resulta que el control y monitoreo se da simultáneamente. Además ciertas situaciones son manejadas por interrupción de hardware y software, es decir cualquier evento que genere una interrupción, es atendido automáticamente.

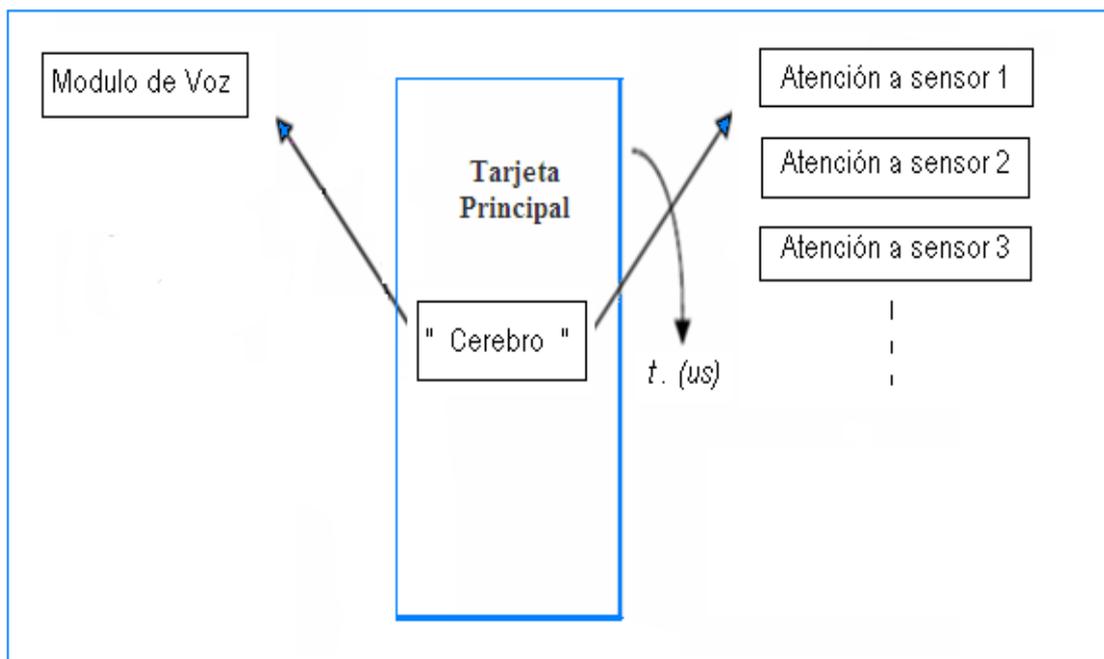


Figura 2. 4 Diagrama del control y monitoreo del bastón serialmente.

2.6 FUNDAMENTO DEL SISTEMA ACTUADOR

2.6.1 SISTEMA DE REPRODUCCIÓN DE MENSAJES DE VOZ

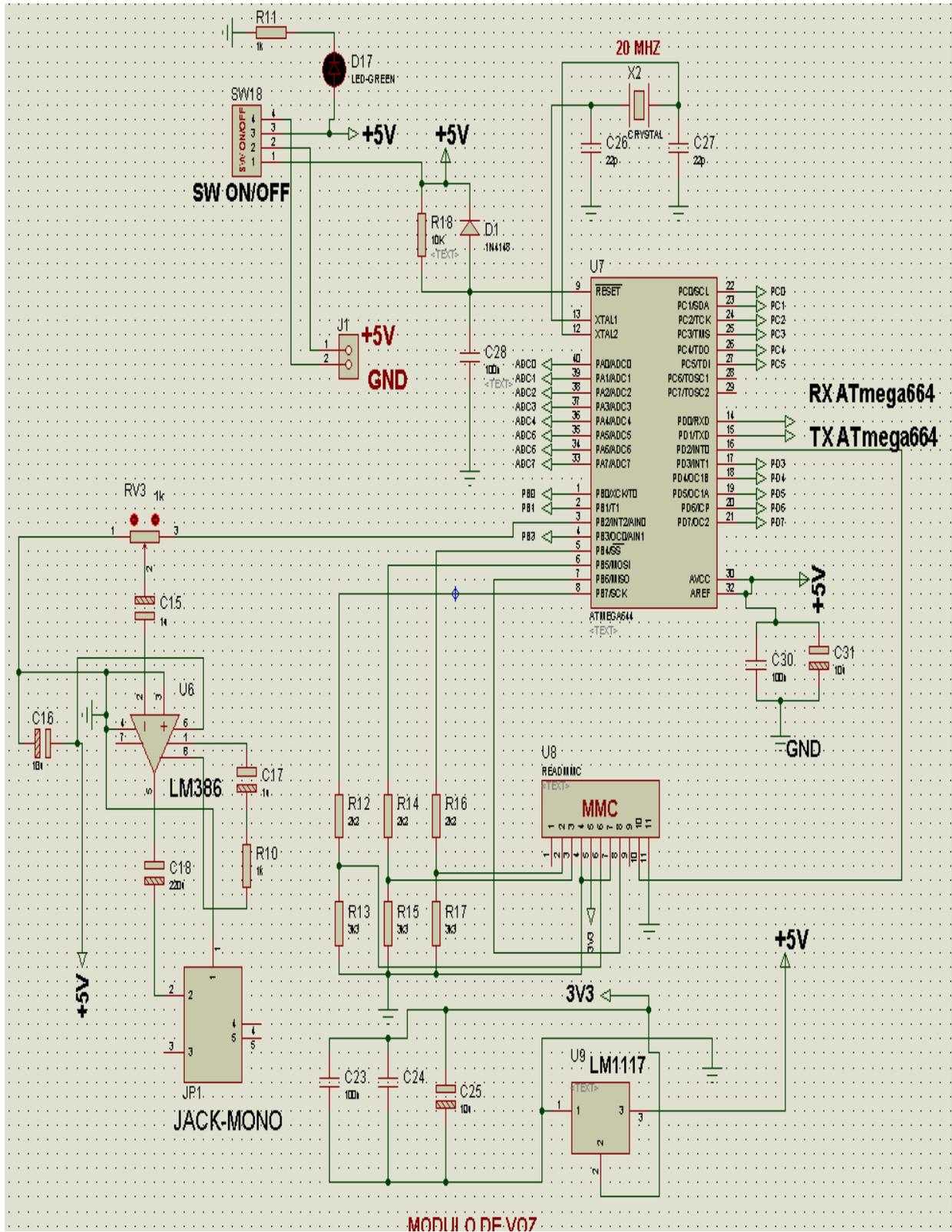


Figura 2. 5 Diagrama circuital del módulo de voz

2.6.2 INTRODUCCIÓN.

En el proceso de realización del bastón electrónico se hizo necesario un sistema capaz de guardar los mensajes de aviso, para que luego sean reproducidos. Para lograr este objetivo se utilizó un módulo de voz que consta de una memoria SD de 1GByte en donde se puede guardar cualquier número de mensajes porque solo depende de la capacidad de la memoria y con una salida de audio monofónica a audífonos.

2.6.3 DESCRIPCIÓN DEL MÓDULO DE VOZ

2.6.3.1 Direccionamiento de la memoria SD

La memoria SD se conecta a los pines de entrada y salida del microcontrolador ATMEGA 644 (PB4, PB5, PB6) que son los encargados de mandar a la dirección a la memoria SD mientras que el pin PB7 es el encargado de habilitar la lectura y escritura y el PB2 es el habilita el FAT 32.

Por ejemplo el microcontrolador ATMEGA 164 detecta un objeto frontal a la distancia de 50 cm, el sensor envía el dato al microcontrolador ATMEGA 164 el cual se comunica por medio de hardware al microcontrolador ATMEGA 644, por medio de software que está grabado en dicho microcontrolador, habilita la lectura manda la dirección de datos donde se encuentra grabado el mensaje de voz y envía la señal de audio al amplificador el cual amplificara dicha señal.

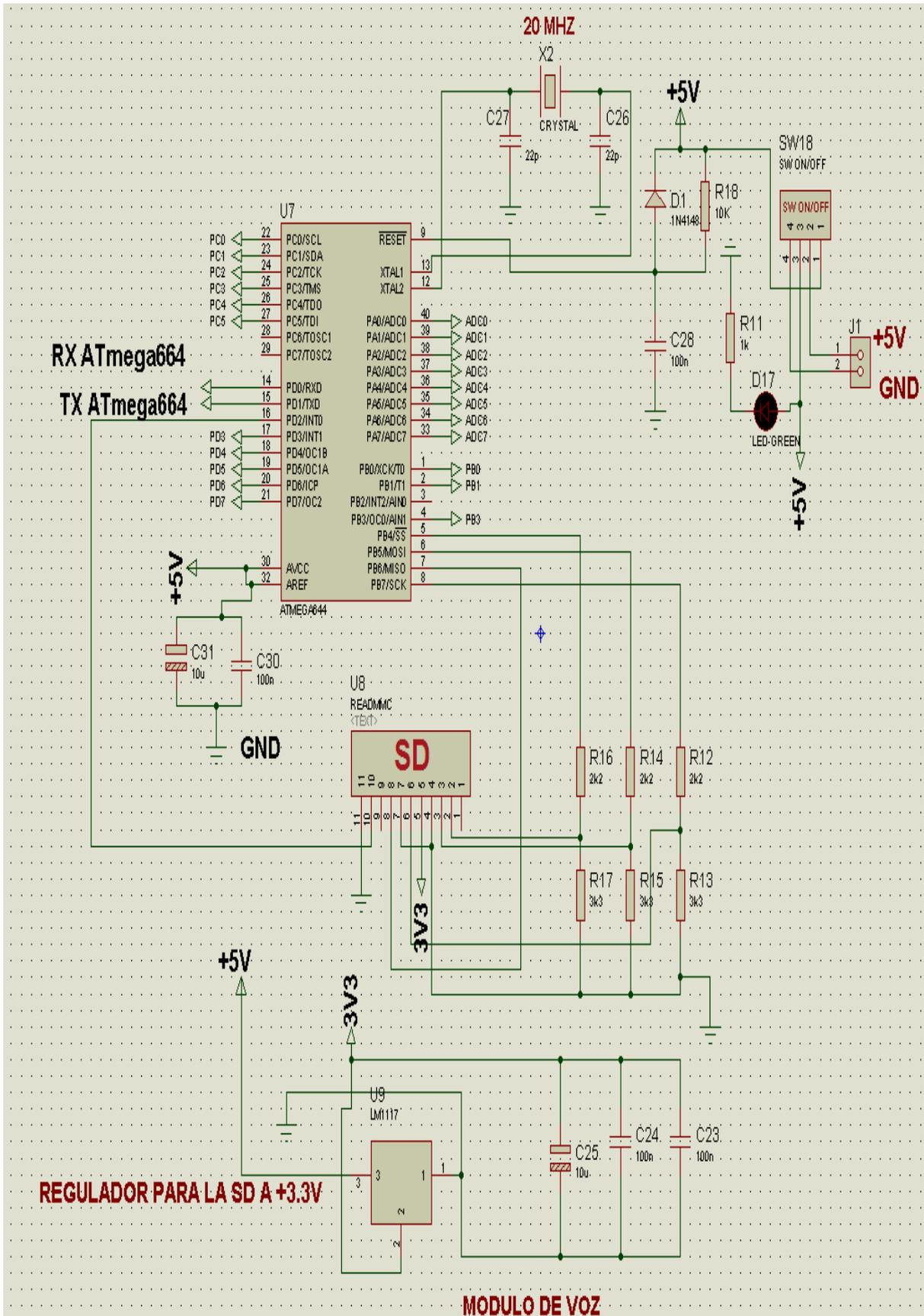


Figura 2. 6 Direccinamiento de la memoria SD

2.6.3.2 Amplificación del mensaje de voz

El CI. LM386 es el que utilizaremos para la amplificación de los mensajes de voz, dicho amplificador operacional se conecta en la entrada inversora a la salida del microcontrolador ATMEGA 644 del pin PB2, por medio de los potenciómetros RV2 y RV3 se regula la ganancia del amplificador, a la salida del amplificador se conecta un jack-monofónico para la conexión de los audífonos

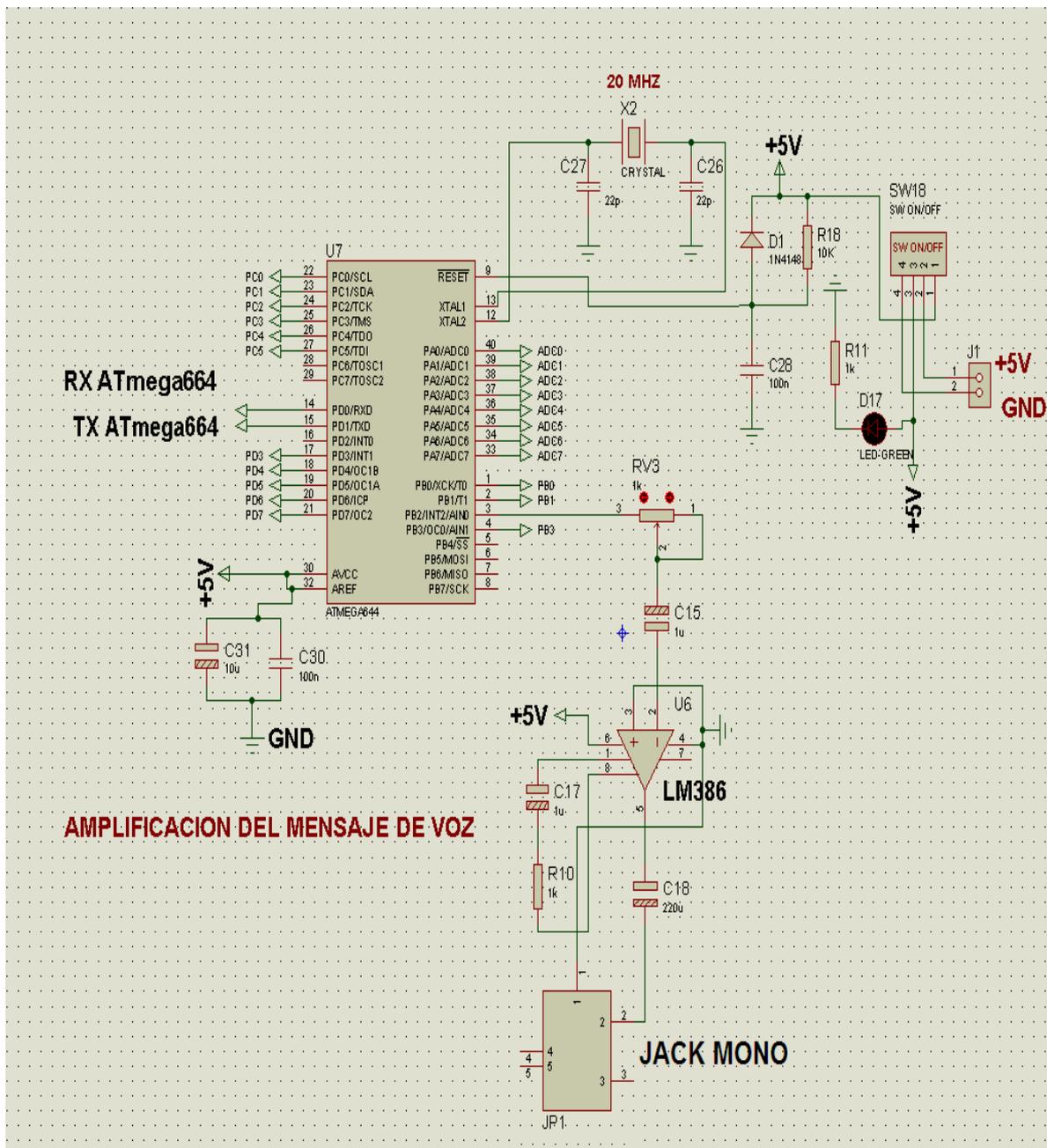


Figura 2. 7 Amplificación del mensaje de voz

2.6.4 PROCESO DE OBTENCIÓN Y GRABACIÓN DE LOS MENSAJES

- Grabación de los mensajes.
- Procesamiento, filtrado y ecualización de los sonidos en una PC.
- Transferencia de los sonidos desde la PC a la memoria flash ISD.

Posteriormente se tratará con mayor detalle cada uno de estos pasos.

2.6.4.1 Obtención de los sonidos.

Los mensajes fueron grabados EXPstudio Audio Editor mediante un micrófono, Una vez grabados los mensajes se procede a guardar en formato WAV a 32 KHz a 8 bits y monofónico.

Después de grabar los mensajes se procedió a editar con el programa EXPstudio Audio Editor, para recortar, filtrar, ecualizar, etc. una determinada señal de audio.

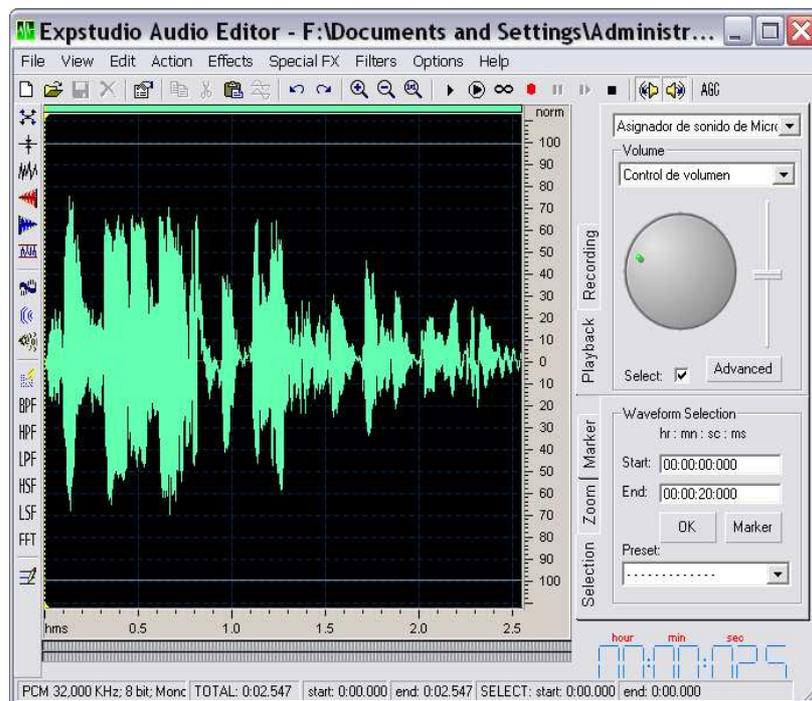


Figura 2. 8 EXPstuder audio editor

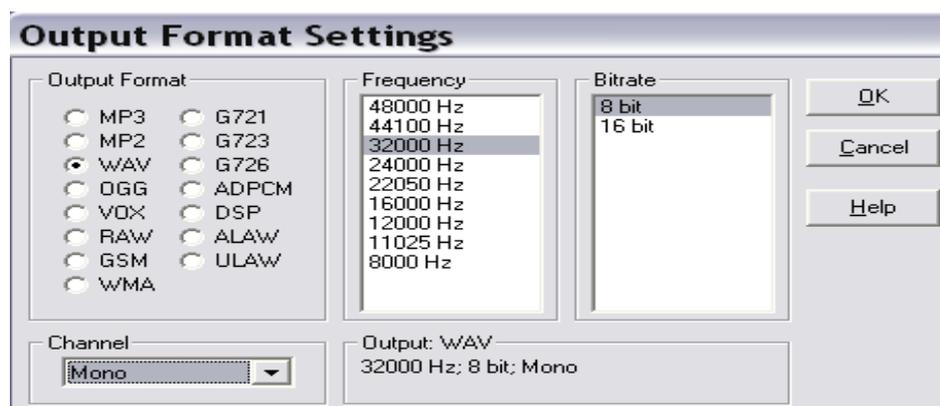


Figura 2. 9 EXPstudio audio editor formato de los mensajes voz

2.6.4.1.1 Nomenclatura y contenido de los mensajes de voz

MENSAJES: SENSOR UNO

Nombre	Comentario
S1005.WAV	Sensor uno objeto presente
S1015. WAV	Sensor uno obstáculo cercano tome las debidas precauciones
S1025.WAV	Sensor uno obstáculo a veinte centímetros
S1035.WAV	Sensor uno obstáculo a treinta centímetros
S1045.WAV	Sensor uno obstáculo a cuarenta centímetros
S1055.WAV	Sensor uno obstáculo a cincuenta centímetros
S1065.WAV	Sensor uno obstáculo a sesenta centímetros
S1075.WAV	Sensor uno obstáculo a setenta centímetros
S1085.WAV	Sensor uno obstáculo a noventa centímetros
S1095.WAV	Sensor uno obstáculo a cien centímetros

Tabla 2. 1 Nomenclatura y contenido de los mensajes de voz S1

MENSAJES: SENSOR DOS

Nombre	Comentario
S2005.WAV	Sensor dos objeto presente
S2015. WAV	Sensor dos obstáculo cercano tome las debidas precauciones
S2025.WAV	Sensor dos obstáculo a veinte centímetros

S2035.WAV	Sensor dos obstáculo a treinta centímetros
S2045.WAV	Sensor dos obstáculo a cuarenta centímetros
S2055.WAV	Sensor dos obstáculo a cincuenta centímetros
S2065.WAV	Sensor dos obstáculo a sesenta centímetros
S2075.WAV	Sensor dos obstáculo a setenta centímetros
S2085.WAV	Sensor dos obstáculo a noventa centímetros
S2095.WAV	Sensor dos obstáculo a cien centímetros

Tabla 2. 2 Nomenclatura y contenido de los mensajes de voz S2.

MENSAJES: SENSOR TRES

Nombre	Comentario
S3007.WAV	Sensor tres se ha detectado una inclinación por favor tomar las debidas precauciones.
S3020 .WAV	Sensor tres se ha detectado un escalón por favor tomar las debidas precauciones.
S3030.WAV	Sensor tres se ha detectado una inclinación profunda evite acercarse

Tabla 2. 3 Nomenclatura y contenido de los mensajes de voz.

Obtenidos ya los diferentes mensajes procesados, se continúa a copiar en la memoria SD de 1 Gbyte.

2.7 CONEXIÓN DE LOS DISPOSITIVOS EN LA TARJETA PRINCIPAL

2.7.1 CONEXIÓN DEL CIRCUITO RESET

Para el RESET de la tarjeta principal se utilizo el circuito de la figura 2.10 que se detallara más adelante.

La entrada MCLR permite reiniciar el estado del micro, llevándose a cabo dos acciones importantes:

- Se carga un 0 en el contador de programa, de forma que después de un reset siempre se ejecuta la instrucción que está en la posición 0 de la memoria de programa.
- Los registros de estado y control toman un estado conocido y determinado.

Existen dos circuitos muy usados para RESET los cuales se indican en la figura 2.10 y en la figura 2.11

- En este circuito el pulsador normalmente abierto está conectado en paralelo con C

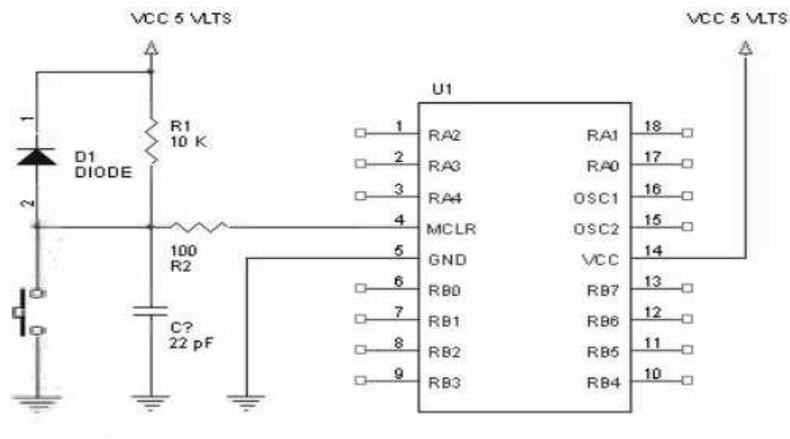


Figura 2. 10 Circuito reset (1)

- En este circuito el pulsador normalmente abierto está conectado como un divisor de tensión.

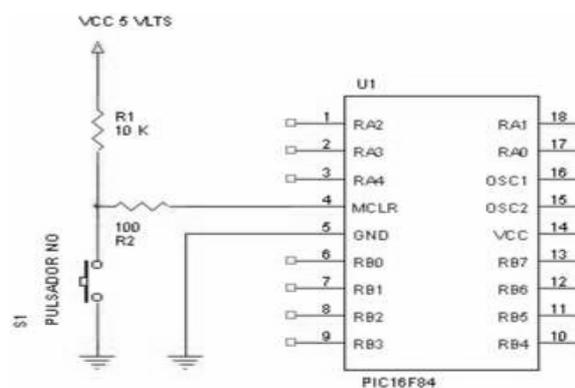


Figura 2. 11 Circuito reset (2)

2.7.2 CONEXIÓN DE OSCILADORES

Para el oscilador de la tarjeta principal se utilizó un cristal de 16Mhz, capacitores de 22pF y de tipo XT que se detallara más adelante.

La frecuencia de trabajo viene dada por el oscilador externo.

Los microprocesadores admiten cuatro tipos de osciladores:

- Oscilador RC: Oscilador de bajo costo formado por una resistencia y un condensador, cuyos valores determinan la frecuencia de oscilación. Proporciona una estabilidad mediocre.
- Oscilador HS: Basado en un cristal de cuarzo, alcanza una velocidad entre 4 y 10 MHz.
- Oscilador XT: Oscilador de cristal o resonador para frecuencias entre 100 KHz y 4 MHz.
- Oscilador LP: Oscilador de bajo consumo, con cristal o resonador para frecuencias entre 35 y 200 KHz.

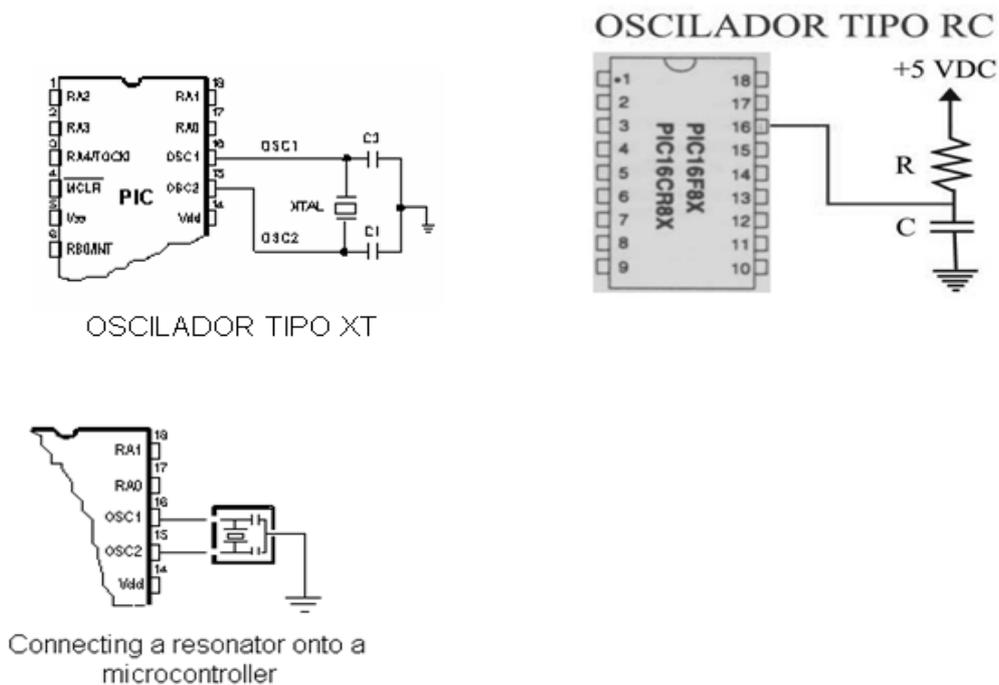


Figura 2. 12 Conexión de osciladores

2.7.3 CONEXIÓN DEL LCD

El LCD es utilizado para mostrar las medidas de distancias de los sensores de ultrasonido en cm.

Los LCD se pueden conectar con el AVR (ATMEGA 164) con un bus de 4 u 8 bits, la diferencia está en el tiempo que se demora, pues la comunicación de 4 bits, primero envía los 4 bits más altos y luego los 4 bits más bajos, mientras que la de 8 bits envía todo al mismo tiempo.

Las conexiones empleadas en nuestro circuito serán la del bus de datos de 4 bits y las de los pines de control, es decir, E, RS y RW. Debido a esto, el microcontrolador deberá utilizar 7 bits para controlar la pantalla como se indica en la figura 2.13.

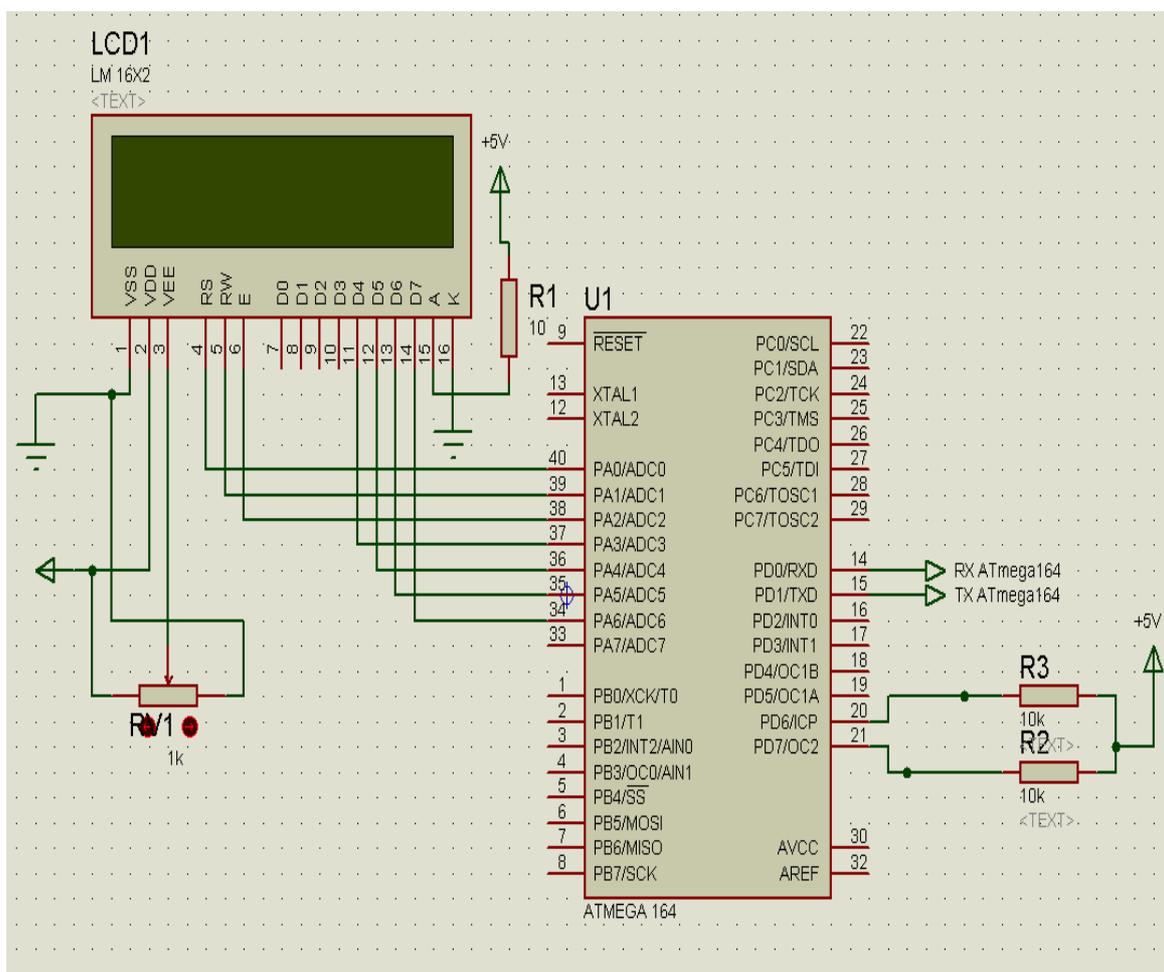


Figura 2. 13 Conexión del LCD

2.7.4 CONEXIÓN DE LOS SENSORES

En este circuito se utiliza para la detección de obstáculos y agujeros, sensor de distancia por ultrasonido ubicados en la parte frontal e inferior del bastón como se indica en la figura 2.14.

Sensores de distancia por ultrasonido SRF08 están conectados al microprocesador ATMEGA 164 por medio del bus I2C que solo utiliza dos pines del microprocesador, está conectado en el Pin PD6 y PD7 las salidas del sensor de distancia por ultrasonido SCL Y SDA respectivamente

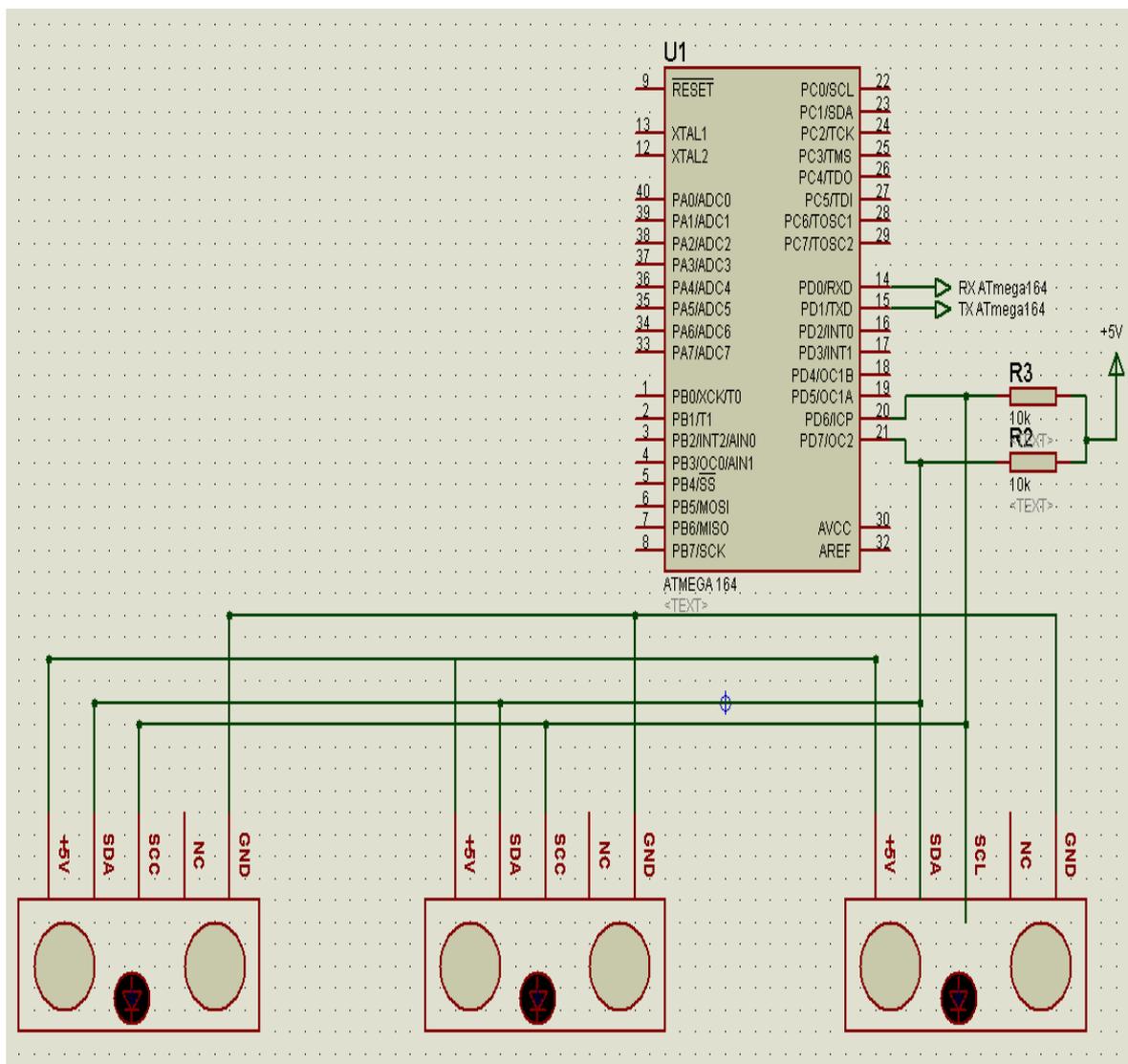


Figura 2. 14 Conexión de los Sensores

2.7.5 CONEXIÓN ENTRE LOS DOS MICROPROCESADORES (ATmega 164 y ATmega 644)

El microcontrolador ATMEGA 164 se comunica con el microcontrolador ATMEGA 644 serialmente por medio de hardware, la transmisión del microprocesador ATMEGA 164 con la recepción del microprocesador ATMEGA 644 y viceversa.

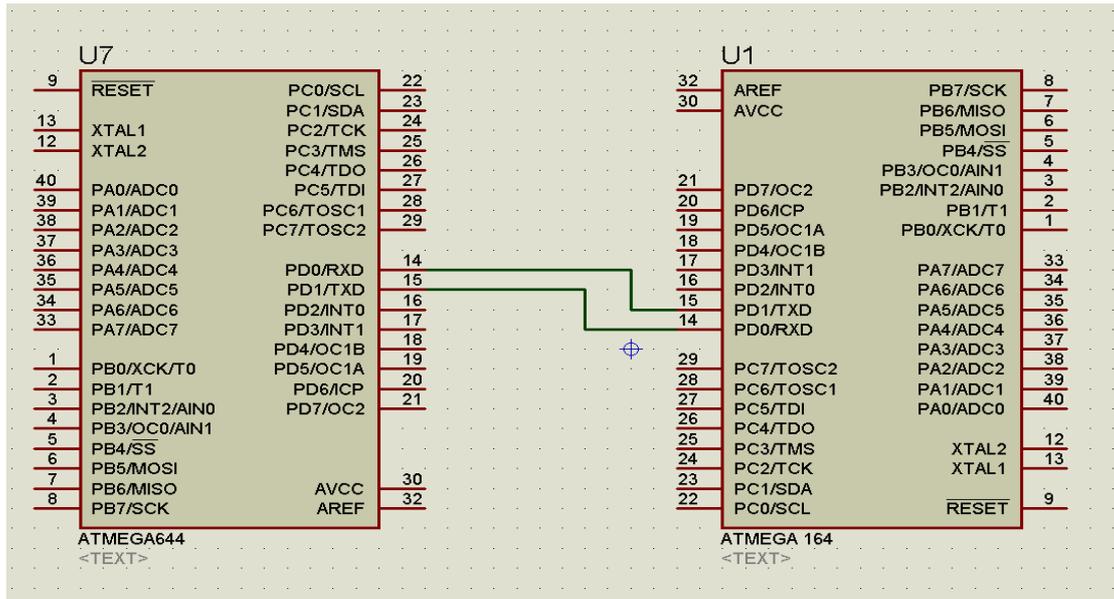


Figura 2. 15 Conexión entre los dos microprocesadores

2.7.6 CARGADOR DE LA BATERÍA

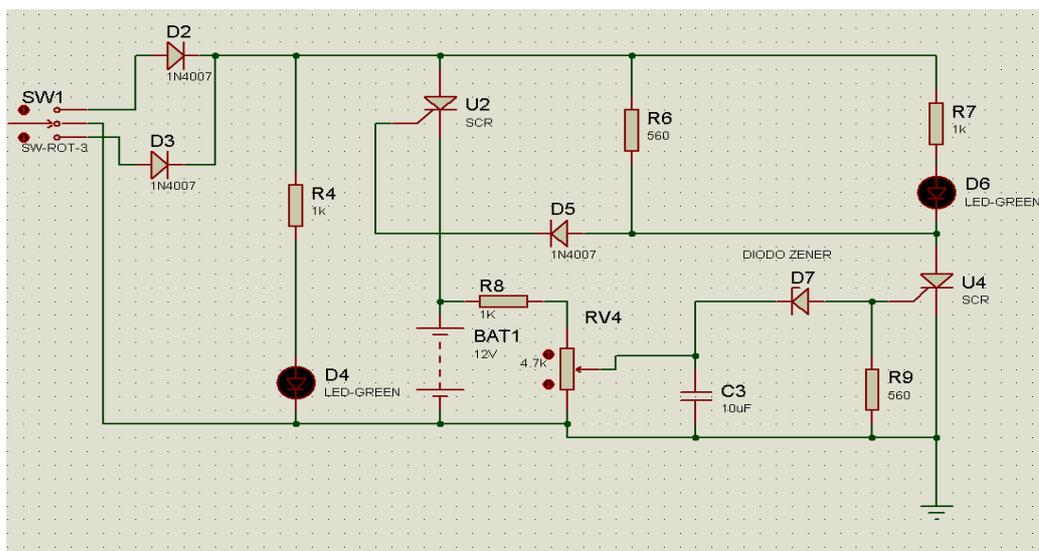


Figura 2. 16Cargador de batería

Es el encargado de cargar la batería de 12V con que funciona el sistema la cual entrega la alimentación a todos los circuitos del bastón electrónico.

El circuito utilizado es automático, avisando, mediante el accionamiento de un led o sistema de aviso, que la batería se encuentra cargada. El circuito es para baterías de 12V.

2.7.6.1 Descripción

La tensión alterna de la red de alimentación es aplicada al bobinado primario de un transformador, pasando por un fusible de protección y por una llave selectora de tensiones.

En el secundario del transformador tenemos una tensión alterna de 15 V, rectificada en onda completa por dos diodos. Como cada diodo sólo conduce la mitad del ciclo para una corriente de 5A tenemos una corriente media de sólo 2,5A, lo que significa que diodos de 4A soportan perfectamente este servicio. No filtramos esta tensión, pues con una tensión continua pulsante, el SCR puede ser desconectado al cortarse la tensión de su compuerta, lo que no ocurriría con una tensión continua pura.

La compuerta del SCR es polarizada por medio de R3 y D3, en el sentido de conducir la corriente siempre que la tensión en cada hemiciclo alcanza aproximadamente 1V.

En la compuerta de este SCR1 tenemos el circuito sensor de carga, formado básicamente por un divisor de tensión, un diodo zener y un segundo SCR (SCR2).

Ajustamos el potenciómetro P1 para obtener la tensión de disparo del SCR2 que, en este caso, corresponde a la tensión zener de D4 cuando la batería esté completamente cargada. Para una batería de 6V, el diodo zener debe ser de 2-1V ó 2.4V. Cuando la batería presente, entre sus terminales, la tensión que corresponde a la carga completa, el diodo D4 conduce y el SCR2 es disparado.

En estas condiciones, el mismo prácticamente pone a tierra la compuerta de SCR1, impidiendo el disparo de este componente, y por lo tanto, interrumpiendo la carga. Al mismo tiempo, la conducción plena de SCR2 hace que el LED2 sea alimentado, así como el circuito oscilador de aviso.

Como la fuente no es filtrada, para el oscilador de aviso precisamos una alimentación separada, pero este circuito es opcional.

El resistor Rx puede ser agregado si la corriente inicial de carga de los acumuladores supera el valor deseado o limitado por el fabricante. Debe ser usado un resistor de alambre de 1 a 10 ohmios con disipación de 10W.

2.7.6.2 Montaje

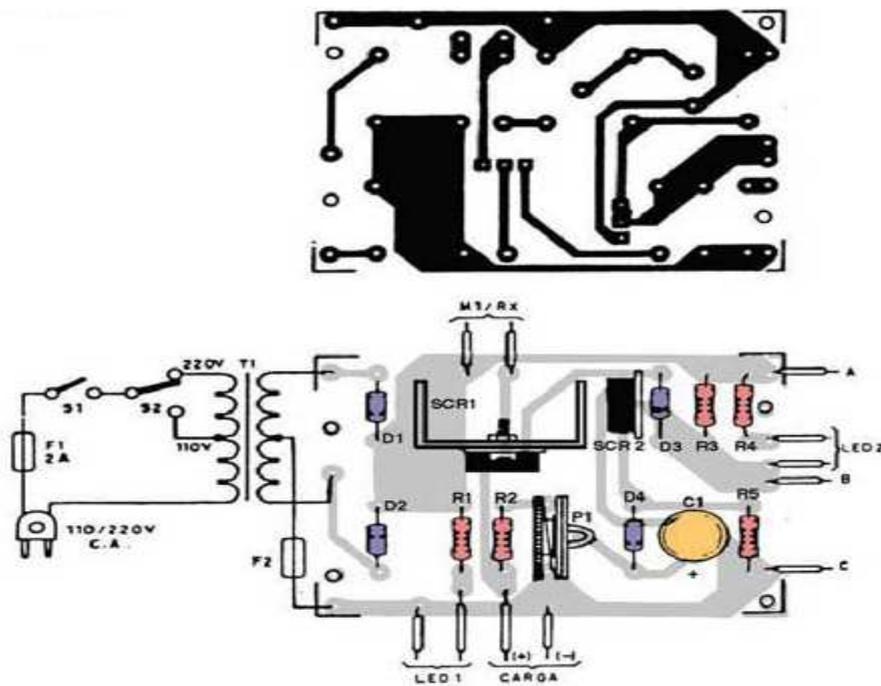


Figura 2. 17 Placa del cargador de batería.

La mayoría de los componentes es de grandes dimensiones y las corrientes en muchos puntos del circuito son intensas, lo que exige el empleo de la placa de circuito impreso adecuada. En la figura mostramos otra versión de placa de circuito impreso, hecha con el programa PCB Wizard 3.

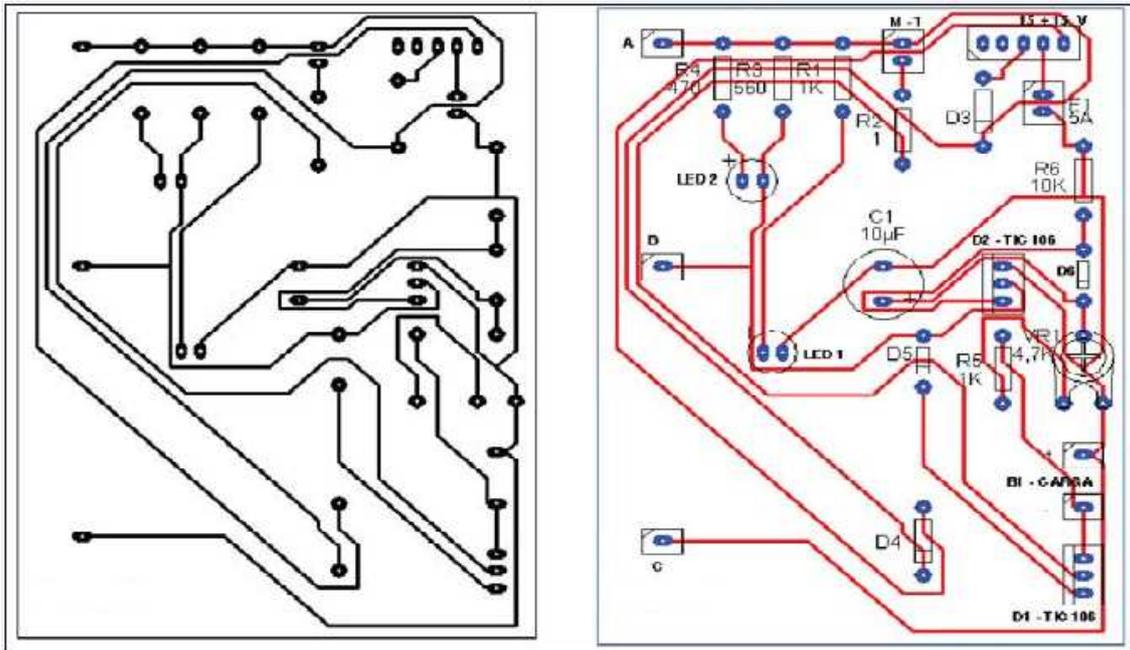


Figura 2. 18 Circuito impreso del cargador de batería.

Los leds son comunes, pudiendo ser de cualquier color.

Como sistema de aviso puede usar cualquier chichara u oscilador que funcione con 12V.

2.7.7 REGULADOR DE VOLTAJE A 5V

Para la etapa de alimentación de la tarjeta principal y el módulo de voz se necesita una fuente de alimentación de 5 voltios ya que la batería que utilizamos es de 12V por lo cual empleamos un regulador de voltaje 7805, este circuito integrado entrega un voltaje fijo estable a pesar de que existan variaciones en el voltaje de entrada, es decir regula la tensión de alimentación a 5V y 5mA y tiene una corriente de entrada de 1.5A. Este CI puede ser conectado con una tensión de alimentación de hasta 30V. A mayor tensión en su entrada, mayor calentamiento del integrado. Si la tensión de la alimentación es superior a 15 V, es aconsejable conectar una pequeña placa de material metálico (aluminio), para disipar el calor generado.

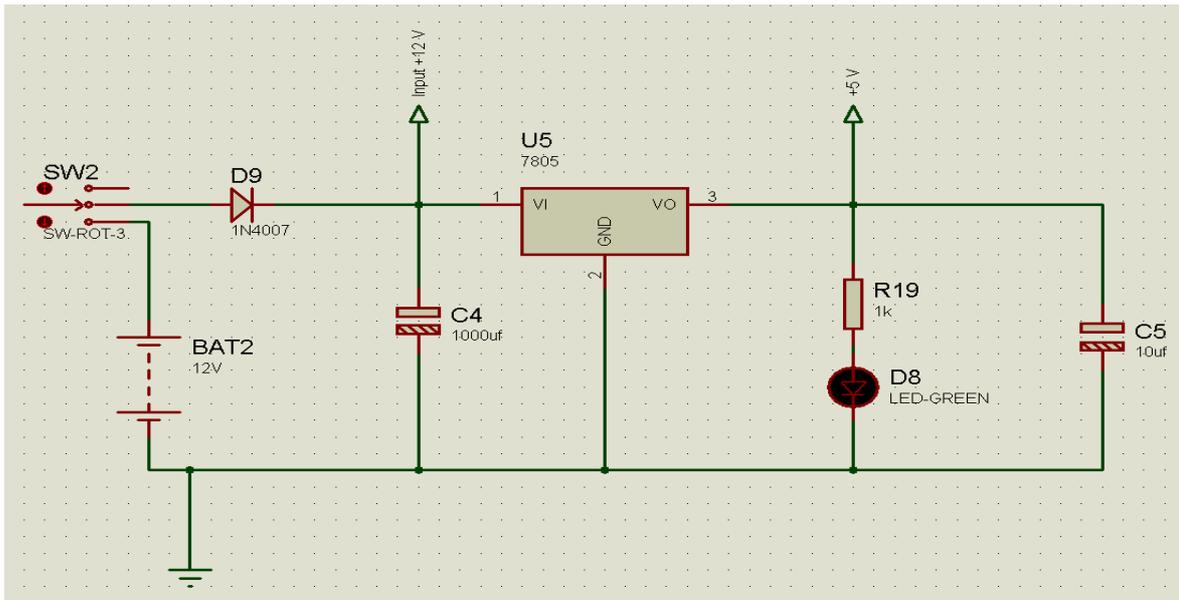


Figura 2. 19 Regulador de voltaje a 5V.

2.7.8 DISEÑO DEL CIRCUITO ESQUEMÁTICO

El programa utilizado para realizar la simulación y la elaboración de las PCB de nuestro proyecto es el Proteus 7.5 y en este capítulo se observa cómo lo hemos realizado el proyecto.

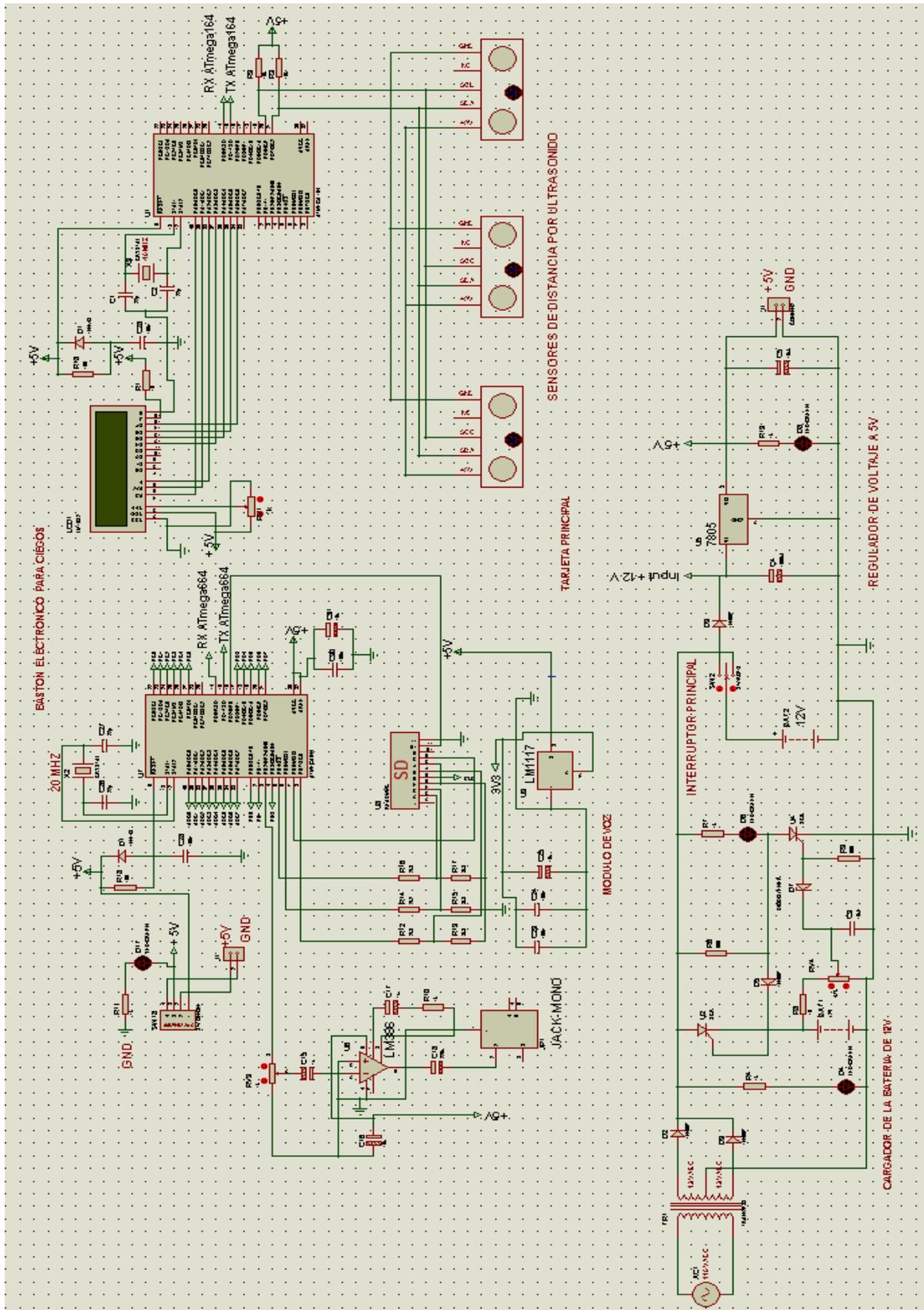


Figura 2. 20 Circuito Esquemático.

2.8 PASOS PARA OBTENER EL CIRCUITO ESQUEMÁTICO

En este documento les vamos a explicar cómo se diseña el circuito electrónico que hemos realizado. En primer lugar abrimos el programa Proteus, haciendo doble clic en el icono de acceso directo del escritorio o abriéndolo desde el menú inicio como se indica a continuación.



Figura 2. 21 Primer paso a seguir.

Cuando tengamos esta ventana, es ahí donde pondremos nuestro esquema lógico del circuito.

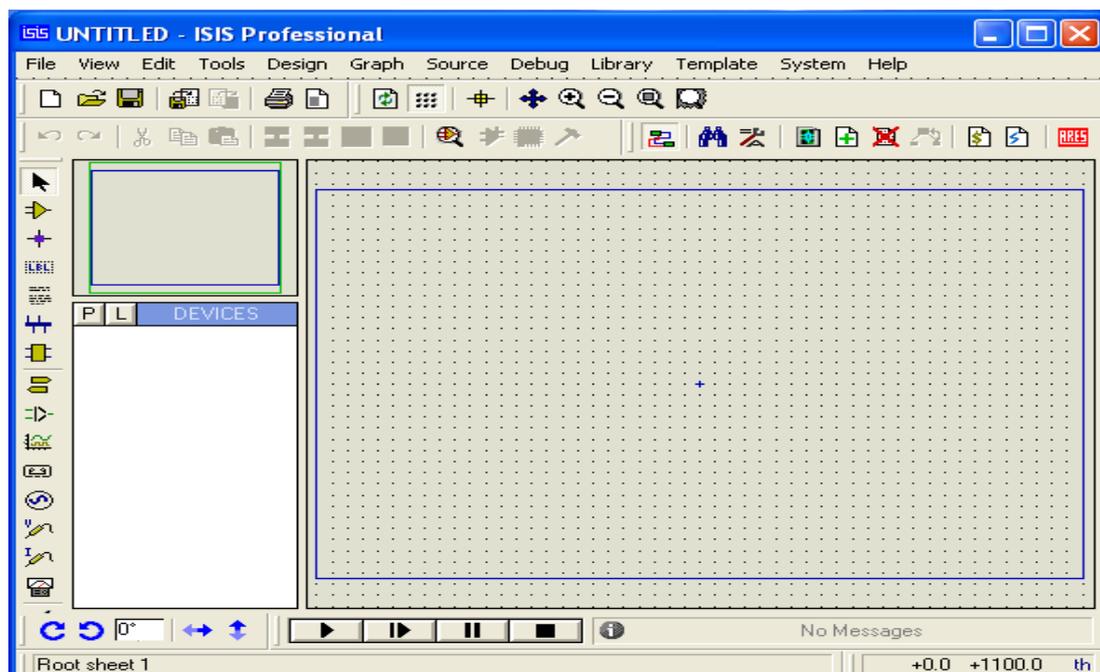


Figura 2. 22 Ventana ISIS

Una vez abierta, nos iremos al menú de la izquierda en donde aparece las letras p y l devices y seleccionaremos el botón en ese momento se abrirá una ventana donde iremos seleccionando los componentes de nuestro diseño lógico.

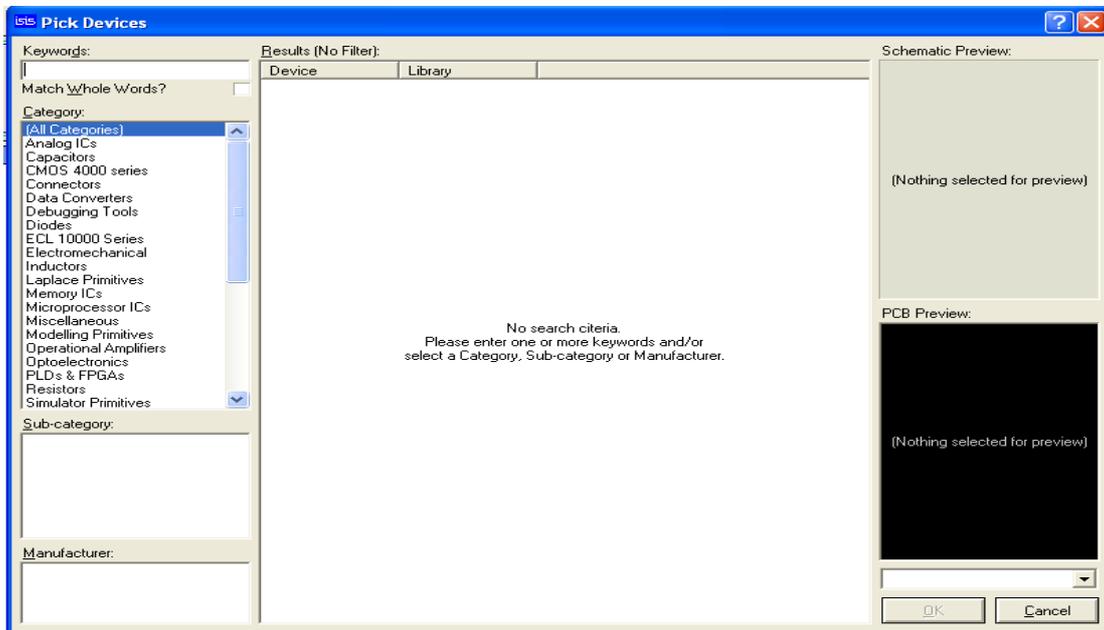


Figura 2. 23 Ventana PICK DEVICES

Dependiendo que categoría sea la seleccionada se abrirá la siguiente pantalla de elementos a escoger.

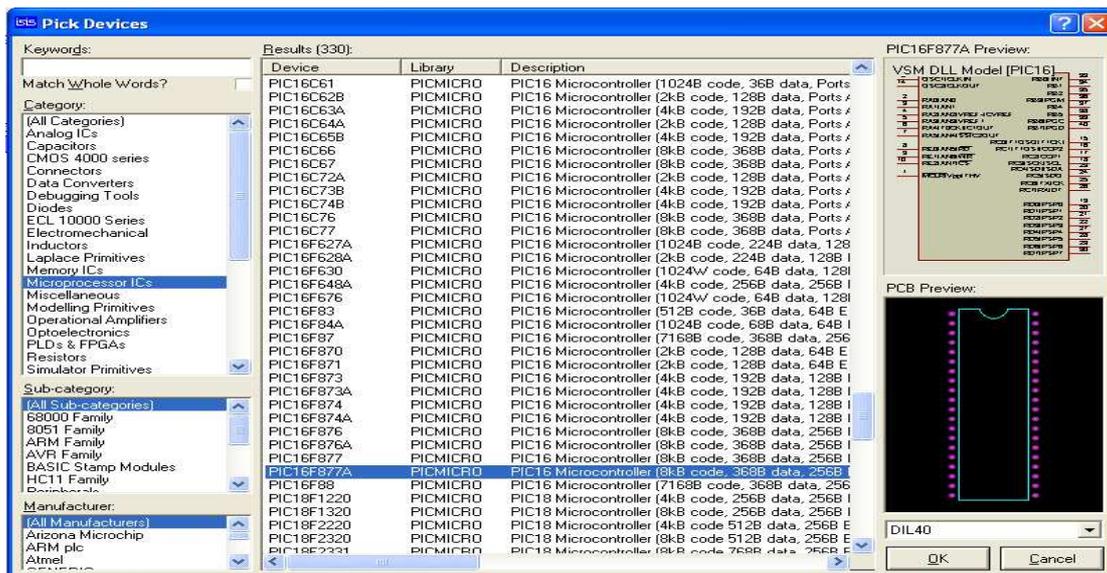


Figura 2. 24 Ventana PICK DEVICES con elementos

Cuando hemos seleccionado todos los componentes que vamos a utilizar en nuestro circuito, dichos componentes aparecerán en la ventana de dispositivos.

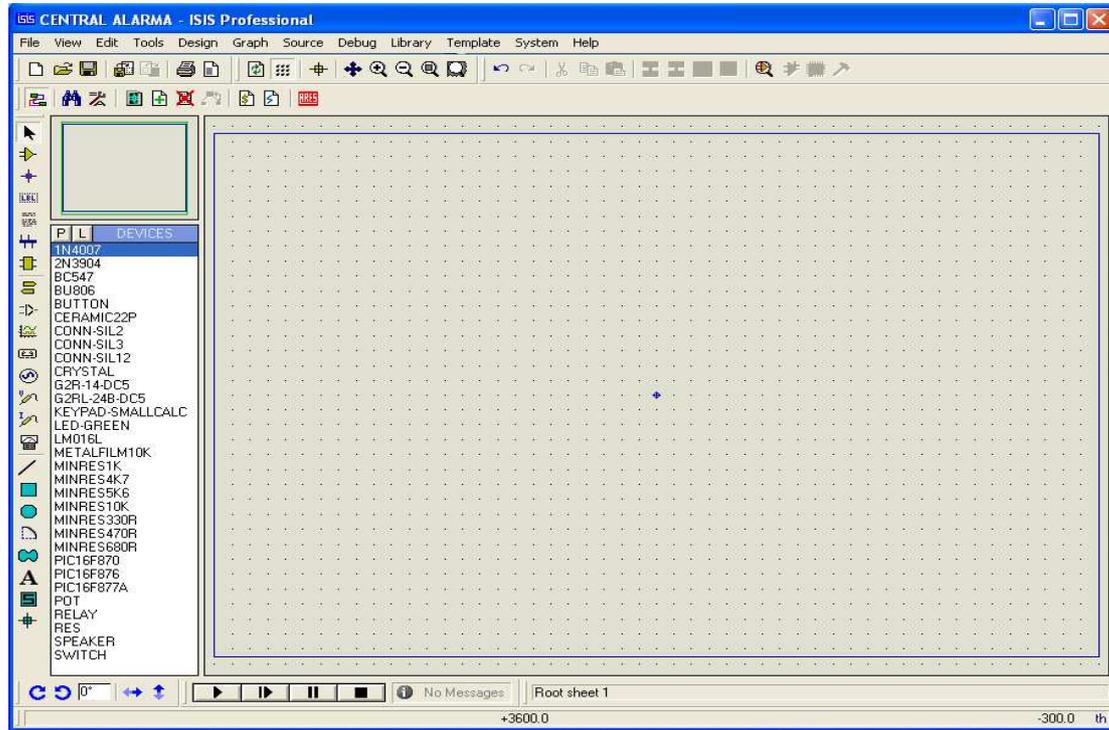


Figura 2. 25 Ventana ISIS con elementos.

Y al final procedemos a realizar las conexiones correspondientes para poder realizar la simulación.

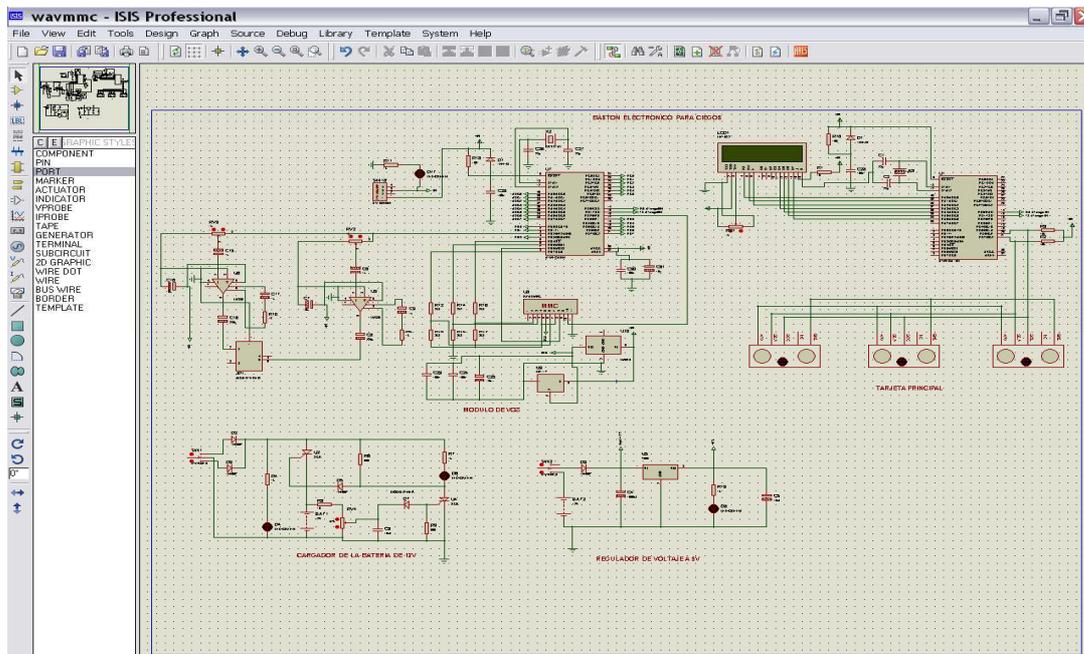


Figura 2. 26 Circuito Esquemático en ISIS

2.9 DISEÑO DE LAS PLACAS DEL CIRCUITO IMPRESO (PCB)

Una vez que ya se ha realizado la simulación del circuito lógico con su respectivo programa, procedemos a realizar el ruteado en el ARES, para ello presionamos el ARES  en la barra de herramientas del ISIS.

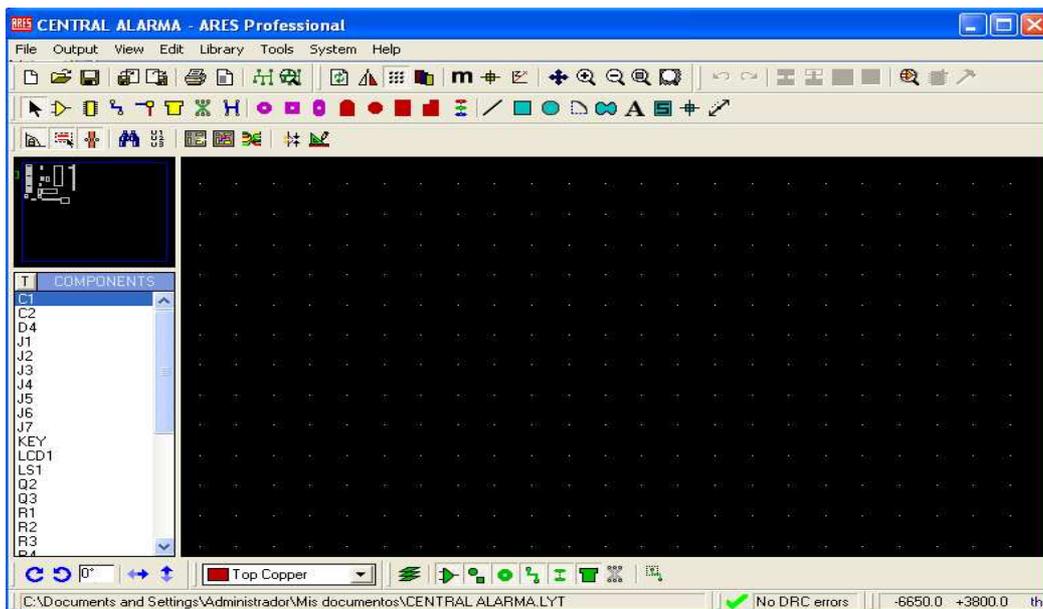


Figura 2. 27 Modulo de voz en ares (1)

Una vez que ya tenemos el ARES con todos los elementos procedemos a ubicarlos de acuerdo a nuestra conveniencia.

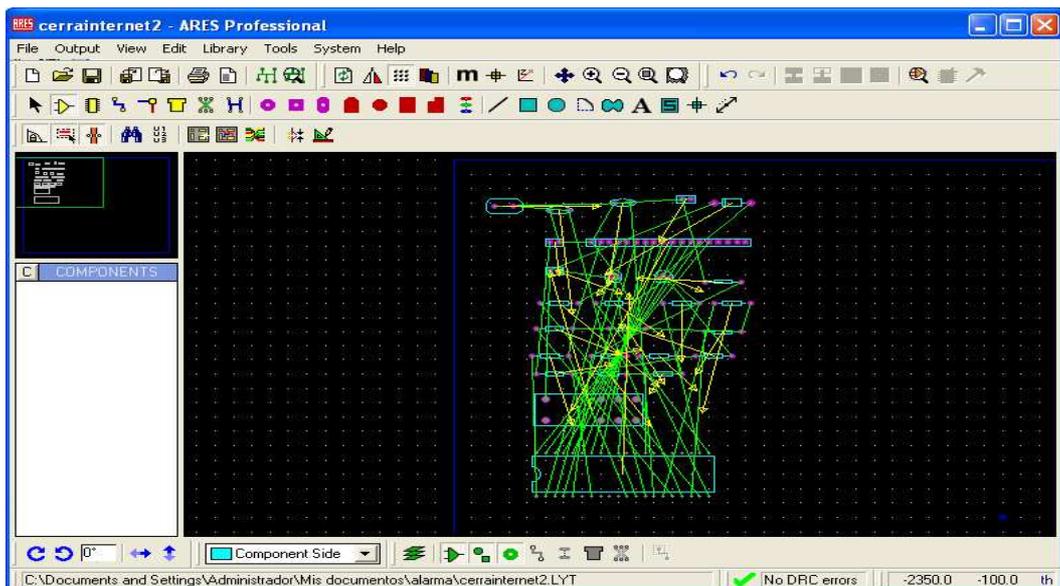


Figura 2. 28 Modulo de voz en ares (2)

Una vez que los componentes ya se encuentran correctamente ubicados procedemos a rutiar el circuito.

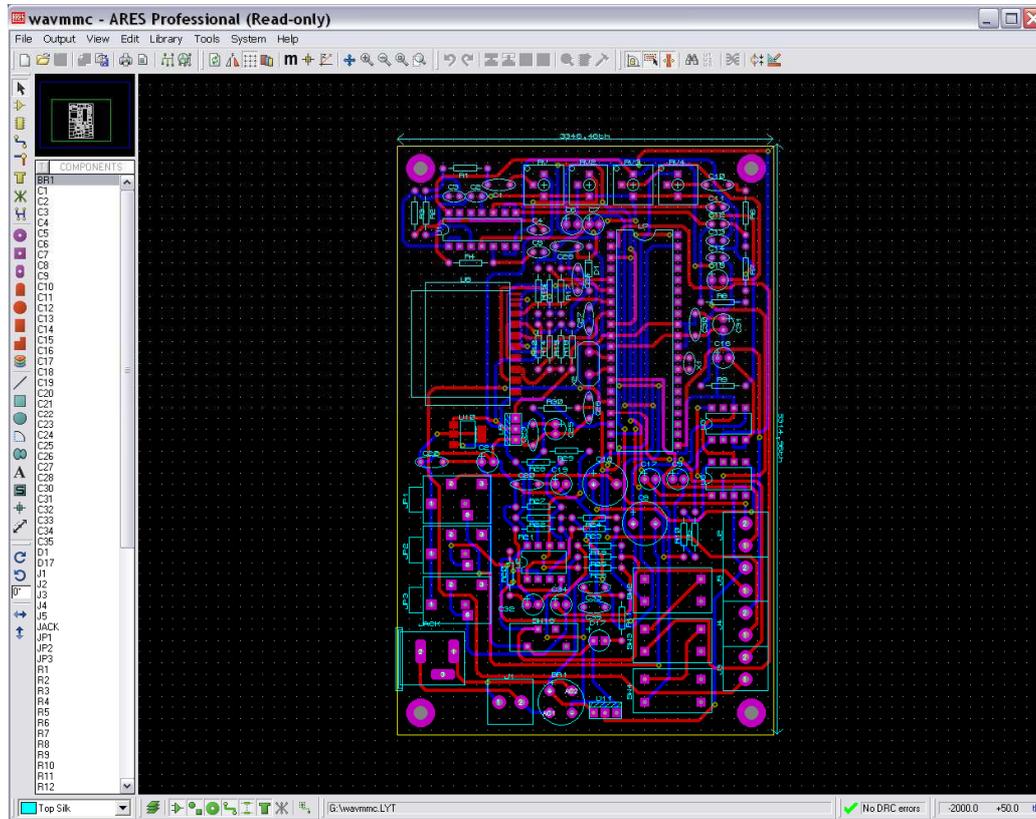


Figura 2. 29 Módulo de voz en ares (3)

Cuando el PCB ya esté listo procedemos a imprimir en un papel de termo transferencia.

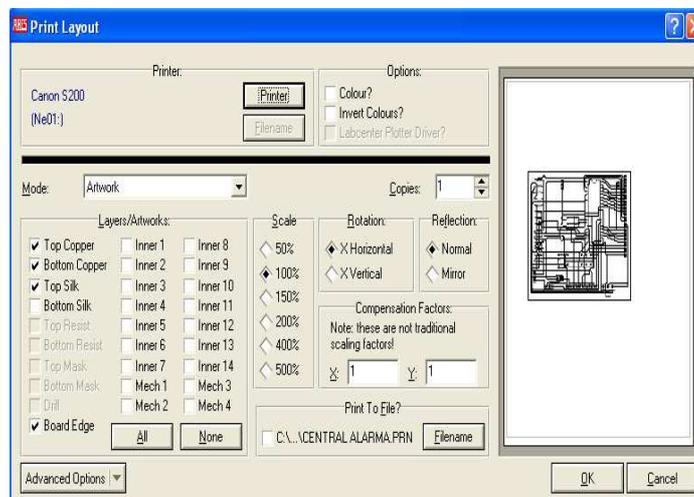


Figura 2. 30 Diseño de impresión.

2.10 CREACIÓN DE LOS CIRCUITO IMPRESOS

Para comenzar se debe realizar el diseño del diagrama circuital para luego realizar el circuito impreso, en cualquier programa que acostumbremos usar, tales como EAGLE, PROTEL, PROTEUS, OrCAD, entre otros. Para el presente proyecto de titulación se utilizo el programa Proteus para Windows en el cual se desarrollo todos los diagramas circuitales.

Solo se va explicar el diseño de la placa "PRINCIPAL"

Las razones porque escogimos este programa son las siguientes.

- El programa se puede instalar en una computadora que tenga como sistema operativo Windows Xp, no necesita de muchos recursos del sistema
- Si no existe un elemento en la librería se lo puede uno mismo crear y guardarlo en la librería para una próxima utilización
- El algoritmo del programa no es complejo sino amistoso con el usuario
- Fácil de manejo.

Después de realizar el diagrama circuital como se observa en la figura 2.18 se procede a realizar el trazado de las pistas para ello se utilizo el programa de **ARES**, en la pantalla una vez que ya témenos todos los elementos procedemos a ubicarlos de acuerdo a nuestra conveniencia y necesidad, teniendo en cuenta las medidas de la placa, las cuales son de 13.8cm de largo y 8.8cm de ancho.

Una vez que esté bien colocado los elementos dependiendo de los requerimientos de conexión con las demás placas y la maqueta del bastón se procede a rutear y como resultado final se tiene el diagrama de pistas.

Para imprimir solo las pistas y no los elementos se tiene que desactivar Top Cooper.

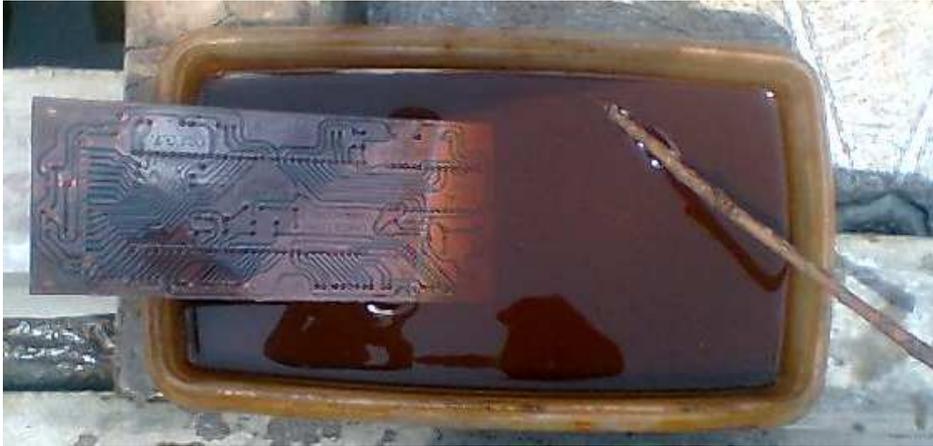


Figura 2. 34 Baño de cloruro férrico.

Para retirar el cobre sobrante, es decir lo que no está protegido por el tóner, se necesita hacer un baño de cloruro férrico, como se indica en la figura 2.22. Se introduce la placa en la solución ya preparada, en un tiempo de 40 minutos (dependiendo de la concentración y de la temperatura).

Una vez que la solución termina de eliminar el cobre expuesto, se retira la placa y se lava con abundante agua del grifo, en esta fase se ven de color negro.

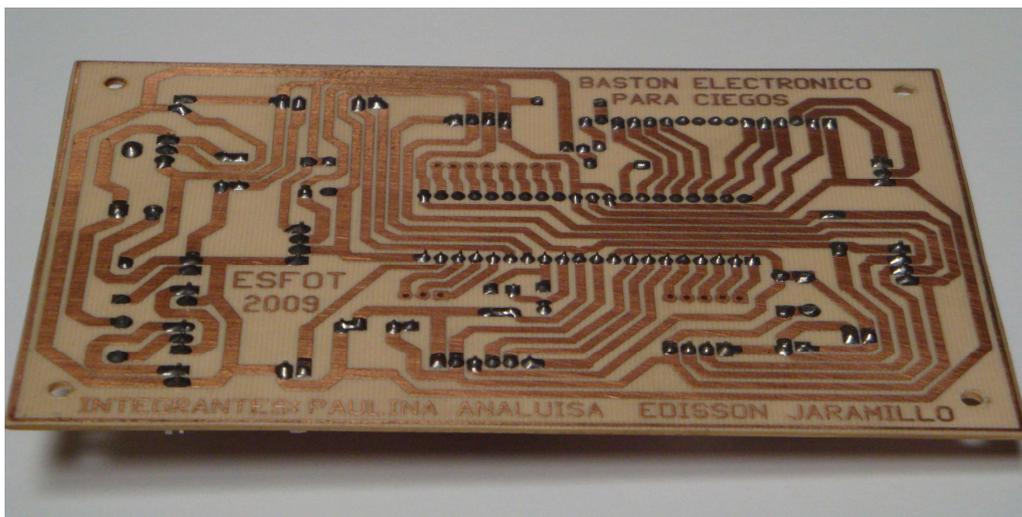


Figura 2. 35 Tarjeta principal sacada del acido y lavada.

Luego solo resta quitar la tinta para que aparezca el cobre, usando nuevamente lana de acero y cubrir el cobre con resina vegetal disuelta en alcohol o barniz para evitar que se oscurezca como resultado da la placa final que se indica en la figura 2.23

El screen de elementos no es más que texto, información, datos y figuras que indican el lugar donde se debe insertar los elementos electrónicos, para lo cual se tiene que imprimir en acetato, colocar al otro lado de la baquelita en el lugar donde no hay cobre, a la misma altura y posición que están las pistas teniendo en cuidado de no colocar al revés

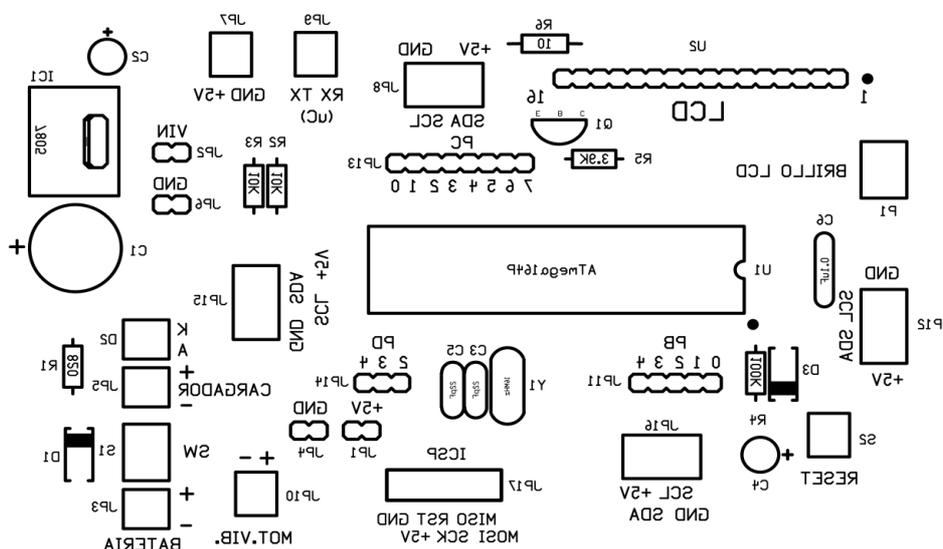


Figura 2. 36 Screen de elementos placa cerebro

Como último paso realizar la perforación de los huecos, en donde van a ir los elementos electrónicos, para después soldarlos.

2.11 DESCRIPCIÓN DE LAS TARJETAS DEL BASTÓN

2.11.1 BASTÓN ELECTRÓNICO

Está conformada por el sistema de detección de obstáculos (sensores de distancia por ultrasonido), audífonos, ruedas, LCD, estructura física.

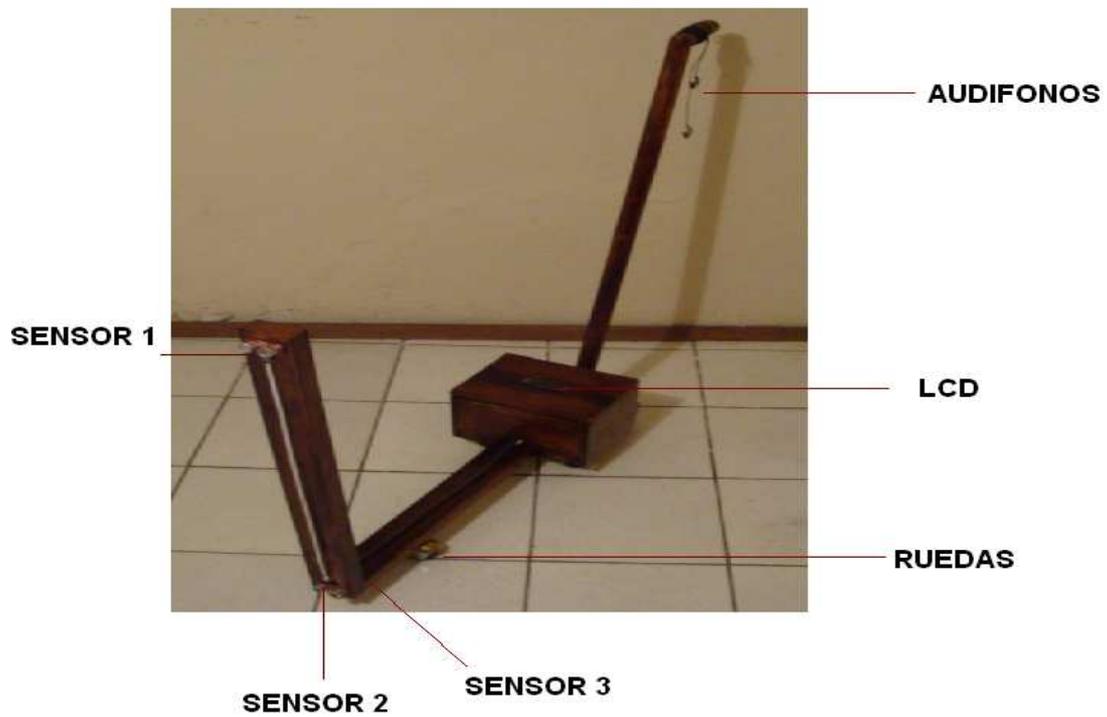


Figura 2. 37 Bastón vista frontal

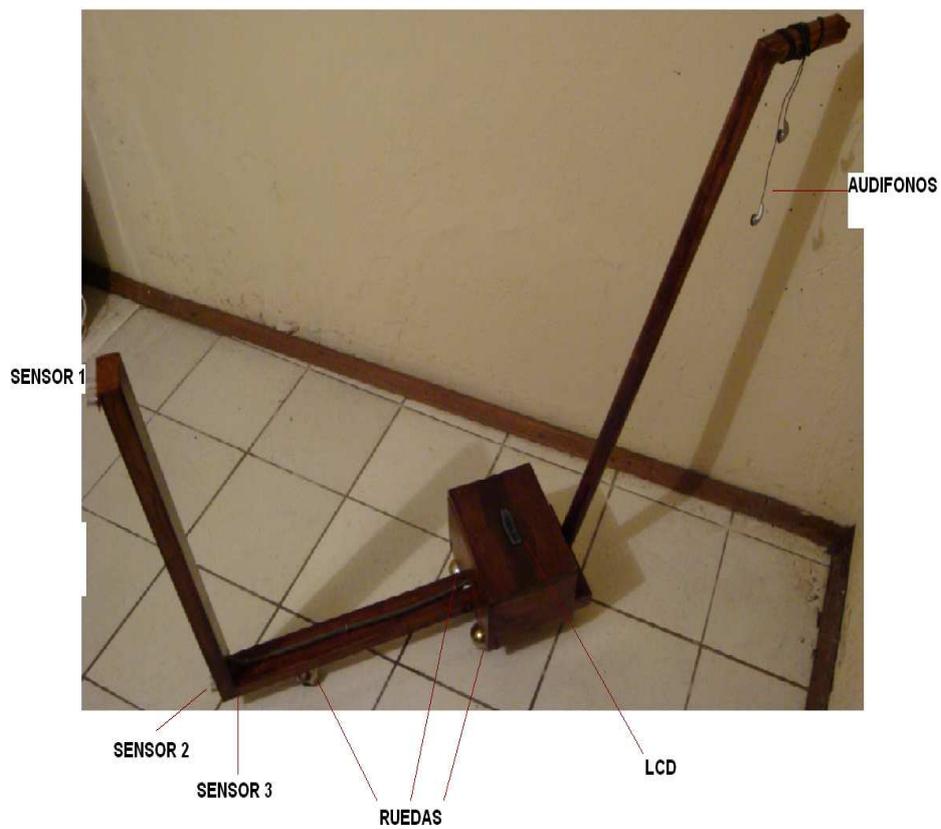


Figura 2. 38 Bastón vista lateral.

2.11.2 TARJETA PRINCIPAL

Está conformada por el sistema de detección de obstáculos, un conector para la fuente de alimentación al circuito de la siguiente etapa, etc.

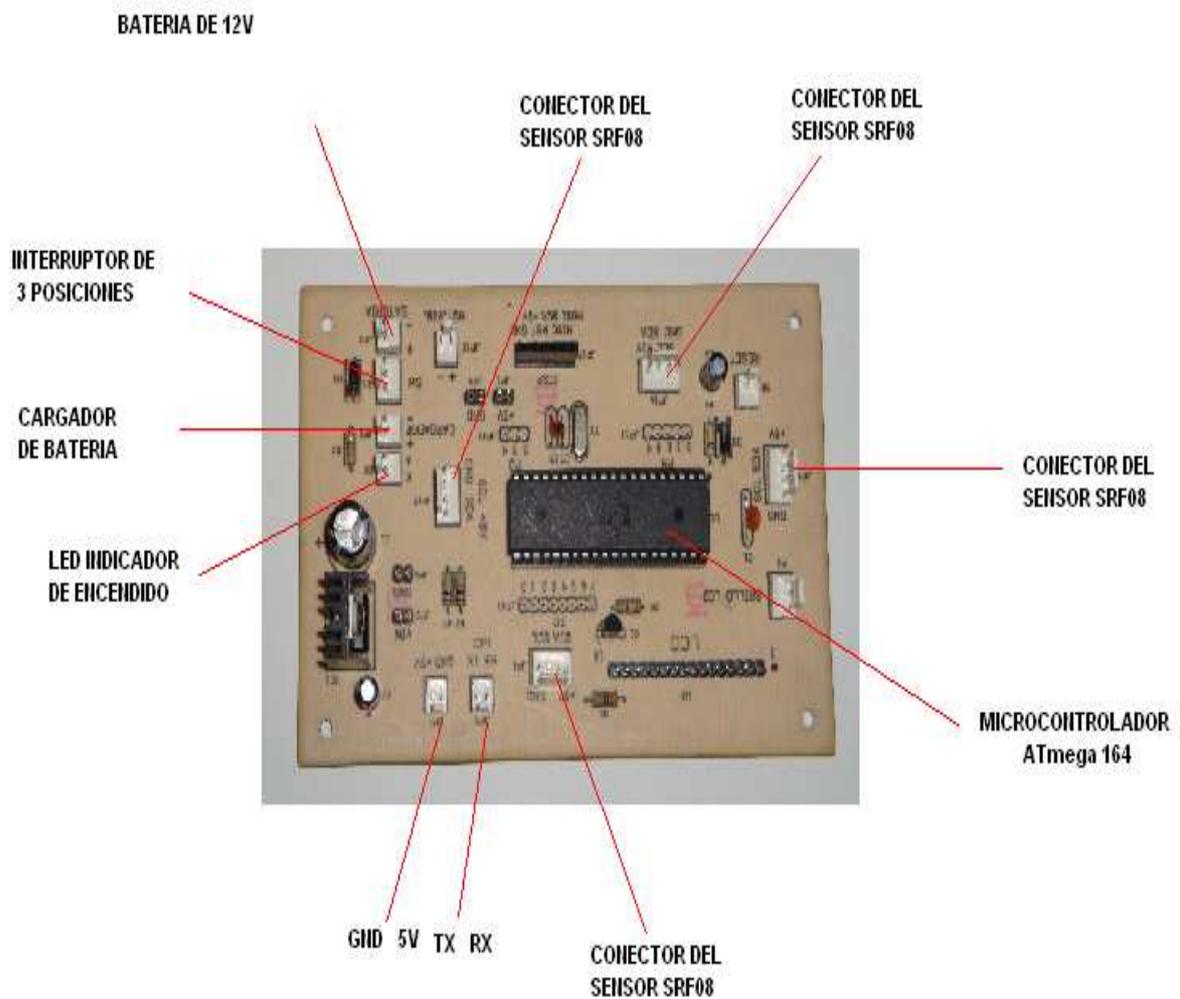


Figura 2. 39 Tarjeta principal

2.11.2 MODULO DE VOZ

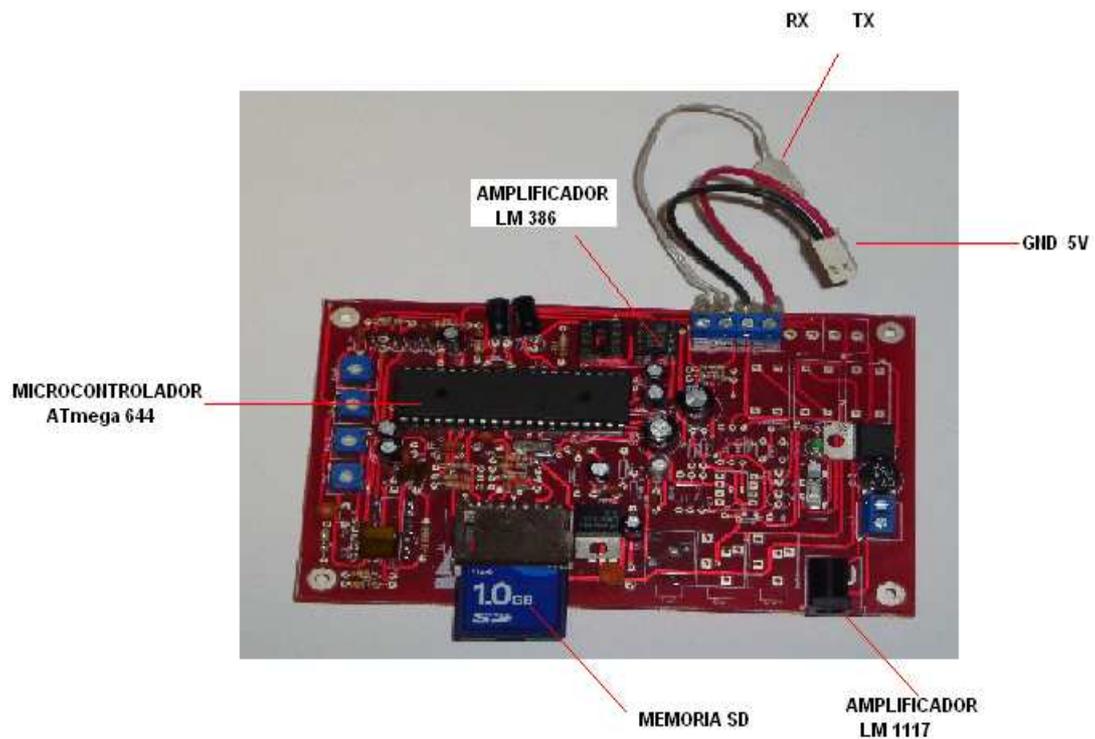


Figura 2. 40 Parte frontal del modulo de voz

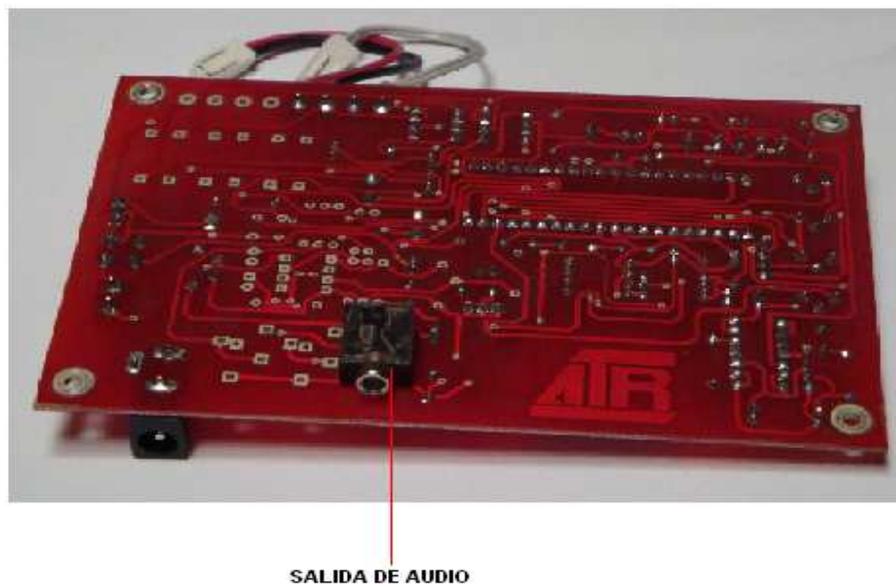


Figura 2. 41 Parte posterior del modulo de voz.

2.12 SOFTWARE DEL BASTÓN

2.12.1 DESARROLLO DEL SOFTWARE

Para desarrollar cualquier proyecto con microcontroladores se debe seguir los siguientes pasos, como se muestra en la figura 2.42

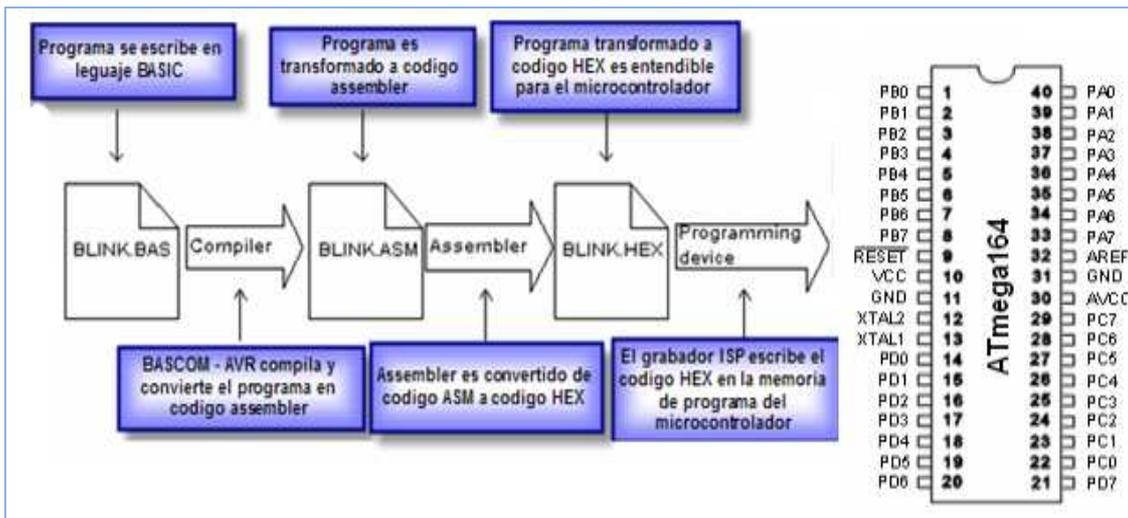


Figura 2. 42 Diagrama de bloques de pasos para programar un microcontrolador

2.12.2 PASOS PARA CREAR UN PROGRAMA

- Escribir el programa en BASIC, crea un archivo BAS.
- Compilar el programa y ver si no contiene errores
- Si no tiene errores se crea un archivo ASM que es un archivo en ensamblador
- El archivo de ensamblador es entendible para la maquina en donde se está programando pero para el microcontrolador por lo cual crea un archivo en Hexadecimal HEX.
- El archivo HEX es entendible para el microcontrolador, este archivo es el que se graba en la memoria de programa por medio de un grabador o programador ISP, como se indica en la figura 2.17

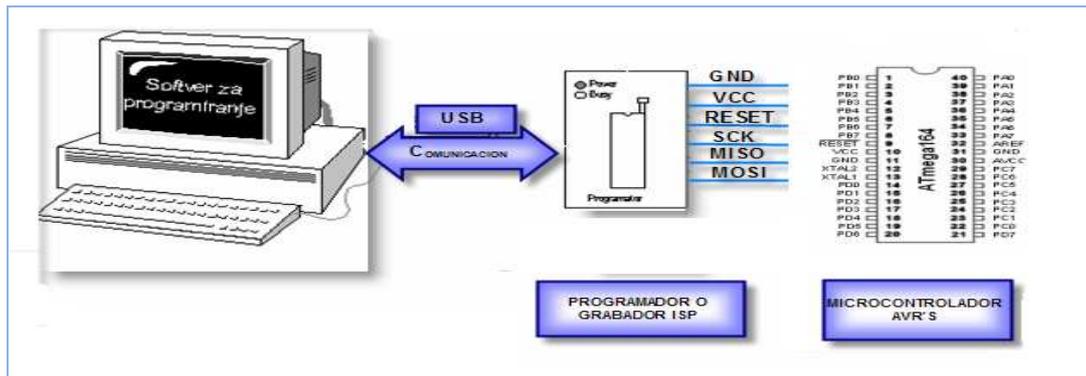


Figura 2. 43 Pasos para grabar el programa realizado en el microcontrolador

El software que se va a utilizar para poder programar el microcontrolador ATMEGA 128 y ATMEGA 64 es el BASCOM AVR versión 1.11.9.1

2.13 LENGUAJE DE PROGRAMACIÓN

2.13.1 BASCOM AVR SOFTWARE⁴⁰

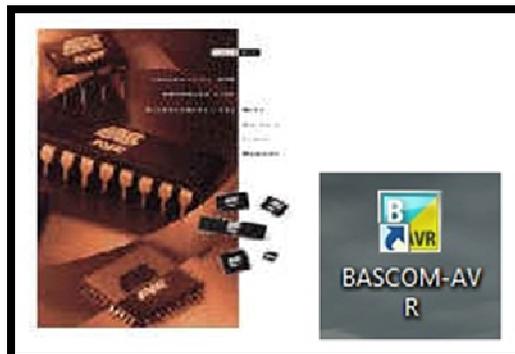


Figura 2. 44 Imagen del icono del programa BASCOM-AVR

Se realizó el programa para el microcontrolador en BASCOM – AVR, que es un Software desarrollado por la empresa MCS para programar los microcontroladores AVR de la empresa ATMEL. El programa BASCOM AVR tiene el objetivo de realizar los proyectos de microcontroladores en el menor tiempo posible utilizando la herramienta Basic por su fácil entendimiento y aplicación.

Está diseñado para funcionar en XP y W95/W98/NT/W2000

⁴⁰ <http://www.dmd.es/bascom-a.htm>

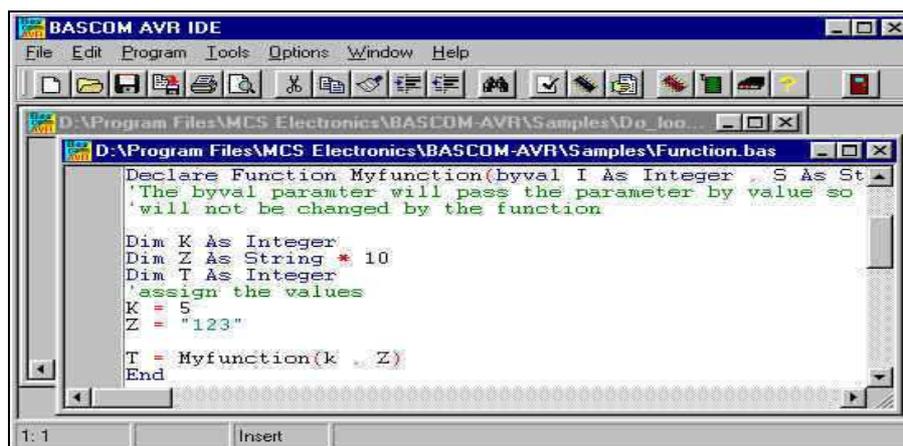


Figura 2. 45 BASCOM

- Rápido código de máquina en lugar de código interpretado.
- Variables y las etiquetas pueden ser tan largo como 32 caracteres.
- Bit, Byte, Integer, Word, Long, único y de cadenas de variables.
- Amplio conjunto de trigonométricas funciones de punto flotante. Fecha y hora de cálculo funciones.
- Compilado programas de trabajo con todos los microprocesadores AVR que tienen memoria interna.
- Las declaraciones son altamente compatibles con Microsoft la VB / QB.
- Comandos especiales para pantallas LCD, chips I2C y 1WIRE chips, PC keyboad, matriz-keyboad, RC5 recepción, el software UART, SPI, con pantalla LCD gráfica, enviar IR RC5, RC6 o código de Sony. TCP / IP W3100A con chip.
- Variables locales, las funciones de usuario, apoyo de biblioteca.
- Integrado emulador de terminal con opción de descarga.
- Integrado simulador para la prueba.
- Apoyó las declaraciones

2.13.2 FUNDAMENTOS DEL LENGUAJE ⁴¹

Los caracteres y el set de instrucciones de BASCOM son fusionados para formar etiquetas, palabras clave, variables y operadores. Estos son combinados para formar las declaraciones que constituyen un programa.

El set de instrucciones de BASCOM consiste en caracteres alfabéticos, numéricos y especiales.

Los caracteres alfabéticos en BASCOM son las letras mayúsculas (A-Z) y letras minúsculas (a-z) del alfabeto. Los caracteres numéricos son los dígitos 0-9.

Las letras A-H pueden ser usadas como parte de números hexadecimales.

Los caracteres de la Tabla 2.4, tienen significado especial en declaraciones y expresiones BASCOM:

Carácter	Nombre
ENTER	Termina la entrada de una línea
	Blanco (o espacio)
'	Apostrofe
*	Asterisco (símbolo de multiplicación)
+	Signo más
,	Coma
-	Signo menos
.	Periodo (punto decimal)
/	Slash (symbol de division)
:	Dos puntos
"	Comillas
;	Punto y coma
<	Menor que
=	Signo igual
>	Mayor que
\	Backslash
^	Exponente

Tabla 2. 4 Caracteres especiales

2.13.2.1 Línea de programa de BASCOM

Las líneas de programa de BASCOM tienen la siguiente sintaxis:

[[Línea - identificador]] [[Declaración]] [[: Declaración]],... [[Comentarios]]

2.13.2.2 Usando identificadores de línea

BASCOM soporta etiquetas de línea alfanuméricas, que podrían ser la combinación de 1 a 32 letras y dígitos, iniciando con una letra y terminando con dos puntos.

Los siguientes ejemplos son etiquetas de línea alfanuméricas correctas:

Alfa:

Pantalla:

Prueba3:

Las siguientes etiquetas de línea son equivalentes:

alfa:

Alfa:

ALFA:

Las etiquetas de línea podrían comenzar en cualquier columna. Los espacios en blanco no son admitidos entre una etiqueta alfabética y los dos puntos.

Una línea puede tener solamente una etiqueta. Cuando hay una etiqueta en la línea, ningún otro identificador puede ser usado en la misma línea. Así la etiqueta es el único identificador de una línea.

2.13.3 DECLARACIONES DE BASCOM

Una declaración de BASCOM puede ser "Ejecutable" o "No ejecutable".

Una sentencia ejecutable promueve el curso de la lógica del programa señalando la acción que se debe realizar.

La expresión no ejecutable lleva a cabo las tareas de asignar el almacenamiento de variables, la declaración y definición del tipo de variables.

Las siguientes declaraciones de BASCOM son ejemplos de declaraciones no ejecutables:

REM (empieza un comentario)

DIM

Un "Comentario" es una declaración no ejecutable, se usa para aclarar una operación del programa y su propósito. Un comentario es presentado por la sentencia de REM o un apóstrofe (').

Las siguientes líneas son equivalentes:

```
PRINT "Bienvenido":      REM etiqueta que informa imprimir.
PRINT "Bienvenido":      ' etiqueta que informa imprimir.
```

Más de una declaración de BASCOM puede ser puesta en una línea, pero los dos puntos (:) separan las declaraciones como se ilustra a continuación.

```
FOR numero = 1 TO 5:  PRINT "numero":  NEXT numero
```

2.13.4 TIPOS DE DATOS

Cada variable en BASCOM tiene un tipo de datos que determina lo que puede ser guardado en la misma.

Bit (1/8 byte). Un bit puede tener un solo valor, 0 ó 1.

Byte (1 byte). Los bytes son guardados como números binarios de 8 bits sin signo, su valor se extiende de 0 a 255.

Integer (dos bytes). Los integers son guardados como números binarios de dieciséis bits con signo, su valor se extiende desde -32,768 a +32,767.

Word (dos bytes). Los words son almacenados como números binarios de dieciséis bits sin signo, se extienden en el valor de 0 a 65535.

Long (cuatro bytes). Los longs son guardados como números binarios de 32 bits con signo que se extienden en el valor de -2147483648 a 2147483647.

Single. Los singles son guardados como números binarios de 32 bits. Se extienden en el valor de 1.5×10^{-45} a 3.4×10^{38} .

Double. Los doubles son guardados como números binarios de 64 bits con signo. Se extienden en el valor de 5.0×10^{-324} a 1.7×10^{308} .

String (hasta 254 byte). Los strings son almacenados como bytes y terminan con un byte 0. Un string con una longitud de 10 bytes ocupará 11 bytes.

Las variables pueden ser guardadas internamente (default), externamente o en EEPROM.

2.13.5 VARIABLES

Una variable es un nombre que se refiere a un objeto ó a un número específico y debe ser asignada con un solo valor numérico (either, integer, byte, long, single ó bit).

La siguiente lista indica algunos ejemplos de las asignaciones a variables:

Un valor constante:

Uno = 5

C = 1.1

El valor de otra variable numérica:

abc = def

k = g

El valor obtenido combinando otras variables, constantes y operadores:

Temp = a + 5

Temp = c + 5

El valor obtenido al llamando de una función:

Temp = Asc (S)

2.13.5.1 Nombres de variables ⁴²

El nombre de una variable BASCOM puede contener hasta 32 caracteres, estos pueden ser letras y números, pero el primer carácter siempre debe ser una letra.

Un nombre de una variable no puede ser una palabra reservada, pero las palabras reservadas compuestas son admitidas.

⁴² Ayuda del programa BASCOM AVR

Las palabras reservadas incluyen todos los comandos de BASCOM, declaraciones, nombres de funciones, registros internos y nombres de operadores

Por ejemplo, la siguiente declaración no es permitida porque AND es una palabra reservada.

```
AND = 8
```

Sin embargo, la siguiente declaración es permitida:

```
ToAND = 8
```

Se puede especificar un número hexadecimal o binario con el prefijo &H o &B.

Ejemplo:

```
a = &HA,    a = &B1010 y    a= 10
```

Antes de asignar una variable, se debe indicar al compilador sobre la misma con la declaración DIM.

Ejemplo: DIM numero As Bit,

```
DIM a as Integer
```

```
DIM k as Byte
```

```
DIM s as String * 10
```

El tipo string necesita un parámetro adicional que especifique la longitud.

También se puede usar DEFINT, DEFBIT, DEFBYTE, DEFWORD, DEFLNG o DEFSNG.

Por ejemplo, c de DEFINT dice al compilador que todas las variables que no son dimensionadas y que inician con el carácter c son del tipo integer.

2.13.6 EXPRESIONES Y OPERADORES

Este punto detalla cómo combinar, modificar, comparar, o conseguir información sobre las expresiones usando los operadores disponibles en BASCOM.

Una expresión puede ser una constante numérica, una variable, o un valor obtenido combinando constantes, variables, y otras expresiones con operadores.

Los operadores llevan a cabo operaciones matemáticas o lógicas sobre valores. Los operadores provistos por BASCOM pueden ser divididos en cuatro categorías, de la siguiente manera:

2.13.6.1 Operadores aritméticos

Usados para efectuar cálculos y son +, -, *, \, / y ^.

Entero

La división de enteros es indicada por la barra invertida (\).

Ejemplo: $Z = X \setminus Y$

Modulo Aritmético

El modulo aritmético es denotado por el modulo operador MOD.

El modulo aritmético provee el resto, en vez del cociente, de una división de enteros.

Ejemplo: $X = 10 \setminus 4$: resto = $10 \text{ MOD } 4$

Exceso y división por cero

La división por cero, produce error.

Por el momento ningún mensaje es producido así que debe asegurarse que esto no ocurra.

2.13.6.2 Operadores de relación

Usados para comparar valores numéricos o de secuencia como se muestra en la Tabla 2.1. El resultado puede ser usado para tomar una decisión respecto al flujo del programa.

Operador	Prueba de relación	Expresión
=	Igualdad	$X = Y$
<>	Desigualdad	$X <> Y$

<	Menor que	$X < Y$
>	Mayor que	$X > Y$
<=	Menor que o igual a	$X <= Y$
>=	Mayor que o igual a	$X >= Y$

Tabla 2. 5 Operadores de relación

2.13.6.3 Operadores lógicos

Los operadores lógicos llevan a cabo pruebas sobre las relaciones, manipulaciones de bits u operadores Booleanos.

Hay cuatro operadores en BASCOM, mostrados en la Tabla 4.3

Operador	Significado
NOT	Complemento Lógico
AND	Conjunción
OR	Disyunción
XOR	OR Exclusivo

Tabla 2. 6 Operadores lógicos

2.13.7 OPERADORES FUNCIONALES

Usados para complementar operadores simples.

Funciones Basic.⁴³

DO LOOP

Do 'inicio del lazo

Instrucciones

Loop 'regreso al inicio del salto

⁴³ GALARZA Juan Apuntes de curso AVR'S

DO LOOP-UNTIL

Do

A=a+1

Loop until a=2 'Cuando a=2 entonces sale del lazo

FOR NEXT

For a=inicio to fin step pasos

instrucciones

Next a

SELECT CASE

Select case variable

Case 1: instrucción

Case 2: instrucción

Case n: instrucción

End select

WHILE WEND

While condición

Instrucciones

Wend

Temporizaciones

WAIT tiempo en segundos

WAITMS tiempo en milisegundos

WAITUS tiempo en microsegundos

SET poner un 1 al pin

RESET poner un 0 al pin

Configuración del cristal: \$crystal=1000000 (en hertz)

Configuración del puerto serial: \$baud=2400

2.13.8 CONFIGURACIÓN DE LOS PINES

Los pines del microcontrolador avr tienen 3 registros los cuales son:

Registro ddr, port, pin. El registro ddr nos configura al pin como entrada salida, port es el registro de salida y pin es el registro de entra.

Ddrb.0=0 entrada alta impedancia

Portb.0=0

Ddrb.0=0 entrada pull up

Portb.0=1

Ddrb.0=1 salida a cero 20 mA

Portb.0=0

Ddrb.0=1 salida a uno 20 mA

Portb.0=1

2.14 DIAGRAMAS DE FLUJO DEL PROGRAMA

2.11.1 SISTEMA DE CONTROL

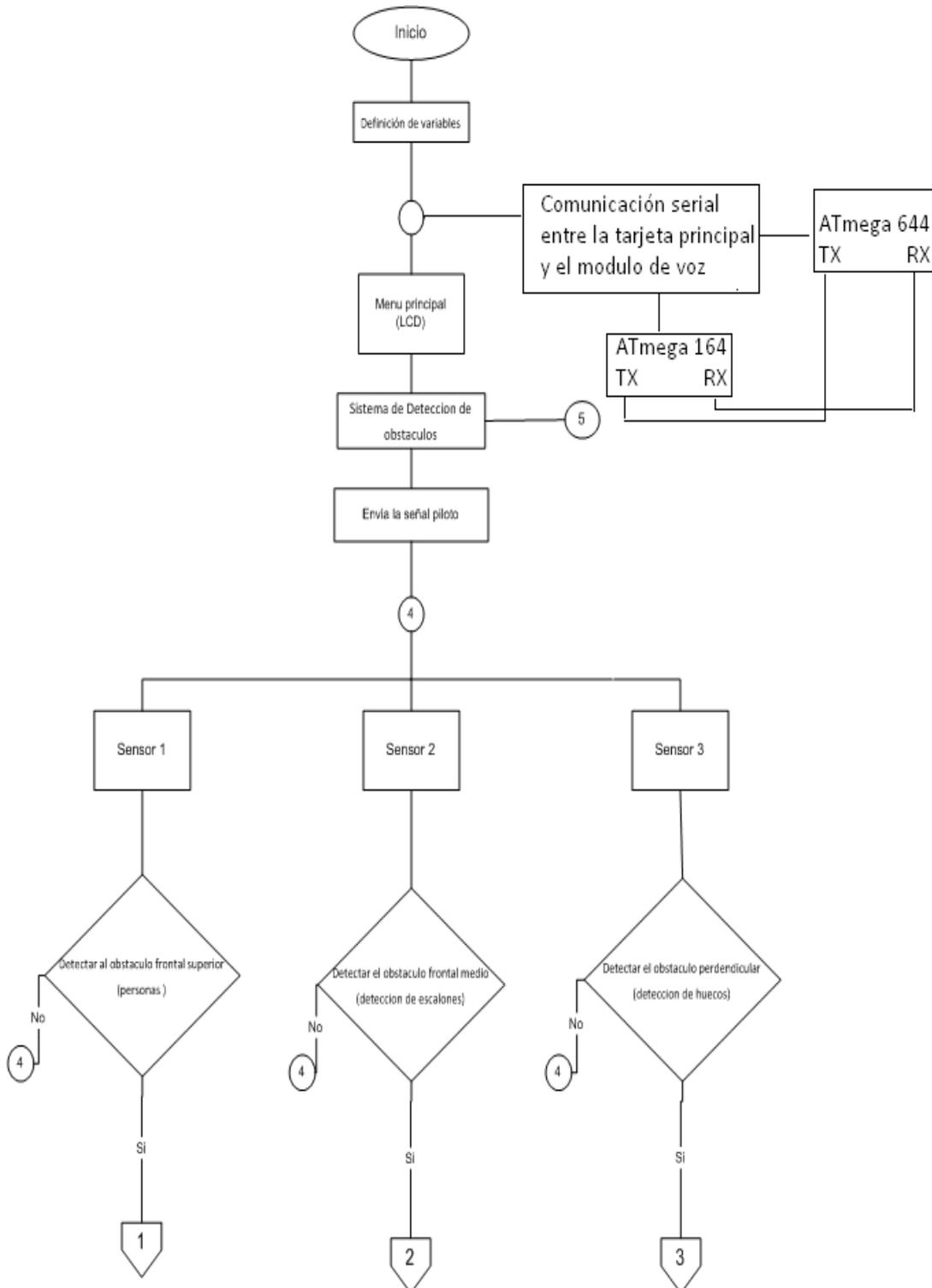


Figura 2. 46 Diagrama de Flujo del sistema de control

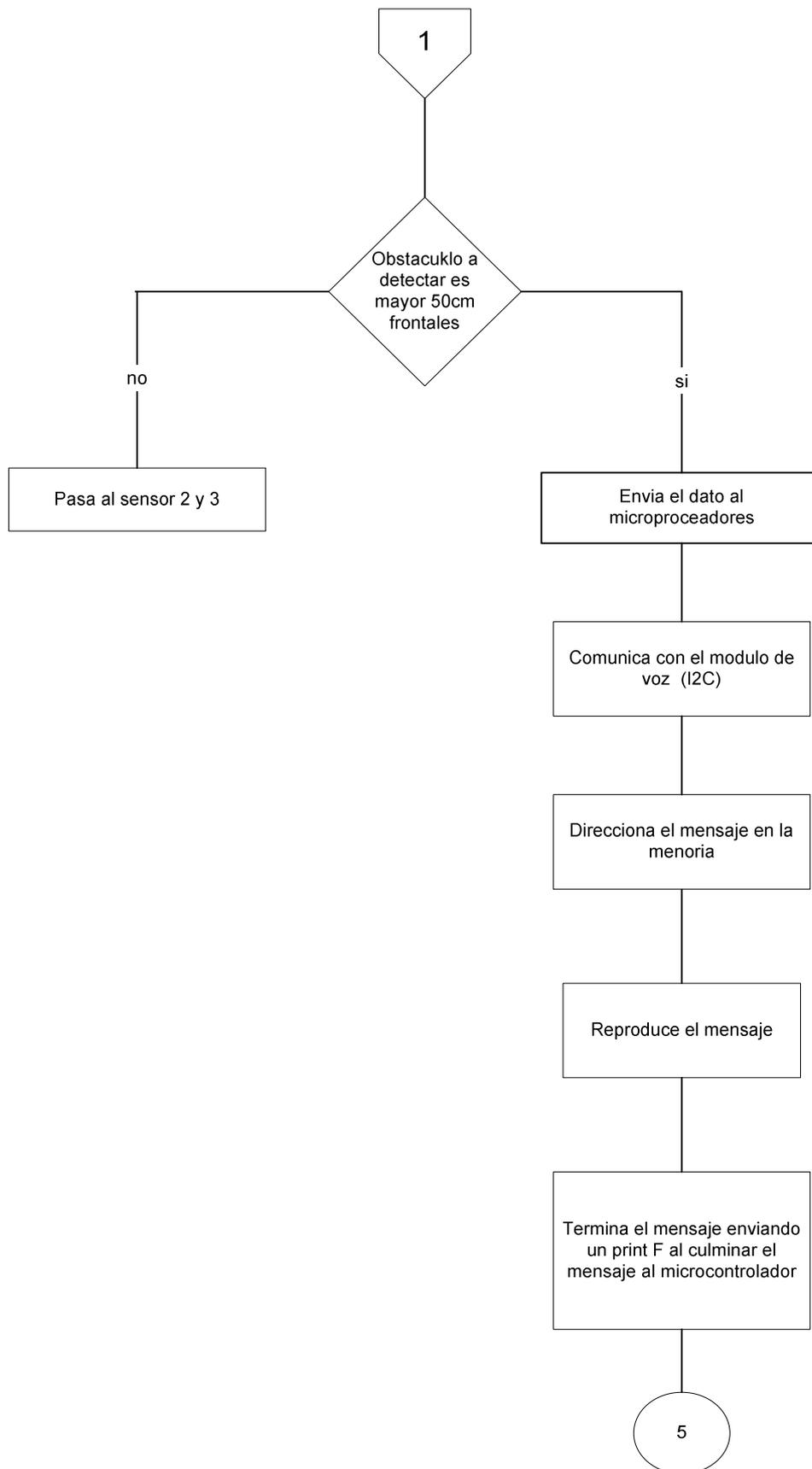


Figura 2. 47 Diagrama de Flujo del sistema de control.

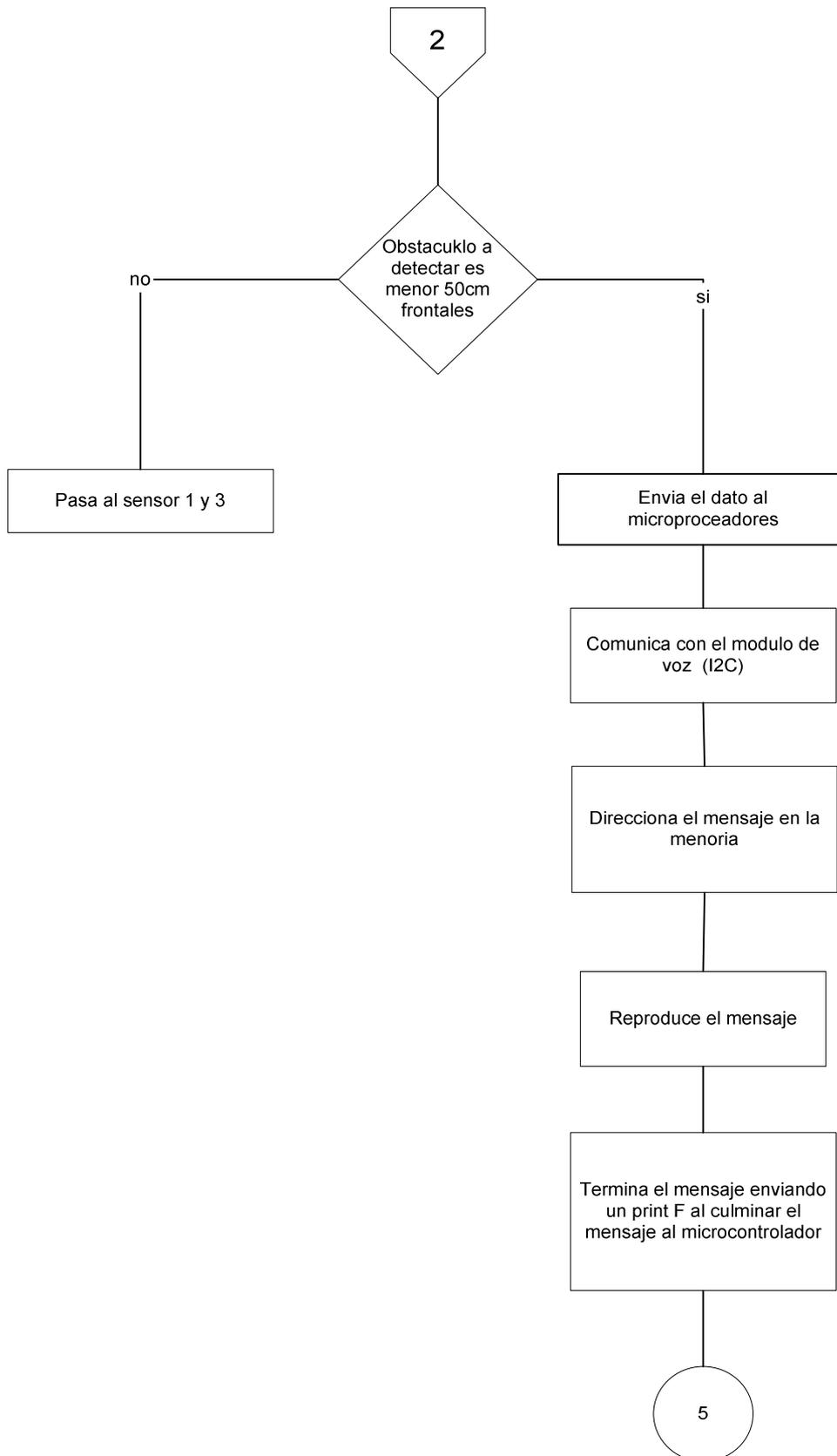


Figura 2. 48 Diagrama de Flujo del sistema de control.

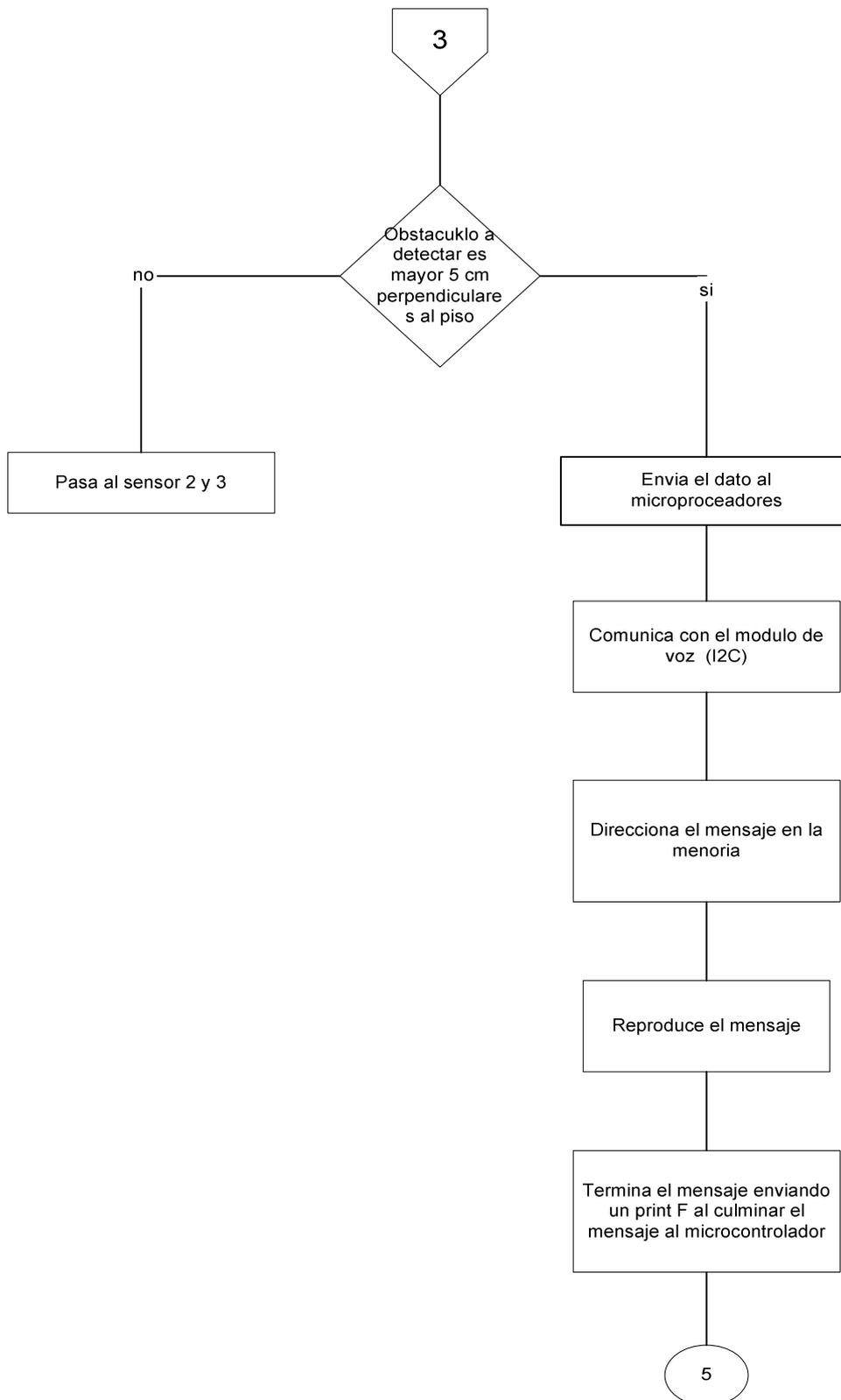


Figura 2. 49 Diagrama de Flujo del sistema de control.

4.11.2 MODULO DE VOZ

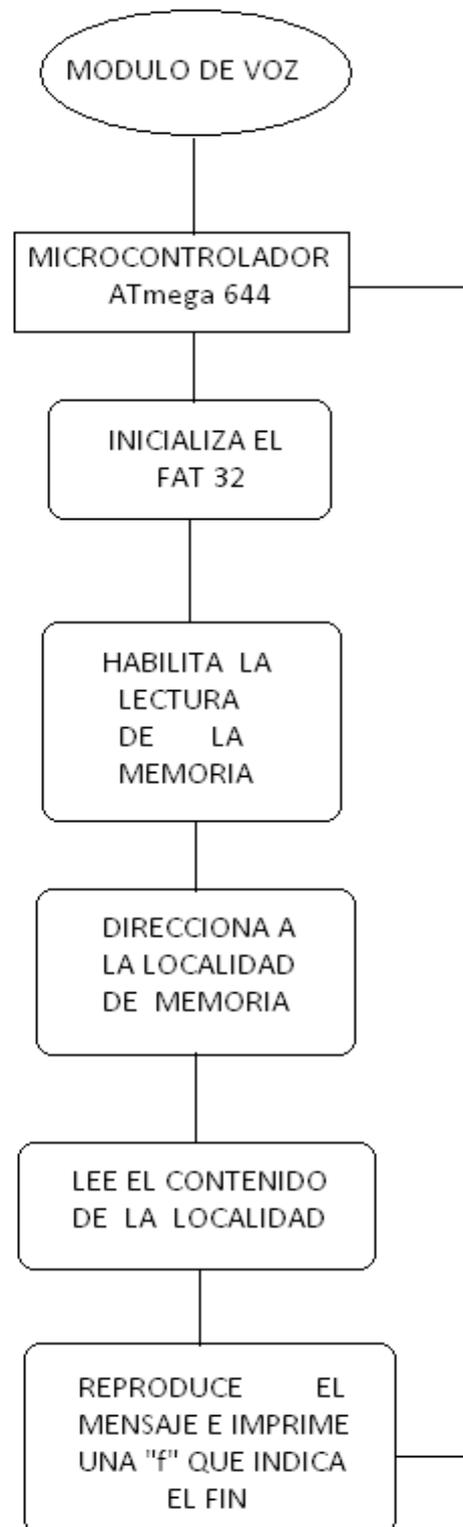


Figura 2. 50 Diagrama de Flujo del modulo de voz.

2.12 PROGRAMA DE LA TARJETA PRINCIPAL (ATMEGA 164)

```

'-----
'name          : SRF08G.bas
'copyright     : (c) 2009, MCS Electronics
'purpose       : Demuestra la utilización de los sensores SRF08
'micro         : Mega164p
'-----

'$sim
$regfile = "m164pDEF.DAT"
$crystal = 16000000
$baud = 9600          'Baud rate com.serial hardware

$hwstack = 32        ' default usa 32 para estack por
hardware
$swstack = 10        ' default usa 10 para stack por
software
$framesize = 40      ' default usa 40 para marco de espacio

Const Retardo = 250

$initmicro

!*****

'Declaracion de las subrutinas
Declare Sub Srf08_ping(byval I2caddr As Byte)
Declare Sub Srf08_results(byval I2caddr As Byte)
Declare Sub Srf08_set_range(byval I2caddr As Byte , Byval Range As Byte)
Declare Sub Srf08_set_gain(byval I2caddr As Byte , Byval Gain As Byte)
Declare Sub Srf08_set_dir(byval I2caddr As Byte , Byval Dirnew As Byte)
Declare Sub Srf08_luz(byval I2caddr As Byte)

!***** INTERUPCION de serial *****

Dim D1 As String * 3

```

Dim Flag1 As Bit	'bandera de control
Dim Flag2 As Bit	'bandera de control
Dim Control As Byte	
On Urxrc Rec_isr	'Define subrutina de Interrupcion
Enable Urxrc	'Enable Interrupts
Config Scl = Portd.6	'Configura señal de Reloj para I2C
Config Sda = Portd.7	'Configura señal de Datos para I2C
Ddra.7 = 1	
Porta.7 = 1	
Bk Alias Porta.7	
Config Lcdpin = Pin , Db4 = Porta.3 , Db5 = Porta.4 , Db6 = Porta.5 , Db7 = Porta.6 , E = Porta.2 , Rs = Porta.0	
Config Lcd = 16 * 2	
Dim Srf08_address1 As Byte	
Dim Srf08_adjust1 As Word	
Dim Srf08_address2 As Byte	
Dim Srf08_adjust2 As Word	
Dim Srf08_address3 As Byte	
Dim Srf08_adjust3 As Word	
Dim Srf08_address4 As Byte	
Dim Srf08_adjust4 As Word	
Dim Srf08_address5 As Byte	
Dim Srf08_adjust5 As Word	
Dim Srf08_address6 As Byte	
Dim Srf08_adjust6 As Word	
Dim Srf08_range1 As Word	
Dim Srf08_range2 As Word	

Dim Srf08_range3 As Word

Dim I2crange As Word

Dim I2cluz As Byte

Dim Texto1 As String * 4 , Texto2 As String * 4 , Texto3 As String * 4

Dim X As Byte , Y As Byte

!*****

'Call Srf08_set_dir(0 , &HE6)

'Do

'Loop

!*****

Srf08_address1 = &HE0

Srf08_address2 = &HE4

Srf08_address3 = &HE6

Call Srf08_set_range(srf08_address1 , 200) 'Setea el rango del SRF08(1)

Call Srf08_set_gain(srf08_address1 , 1) 'Setea la ganancia del SRF08(1)

Call Srf08_set_range(srf08_address2 , 'Setea el rango del SRF08(2)

Call Srf08_set_gain(srf08_address2 , 1) 'Setea la ganancia del SRF08(2)

Call Srf08_set_range(srf08_address3 , 200) 'Setea el rango del SRF08(3)

Call Srf08_set_gain(srf08_address3 , 1) 'Setea la ganancia del SRF08(3)

Srf08_adjust1 = 0 'Variable para ajustar el dato de salida(1)

Srf08_adjust2 = 0 'Variable para ajustar el dato de salida(2)

Srf08_adjust3 = 0 'Variable para ajustar el dato de salida(3)

Flag1 = 0

Flag2 = 0

Bk = 1

Wait 1

```
For X = 1 To 2
```

```
  Print "S1050.WAV"
```

```
  Wait 5
```

```
Next X
```

```
'Enable Interrupts
```

```
Cls
```

```
Lcd " BASTON "
```

```
Locate 2 , 1
```

```
Lcd " PARA CIEGOS "
```

```
'Texto2 = ""
```

```
'Texto2 = "S1" + "050" + ".WAV"
```

```
'Print "S1050.WAV"
```

```
Wait 3
```

```
Cls
```

```
Do
```

```
  Call Srf08_ping(0)
```

```
  Waitms 70
```

```
  Call Srf08_results(srf08_address1)
```

```
  Srf08_range1 = I2crange
```

```
  Call Srf08_results(srf08_address2)
```

```
  Srf08_range2 = I2crange
```

```
  Call Srf08_results(srf08_address3)
```

```
  Srf08_range3 = I2crange
```

```
  Locate 1 , 1
```

```
  Lcd "1=" ; Srf08_range1 ; "cm " ; " "
```

```
  Locate 1 , 9
```

```
  Lcd "2=" ; Srf08_range2 ; "cm " ; " "
```

```
  Locate 2 , 1
```

```
  Lcd "3=" ; Srf08_range3 ; "cm " ; " "
```

```
' If Flag2 = 0 Then
```

```

If Srf08_range3 > 7 Then
  Flag1 = 1
  Flag2 = 1
  Texto1 = Str(srf08_range3)
  Texto2 = "S3" + Format(texto1 , "000" ) + ".WAV"
  Texto2 = "S3" + "020" + ".WAV"
  Print Texto2
End If
' End If

' If Flag1 = 0 Then
'   Flag1 = 1
  If Srf08_range1 >= Srf08_range2 Then
    Texto1 = Str(srf08_range1)
    Texto2 = "S1" + Format(texto1 , "000" ) + ".WAV"
    Texto2 = "S1" + "050" + ".WAV"
    Print Texto2
  Else
    Texto1 = Str(srf08_range2)
    Texto2 = "S2" + Format(texto1 , "000" ) + ".WAV"
    Texto2 = "S2" + "080" + ".WAV"
    Print Texto2
  End If
' End If
  Waitms 200

Loop

End                               'Fin del Programa

'----- Subrutinas -----
Sub Srf08_ping(byval I2caddr As Byte)      'Envia comando para leer en cm
  I2cstart

```

```

I2cwrite I2caddr
I2cwrite 0
I2cwrite 81
I2cstop
End Sub

```

```

'-----
Sub Srf08_set_range(byval I2caddr As Byte , Byval Range As Byte)

```

```

'Cambia rango de medida del SRF08

```

```

I2cstart
I2cwrite I2caddr
I2cwrite 2
I2cwrite Range
I2cstop
End Sub

```

```

'-----
Sub Srf08_set_gain(byval I2caddr As Byte , Byval Gain As Byte)

```

```

'Cambia la ganancia del SRF08

```

```

I2cstart
I2cwrite I2caddr
I2cwrite 1
I2cwrite Gain
I2cstop
End Sub

```

```

'-----
Sub Srf08_results(byval I2caddr As Byte)      'Lee el resultado de la medición

```

```

Dim Lsb As Byte
Dim Msb As Byte
I2cstart
I2cwrite I2caddr
I2cwrite 2
I2cstart

```

```

Incr I2caddr
I2cwbyte I2caddr
I2crbyte Msb , Ack
I2crbyte Lsb , Nack
I2cstop
I2crange = Makeint(Lsb , Msb)

```

```
End Sub
```

```

'-----
Sub Srf08_set_dir(byval I2caddr As Byte , Byval Dirnew As Byte) 'Cambia la
direccion I2C del SRF08

```

```

I2cstart
I2cwbyte I2caddr
I2cwbyte 0
I2cwbyte &HA0
I2cstop
I2cstart
I2cwbyte I2caddr
I2cwbyte 0
I2cwbyte &HAA
I2cstop
I2cstart
I2cwbyte I2caddr
I2cwbyte 0
I2cwbyte &HA5
I2cstop
I2cstart
I2cwbyte I2caddr
I2cwbyte 0
I2cwbyte Dirnew
I2cstop

```

```
End Sub
```

```
'-----
```

```

Sub Srf08_luz(byval I2caddr As Byte)           'Lee el valor del sensor de Luz
    I2cstart
    I2cwbyte I2caddr
    I2cwbyte 1
    I2cstart
    Incr I2caddr
    I2cwbyte I2caddr
    I2crbyte I2cluz , Nack
    I2cstop
End Sub

```

```

'-----
_init_micro:
Ddra.1 = 1
Porta.1 = 0
Return

```

```

!*****

```

```

Rec_isr:
    Input D1 Noecho
    If D1 = "F" Then
        Flag1 = 0
        Flag2 = 0
    End If
Return

```

```

!*****

```

2.13 SOFTWARE MODULO DE VOZ

```

'$regfile = "M644def.dat"
$crystal = 8000000           '24000000
$baud = 9600

```

```
Dim Sample As Byte
Dim Sample1 As Byte
Dim Busy As Byte
Dim S As String * 10
Dim J As Word
```

```
Ddrb.1 = 1
Portb.1 = 0
Ddrb.3 = 1
Portb.4 = 0
```

```
'Config Porta = Output
'Config Portc = Output
Wait 1
```

```
'Do
'Print "hola"
'Waitms 300
'Toggle Porta
'Loop
```

```
Print "Config_MMC.bas"
```

```
Config Timer1 = Pwm , Prescale = 1 , Compare A Pwm = Clear Up , Pwm = 8
'Config Clock = Soft 'Enable Interrupts
'Config Date = Mdy , Separator = /
Dim Btemp1 As Byte
Print "Wait for Drive"
```

```
'Cls
'Cursor Off
```

```
' Include here you driver for Compactflash/HardDisk or other
```

```

$include "Config_MMC.bas"                ' Does drive init too
'$Include "Config_HardDisk_M128.bas"

If Gbdriveerror = 0 Then

' Include AVR-DOS Configuration and library
$include "Config_AVR-DOS.BAS"

Waitms 100

Print "Init File System ... ";
Btemp1 = Initfilesystem(1)                ' Partition 1
                                           ' use 0 for drive without Master boot record

If Btemp1 <> 0 Then
    Print "Error: " ; Btemp1 ; " at Init file system"
Else
    Print "Filesystem type: " ; Gbfilesystem
    Print "FAT Start Sector: " ; Glfatfirstsector
    Print "Root Start Sector: " ; Glrootfirstsector
    Print "Data First Sector: " ; Gldatafirstsector
    Print "Max. Cluster Nummber: " ; Glmaxclusternumber
    Print "Sectors per Cluster: " ; Gbsectorspercluster
    Print "Root Entries: " ; Gwrootentries
    Print "Sectors per FAT: " ; Glsectorsperfat
    Print "Number of FATs: " ; Gbnumberoffats
    Print "Disk size: " ; Disksize() ; " kB"
    Print "Disk free: " ; Diskfree() ; " kB"
    Print " OK"

'Set Portd.7
*****

' Print "Dir *.wav"                        'dir mp3
' Print

```

```

' -----
' 'crea una lista de mp3 presenti sulla MMC
' -----

Open "musica.txt" For Output As #9          'create file.list in MMC
  S = Dir( "*.wav")
  While Len(s) > 0
    'Print S ; " " ; Filedate() ; " " ; Filetime() ; " " ; Filelen()
    Print #9 , S                          ; "hola"          'memorisa la lista
  S = Dir()
  Wend
Close #9

' Open "Lista.txt" For Input As #9          'OPEN LA LISTA
'   While Eof(#9) = 0
'     Line Input #9 , S                    'READ filename Mp3
'     Print S
'   Wend
' Close #9
*****

' 'Open "4.wav" For Binary As #3           ' Open speech file.
' ' Print "4.wav" ; " (" ; Lof(#3) ; " bytes)"      ' Log filename and duration.
' ' While Eof(#3) <> 255                       ' Any more samples?
' '   Get #3 , Sample                          ' Yes, get a sample from file
' '   Pwm1a = Sample                          ' and bang it to the PWM.
' '   Pwm1b = Sample
' '   For Busy = 1 To 5                       ' Wait a while (just a rude
' '     NOP                                    ' busy-wait loop).
' '   Next
' ' Wend                                       ' And look for other samples.
' Close #3
*****

' Reset Portd.7
'

```

```

End If
Else
  Print "Error during Drive Init: " ; Gbdriveerror
End If

Print "ingrese canción"
Pwm1a = 0

Do
'Home U
'Lcd "ingrese canción"
'Home L
'Lcd "8.raw"
J = 0
Open "7.wav" For Binary As #3          ' Open speech file.
  Print "7.wav" ; " (" ; Lof(#3) ; " bytes)"  ' Log filename and duration.
  While Eof(#3) <> 255                    ' Any more samples?
    Get #3 , Sample
    Incr J
    If J > 1000 Then
      Pwm1a = Sample
      'Pwm1b = Sample
      J = 1100
    End If
    For Busy = 1 To 5                    'Waitus 100
      nop                                ' Wait a while (just a rude
    Next

  Wend

  Pwm1a = 0
  Close #3
*****

Wait 3

```

```

'Cls
'Home U
'Lcd "8.wav"
J = 0
Open "8.wav" For Binary As #3           ' Open speech file.
    Print "8.wav" ; " (" ; Lof(#3) ; " bytes)"   ' Log filename and duration.
    J = 0
    Do
    Get #3 , Sample
    Print Hex(sample) ; Chr(sample)
    Incr J
    Loop Until J = 4

    While Eof(#3) <> 255                   ' Any more samples?

        Get #3 , Sample
        Pwm1a = Sample
        'Get #3 , Sample1
        'Portc = Sample1
        For Busy = 1 To 4
        ' Ddrd.5 = 0
        nop                               ' Wait a while (just a rude
        Next
        'Ddrd.5 = 1
    Wend
    Pwm1a = 0
    Close #3

```

```

'Cls
'Home U
'Lcd "8.raw"
J = 0
Open "8.WAV" For Binary As #3           ' Open speech file.

```

```

Print "8.WAV" ; " (" ; Lof(#3) ; " bytes)"      ' Log filename and duration.
While Eof(#3) <> 255                            ' Any more samples?
  Get #3 , Sample
  Incr J
  If J > 1000 Then
    Pwm1a = Sample
    Pwm1b = Sample
    J = 1100
  End If

  For Busy = 1 To 4
    nop                                          ' Wait a while (just a rude
  Next                                          '

Wend
Pwm1a = 0
Close #3

*****

' Open "9.wav" For Binary As #3                ' Open speech file.
' Print "9.wav" ; " (" ; Lof(#3) ; " bytes)"  ' Log filename and duration.
' While Eof(#3) <> 255                        ' Any more samples?
'   Get #3 , Sample
'   Incr J
'   If J > 1000 Then
'     Pwm1a = Sample
'     J = 1100
'   End If
'   For Busy = 1 To 10
'     nop                                      ' Wait a while (just a rude
'   Next                                      '

' Wend
' Pwm1a = 127
' Close #3

*****

```

```

'Cls
'Home U
'LCD "8.wav"
J = 0
Open "9.wav" For Binary As #3           ' Open speech file.
  Print "9.wav" ; " (" ; Lof(#3) ; " bytes)" ' Log filename and duration.
  While Eof(#3) <> 255                  ' Any more samples?
    Get #3 , Sample
    Incr J
    If J > 1000 Then
      Pwm1a = Sample
      J = 1100
    End If
    'For Busy = 1 To 1
    'nop                                ' Wait a while (just a rude
    'Next                                '
  Wend
  Pwm1a = 0
  Close #3

```

!*****

Loop

CAPITULO 3

PRUEBAS Y RESULTADOS

3.1 PRUEBAS

Las pruebas son de vital importancia en el ciclo de desarrollo tanto del hardware como del software, consistiendo en la revisión final de los requerimientos, análisis y diseño para finalmente realizar la implementación. El objetivo de las pruebas es encontrar fallas o errores para luego hacer una depuración del sistema y así asegurar que el proyecto ha sido desarrollado de acuerdo a los requerimientos y que todos los errores han sido detectados.

En este punto se describirán los diferentes tipos de pruebas que se realizaron.

3.1.1 FORMATO PARA PRUEBAS DE LOS DIFERENTES SISTEMAS

Se realizaron las pruebas a cada sistema del bastón, con el fin de lograr una depuración al detectar posibles errores.

Para llevar a cabo las pruebas del sistema se ha definido el siguiente formato:

Número:		<Número de la prueba>		
Prueba:		<Nombre de la prueba>		
No.	Acción	Resultado	Intentos	Éxito
1	<Acción 1>	<Resultado obtenido 1>		S/N
2	<Acción 2>			S/N
...	...			S/N

n	<Acción n>			S/N
Conclusión:		<Conclusión de la prueba>		
Observación:		<Observación de la prueba>		

Tabla 3. 1 Formato de pruebas de los diferentes sistemas de sensores

Como se pudo observar el capítulo tres del presente proyecto de titulación se explico todo sobre el hardware del robot, al cual en el sistema de control global se dividió en tres partes: Sistemas de sensores, sistema de control y sistemas de actuadores, para realizar las pruebas vamos a realizar de la misma manera.

3.1.2 SISTEMA DE DETECCIÓN DE OBSTÁCULOS “SRF08”

Sensor 1

Número	1			
Prueba	Detección de obstáculos del sensor 1			
No.	Acción	Resultado	Intentos	Éxito
1	Detecta un obstáculo localizado mayor a 50cm de la parte frontal del bastón electrónico al estar en movimiento	Detecta el receptor de ultrasonido la señal reflejada por medio del obstáculo, el microcontrolador, proceso la señal haciendo que envíe un mensaje de voz a la persona que está utilizando el bastón para evitar su colisión con dicho obstáculo.	1	Si
Conclusión:		El sistema de detección de obstáculos funciono correctamente		
Observación:		Para que funcione correctamente, se debe colocar a una adecuada altura ya que los lóbulos de radiación se interfieren entre ellos y nos dan los valores reales de medida.		

Sensor 2

Número:		1		
Prueba:		Detección de obstáculos de sensor 2		
No.	Acción	Resultado	Intentos	Éxito
1	Detecta un obstáculo localizado menor a 50 cm de la parte frontal del bastón electrónico al estar en movimiento	Detecta el receptor de ultrasonido la señal reflejada por medio del obstáculo, el microcontrolador proceso la señal haciendo que envié un mensaje de voz a la persona que está utilizando el bastón para evitar su colisión con dicho obstáculo.	1	Si
Conclusión:		El sistema de detección de obstáculos funciona correctamente		
Observación:		Para que funcione correctamente, se debe colocar a una adecuada altura ya que los lóbulos de radiación se interfieren entre ellos y nos dan los valores reales de medida.		

Sensor 3

Número: 1		1		
Prueba:		Detección de obstáculos del sensor 3		
No.	Acción	Resultado	Intentos	Éxito
1	Detecta un obstáculo localizado a una altura de 5 cm de la parte perpendicular del bastón electrónico	Detecta el receptor de ultrasonido la señal reflejada por medio del obstáculo, el microcontrolador, proceso la señal haciendo que envié un mensaje de voz a la persona que está utilizando el bastón para evitar su colisión con	1	Si

	al estar en movimiento	dicho obstáculo.		
Conclusión:		El sistema de detección de obstáculos funciona correctamente		
Observación:		Para que funcione correctamente, se debe colocar a una adecuada altura ya que los lóbulos de radiación se interfieren entre ellos y nos dan los valores reales de medida.		

3.1.3 SISTEMA DE CONTROL “TARJETA PRINCIPAL”

Número:		1		
Prueba:		Sistema de control “Tarjeta principal”		
No.	Acción	Resultado	Intentos	Éxito
1	Al activarse un sensor manda una señal al microcontrolador ATMEGA 164	Haciendo que con dicha señal se procese por medio de software y produzca un determinado mensaje dependiendo a qué distancia se encuentre el obstáculo también se comunica con el microprocesador ATMEGA 644 que es el modulo de voz.	1	Si
Conclusión:		El sistema de control “Tarjeta principal” funciona correctamente		
Observación:		Para funcionar correctamente el sistema de control, al cual están conectados todos los sensores y actuadores, se debe de programar bien en el software BASCOM-AVR, y también realizar bien la grabación teniendo en cuenta los fuse bits		

3.1.4 SISTEMA DE ACTUADORES “MODULO DE VOZ”

Número:		1		
Prueba:		Reproductor de sonido “Modulo de Voz”		
No.	Acción	Resultado	Intentos	Éxito
1	Comunicación entre el microcontrolador ATMEGA 164 y el microcontrolador del ATMEGA 644 por medio de protocolo I2C	El microcontrolador ATMEGA 164 se comunica con el ATMEGA 644, el cual direcciona en la memoria flash y reproduce el mensaje.	1	Si
Conclusión:		El sistema de reproducción de sonido funciono correctamente		
Observación:		Hay que tomar en cuenta que el modulo de voz debe estar bien configurado con el hardware y el software		

3.2 RESULTADOS

El montaje y las pruebas finales se realizaron de manera progresiva, es decir que se fueron armando por etapas hasta llegar al prototipo final. Como resultado se tiene que es sistema de lazo abierto este bien implementado tanto en los sensores, control y actuadores funcionan correctamente.

CAPITULO 4

CONCLUSIONES Y RECOMENDACIONES

4.1 CONCLUSIONES

- El bastón que se construyó ayudará a las personas no videntes permitiéndoles movilidad en un medio físico desconocido, evitando que ellos se acerquen a objetos sin antes saberlo y estar prevenidos.
- En el mercado existen ayudas dedicadas a personas no videntes pero tienen costos demasiado elevados, en algunos casos estos modelos solo están en investigación, todo esto dificulta la adquisición de este elemento que es de vital importancia para una persona que carece de este sentido tan importante.
- A las pruebas que fue sometido el bastón se observó que siempre las especificaciones técnicas que dan los manuales no siempre se pueden cumplir, solo son aplicables a ciertos lugares, como fue en el caso de los sensores que en diferentes superficies actúan diferente por lo que siempre es necesario realizar pruebas.
- Para poder tener una buena comunicación se utilizó los microcontroladores ATMEGA 164 y ATMEGA 644 debido a que estos presentaban buenas características como es, la cantidad de memoria flash, números de pines de entrada y salida, entre otras, lo que permitió una comunicación más ágil entre las tarjetas.
- El manejo del módulo de voz, nos da un mayor campo de innovación tecnológica. En nuestro proyecto este módulo nos ayudó en la reproducción de mensajes reales, que le dieron un toque especial al bastón.

- En el desarrollo del bastón electrónico para personas no videntes se emplearon varias etapas en la construcción como son: detección de obstáculos, control y actuadores, estas etapas son importantes para una óptima comunicación entre las diferentes tarjetas todo esto nos permite tener un buen control sobre el bastón.
- En el proceso de ejecución y control de acciones, como también del monitoreo de cada uno de los sensores se ejecuto de manera serial; es decir que nos sirve para ejecutar una sola acción a la vez (en tiempo del orden de micro-segundos), para la percepción humana porque resulta que el control, monitoreo y la presentación de los datos en el LCD se da simultáneamente.
- Por medio de este proyecto nos damos cuenta que la innovación tecnológica avanza demasiado rápido, permitiéndonos interactuar de diferentes formas con los objetos que tenemos , en este caso con el uso de los sensores con la ayuda de otros elementos nos permite detectar algún objeto y que se convierta en un aviso de peligro hacia la persona no vidente, todo esto nos hace pensar que un robot o un bastón puede tomar decisiones por si mismo y volverse importante para una persona , pero nunca nos olvidemos que los seres humanos somos únicos e independientes y no necesitamos de programas de computadora para poder realizar un movimiento que es parte de todos los sentidos con los que cuenta una persona.

4.2 RECOMENDACIONES

- Para innovar en el desarrollo de la estructura física del bastón se debe investigar otros tipos de sensores que permitan colocarse de indistinta manera en la estructura física para que no causen interferencia entre ellos.
- Para tener un mejor aviso de posibles peligros en el trayecto de uso del bastón se sugiere implementar una manilla vibratoria que provea de estímulos al usuario.
- Para realizar la recarga de la batería solo se debe de usar el cargador el cargador de batería de 12V para evitar dañarla.
- El bastón para mayor estabilidad mecánica (ruedas) debe de ser utilizado en un piso lizo o alfombrado o en lugares de cemento y tierra firme.
- A la hora de escoger un microcontrolador para emplearlo en cualquier proyecto de titulación hay que tomar en cuenta algunos factores tales como:
 - La documentación, ayudas pedagógicas
 - Herramientas de desarrollo disponible
 - Su precio
 - La cantidad de fabricantes que lo producen
 - Características propias del microcontrolador como. Tipo y cantidad de memoria de programa, numero de temporizadores, interrupciones, velocidad de procesamiento de datos, numero de entradas y salida, consumo de energía, ancho de la palabra, diseño de la placa, etc.
- Para la realización del PCB de cualquier proyecto de titulación se seleccione el software adecuado de acuerdo a la necesidad, por ejemplo

PROTEUS permite la simulación de la mayoría de los circuitos, pero en la realización del circuito impreso no tiene las suficientes librerías de los encapsulados de algunos dispositivos electrónicos.

BIBLIOGRAFIA

Libros:

1. PALLÁS ARENY, Ramón
Sensores y Acondicionamiento de Señal
Tercera edición, Edición original publicada por Marcambo, S.A., Barcelona – España, Págs. 2-4.
2. COSTALES, Alcívar
Apuntes de control con microcontroladores
3. REYES A. Carlos
Microcontroladores PIC Programación en Basic 2da Edición
4. Galarza Juan, Apuntes de microcontroladores

Direcciones de Internet:

1. Diagramas de flujo
http://es.wikipedia.org/wiki/Diagrama_de_flujo
2. Tipos de LCD
<http://es.wikipedia.org/wiki/LCD>
3. Introducción a los microcontroladores
<http://www.monografias.com/trabajos12/microco/microco.shtml#>
4. Microcontroladores
<http://www.monografias.com/trabajos12/microco/microco.shtml>
www.monografias.com/trabajos12/microco/microco.shtml
<http://es.wikipedia.org/wiki/Microcontrolador>
5. Microcontroladores AVR

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

6. Bascom AVR

<http://www.dmd.es/bascom-a.htm>

Ayuda del programa BASCOM AVR

7. ATmega 644 /ATmega 164

<http://www.atmel.com/avr>.

Datasheet

8. WIKIPEDIA, LA ENCICLOPEDIA LIBRE.HTM

[www. PWM - Wiki pedía, la enciclopedia libre.htm](http://www.pwm-wiki.com/)

[www. Modulación por ancho de pulso - Wiki pedía, la enciclopedia libre.htm](http://www.pwm-wiki.com/)

9. LINK para la descarga de PDF's de información de LCD gráficas y sus controladores:

<http://www.eio.com/datashet.htm>

10. Una vara para ciegos

<http://www.impresionante.net/10-03-2008/general/una-vara-para-ciegos>

11. Bastón blanco

http://www.geocities.com/asocciegos/baston_blanco.html

12. Visión de murciélago para ciegos

<http://dctecno.com.ar/tecnologia/11-Vision-de-murcielago-para-ciegos.html>

ANEXOS

ANEXO 1:
DESCRIPCIÓN DEL
MICROCONTROLADOR
ATMEGA 164/644

Features

- High-performance, Low-power AVR® 8-bit Microcontroller
- Advanced RISC Architecture
 - 131 Powerful Instructions - Most Single-clock Cycle Execution
 - 32 x 8 General Purpose Working Registers
 - Fully Static Operation
 - Up to 20 MIPS Throughput at 20 MHz
 - On-chip 2-cycle Multiplier
- Nonvolatile Program and Data Memories
 - 16/32/64K Bytes of In-System Self-Programmable Flash
Endurance: 10,000 Write/Erase Cycles
 - Optional Boot Code Section with Independent Lock Bits
In-System Programming by On-chip Boot Program
True Read-While-Write Operation
 - 512B/1K/2K Bytes EEPROM
Endurance: 100,000 Write/Erase Cycles
 - 1/2/4K Bytes Internal SRAM
 - Programming Lock for Software Security
- JTAG (IEEE std. 1149.1 Compliant) Interface
 - Boundary-scan Capabilities According to the JTAG Standard
 - Extensive On-chip Debug Support
 - Programming of Flash, EEPROM, Fuses, and Lock Bits through the JTAG Interface
- Peripheral Features
 - Two 8-bit Timer/Counters with Separate Prescalers and Compare Modes
 - One 16-bit Timer/Counter with Separate Prescaler, Compare Mode, and Capture Mode
 - Real Time Counter with Separate Oscillator
 - Six PWM Channels
 - 8-channel, 10-bit ADC
Differential mode with selectable gain at 1x, 10x or 200x
 - Byte-oriented Two-wire Serial Interface
 - One/Two Programmable Serial USART (ATmega644, ATmega164/324)
 - Master/Slave SPI Serial Interface
 - Programmable Watchdog Timer with Separate On-chip Oscillator
 - On-chip Analog Comparator
 - Interrupt and Wake-up on Pin Change
- Special Microcontroller Features
 - Power-on Reset and Programmable Brown-out Detection
 - Internal Calibrated RC Oscillator
 - External and Internal Interrupt Sources
 - Six Sleep Modes: Idle, ADC Noise Reduction, Power-save, Power-down, Standby and Extended Standby
- I/O and Packages
 - 32 Programmable I/O Lines
 - 40-pin PDIP, 44-lead TQFP, and 44-pad QFN/MLF
- Operating Voltages
 - 1.8 - 5.5V for ATmega164/324/644V
 - 2.7 - 5.5V for ATmega164/324/644
- Speed Grades
 - ATmega164/324/644V: 0 - 4MHz @ 1.8 - 5.5V, 0 - 10MHz @ 2.7 - 5.5V
 - ATmega164/324/644: 0 - 10MHz @ 2.7 - 5.5V, 0 - 20MHz @ 4.5 - 5.5V
- Power Consumption at 1 MHz, 3V, 25°C for ATmega644
 - Active: 240 µA @ 1.8V, 1 MHz
 - Power-down Mode: 0.1 µA @ 1.8V



8-bit **AVR**[®]
Microcontroller
with 16/32/64K
Bytes In-System
Programmable
Flash

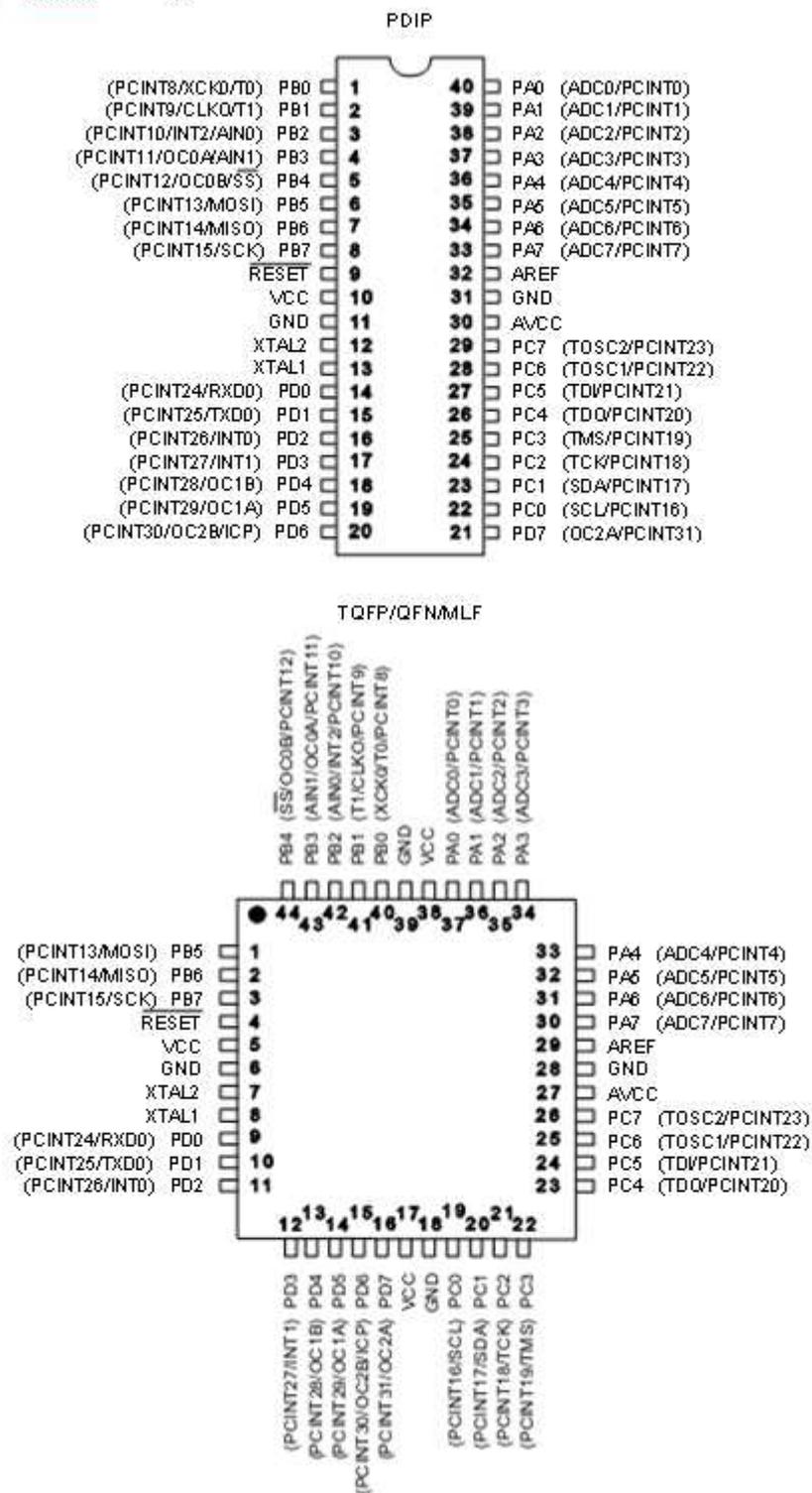
[ATmega164/V](#)

[ATmega324/V](#)

[ATmega644/V](#)

Advance
Information
Summary

Figure 2. Pinout ATmega644



Note: The large center pad underneath the QFN/MLF package should be soldered to the board to ensure good mechanical stability.

Disclaimer

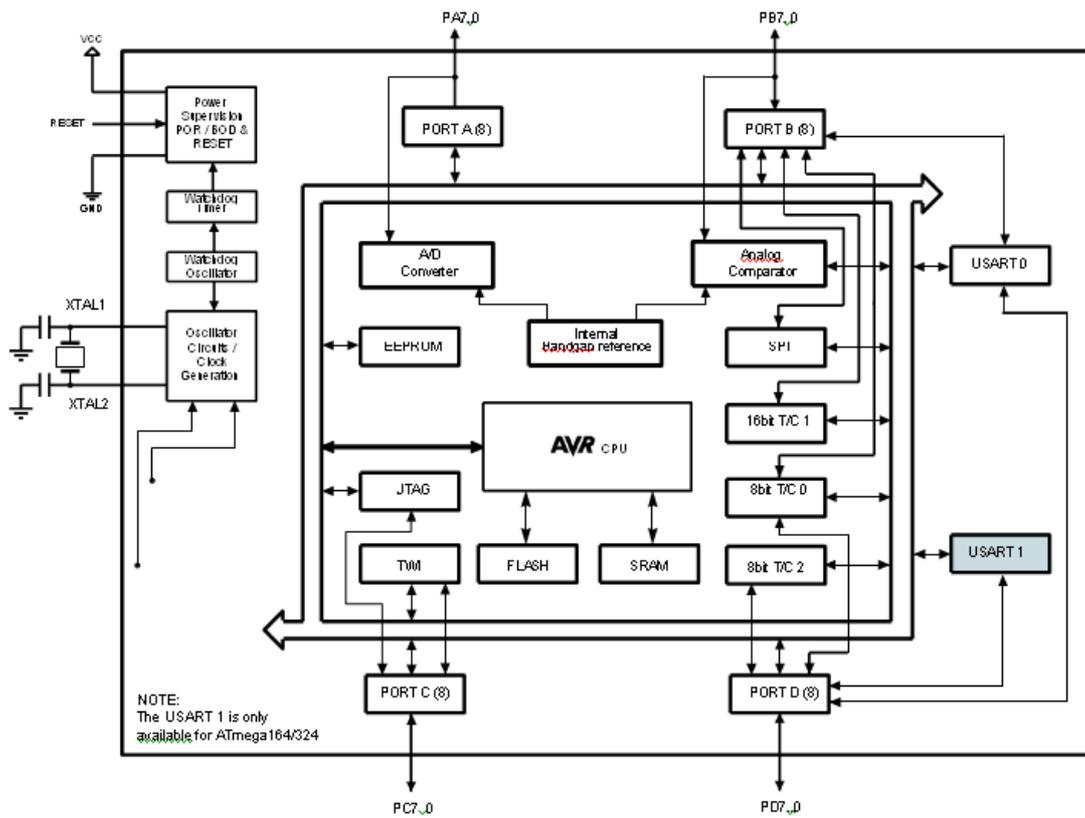
Typical values contained in this datasheet are based on simulations and characterization of other AVR microcontrollers manufactured on the same process technology. Min and Max values will be available after the device is characterized.

Overview

The ATmega164/324/644 is a low-power CMOS 8-bit microcontroller based on the AVR enhanced RISC architecture. By executing powerful instructions in a single clock cycle, the ATmega164/324/644 achieves throughputs approaching 1 MIPS per MHz allowing the system designer to optimize power consumption versus processing speed.

Block Diagram

Figure 3. Block Diagram



ATmega164/324/644

The AVR core combines a rich instruction set with 32 general purpose working registers. All the 32 registers are directly connected to the Arithmetic Logic Unit (ALU), allowing two independent registers to be accessed in one single instruction executed in one clock cycle. The resulting architecture is more code efficient while achieving throughputs up to ten times faster than conventional CISC microcontrollers.

The ATmega164/324/644 provides the following features: 16/32/64K bytes of In-System Programmable Flash with Read-While-Write capabilities, 512B/1K/2K bytes EEPROM, 1/2/4K bytes SRAM, 32 general purpose I/O lines, 32 general purpose working registers, Real Time Counter (RTC), three flexible Timer/Counters with compare modes and PWM, 2 USARTs, a byte oriented 2-wire Serial Interface, a 8-channel, 10-bit ADC with optional differential input stage with programmable gain, programmable Watchdog Timer with Internal Oscillator, an SPI serial port, IEEE std. 1149.1 compliant JTAG test interface, also used for accessing the On-chip Debug system and programming and six software selectable power saving modes. The Idle mode stops the CPU while allowing the SRAM, Timer/Counters, SPI port, and interrupt system to continue functioning. The Power-down mode saves the register contents but freezes the Oscillator, disabling all other chip functions until the next interrupt or Hardware Reset. In Power-save mode, the asynchronous timer continues to run, allowing the user to maintain a timer base while the rest of the device is sleeping. The ADC Noise Reduction mode stops the CPU and all I/O modules except Asynchronous Timer and ADC, to minimize switching noise during ADC conversions. In Standby mode, the Crystal/Resonator Oscillator is running while the rest of the device is sleeping. This allows very fast start-up combined with low power consumption. In Extended Standby mode, both the main Oscillator and the Asynchronous Timer continue to run.

The device is manufactured using Atmel's high-density nonvolatile memory technology. The On-chip ISP Flash allows the program memory to be reprogrammed in-system through an SPI serial interface, by a conventional nonvolatile memory programmer, or by an On-chip Boot program running on the AVR core. The boot program can use any interface to download the application program in the application Flash memory. Software in the Boot Flash section will continue to run while the Application Flash section is updated, providing true Read-While-Write operation. By combining an 8-bit RISC CPU with In-System Self-Programmable Flash on a monolithic chip, the Atmel ATmega164/324/644 is a powerful microcontroller that provides a highly flexible and cost effective solution to many embedded control applications.

The ATmega164/324/644 AVR is supported with a full suite of program and system development tools including: C compilers, macro assemblers, program debugger/simulators, in-circuit emulators, and evaluation kits.

Comparison Between
ATmega164,
ATmega324 and
ATmega644

Table 1 Differences between ATmega164 and ATmega644

Device	Flash	EEPROM	RAM
ATmega164	16 Kbyte	512 Bytes	1 Kbyte
ATmega324	32 Kbyte	1 Kbyte	2 Kbyte
ATmega644	64 Kbyte	2 Kbyte	4 Kbyte

Pin Descriptions

VCC	Digital supply voltage.
GND	Ground.
Port A (PA7 _{..} PA0)	<p>Port A serves as <u>analog</u> inputs to the <u>Analog-to-digital</u> Converter.</p> <p>Port A also serves as an 8-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port A output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port A pins that are externally pulled low will source current if the pull-up resistors are activated. The Port A pins are tri-stated when a reset condition becomes active, even if the clock is not running.</p> <p>Port A also serves the functions of various special features of the ATmega164/324/644 as listed on page 71.</p>
Port B (PB7 _{..} PB0)	<p>Port B is an 8-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port B output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port B pins that are externally pulled low will source current if the pull-up resistors are activated. The Port B pins are tri-stated when a reset condition becomes active, even if the clock is not running.</p> <p>Port B also serves the functions of various special features of the ATmega164/324/644 as listed on page 73.</p>
Port C (PC7 _{..} PC0)	<p>Port C is an 8-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port C output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port C pins that are externally pulled low will source current if the pull-up resistors are activated. The Port C pins are tri-stated when a reset condition becomes active, even if the clock is not running.</p> <p>Port C also serves the functions of the JTAG interface, along with special features of the ATmega164/324/644 as listed on page 76.</p>
Port D (PD7 _{..} PD0)	<p>Port D is an 8-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port D output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port D pins that are externally pulled low will source current if the pull-up resistors are activated. The Port D pins are tri-stated when a reset condition becomes active, even if the clock is not running.</p> <p>Port D also serves the functions of various special features of the ATmega164/324/644 as listed on page 78.</p>
RESET	Reset input. A low level on this pin for longer than the minimum pulse length will generate a reset, even if the clock is not running. The minimum pulse length is given in Table 20 on page 44. Shorter pulses are not guaranteed to generate a reset.
XTAL1	Input to the inverting Oscillator amplifier and input to the internal clock operating circuit.
XTAL2	Output from the inverting Oscillator amplifier.
AVCC	AVCC is the supply voltage pin for Port F and the <u>Analog-to-digital</u> Converter. It should be externally connected to V _{CC} , even if the ADC is not used. If the ADC is used, it should be connected to V _{CC} through a low-pass filter.
AREF	This is the <u>analog</u> reference pin for the <u>Analog-to-digital</u> Converter.

ANEXO 2:
DESCRIPCIÓN DEL SENSOR DE
DISTANCIA POR
ULTRASONIDOS SRF08

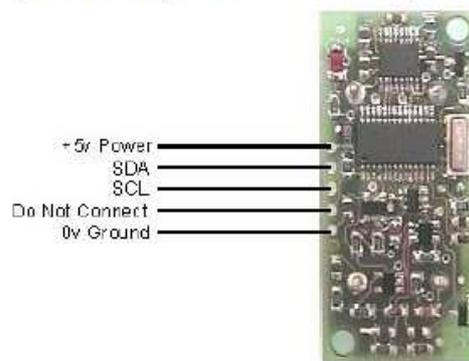
SRF08 Ultra sonic range finder

Technical Specification

Communication with the SRF08 ultrasonic rangefinder is via the I2C bus. This is available on popular controllers such as the OOPic and Stamp BS2p, as well as a wide variety of micro-controllers. To the programmer the SRF08 behaves in the same way as the ubiquitous 24xx series eeprom's, except that the I2C address is different. The default shipped address of the SRF08 is 0xE0. It can be changed by the user to any of 16 addresses E0, E2, E4, E6, E8, EA, EC, EE, F0, F2, F4, F6, F8, FA, FC or FE, therefore up to 16 sonar's can be used. In addition to the above addresses, all sonar's on the I2C bus will respond to address 0 – the General Broadcast address. This means that writing a ranging command to I2C address 0 (0x00) will start all sonar's ranging at the same time. This should be useful in ANN Mode (See below). The results must be read individually from each sonar's real address. We have [examples](#) of using the SRF08 module with a wide range of popular controllers.

Connections

The "Do Not Connect" pin should be left unconnected. It is actually the CPU MCLR line and is used once only in our workshop to program the PIC16F872 on-board after assembly, and has an internal pull-up resistor. The SCL and SDA lines should each have a pull-up resistor to +5v somewhere on the I2C bus. You only need one pair of resistors, not a pair for every module. They are normally located with the bus master rather than the slaves. The SRF08 is always a slave – never a bus master. If you need them, I recommend 1.8k resistors. Some modules such as the OOPic already have pull-up resistors and you do not need to add any more.



Registers

The SRF08 appears as a set of 36 registers.

Location	Read	Write
0	Software Revision	Command Register
1	Light Sensor	Max Gain Register (default 31)
2	1st Echo High Byte	Range Register (default 255)
3	1st Echo Low Byte	N/A
~~~~	~~~~	~~~~
34	17th Echo High Byte	N/A
35	17th Echo Low Byte	N/A

Only locations 0, 1 and 2 can be written to. Location 0 is the command register and is used to start a ranging session. It cannot be read. Reading from location 0 returns the SRF08 software revision. By default, the ranging lasts for 65mS, but can be changed by writing to the range register at location 2. If you do so, then you will likely need to change the analogue gain by writing to location 1. See the Changing Range and Analogue Gain sections below.

Location 1 is the onboard light sensor. This data is updated every time a new ranging command has completed

and can be read when range data is read. The next two locations, 2 and 3, are the 16bit unsigned result from the latest ranging – high byte first. The meaning of this value depends on the command used, and is either the range in inches, or the range in cm or the flight time in uS. A value of zero indicates that no objects were detected. There are up to a further 16 results indicating echo's from more distant objects.

#### Commands

There are three commands to initiate a ranging (80 to 82), to return the result in inches, centimeters or microseconds. There is also an ANN mode (Artificial Neural Network) mode which is described later and a set of commands to change the I2C address.

Command		Action
Decimal	Hex	
80	0x50	Ranging Mode – Result in inches
81	0x51	Ranging Mode – Result in centimeters
82	0x52	Ranging Mode – Result in micro-seconds
83	0x53	ANN Mode – Result in inches
84	0x54	ANN Mode – Result in centimeters
85	0x55	ANN Mode – Result in micro-seconds
160	0xA0	1st in sequence to change I2C address
165	0xA5	3rd in sequence to change I2C address
170	0xAA	2nd in sequence to change I2C address

#### Ranging Mode

To initiate a ranging, write one of the above commands to the command register and wait the required amount of time for completion and read as many results as you wish. The echo buffer is cleared at the start of each ranging. The first echo range is placed in locations 2,3. the second in 4,5, etc. If a location (high and low bytes) is 0, then there will be no further reading in the rest of the registers. The default and recommended time for completion of ranging is 65mS, however you can shorten this by writing to the range register before issuing a ranging command. Light sensor data at location 1 will also have been updated after a ranging command.

#### ANN Mode

ANN mode (Artificial Neural Network) is designed to provide the multi echo data in a way that is easier to input to a neural network, at least I hope it is – I've not actually done it yet. ANN mode provides a 32 byte buffer (locations 4 to 35 inclusive) where each byte represents the 65536uS maximum flight time divided into 32 chunks of 2048uS each – equivalent to about 352mm of range. If an echo is received within a bytes time slot then it will be set to non-zero, otherwise it will be zero. So if an echo is received from within the first 352mm, location 4 will be non-zero. If an object is detected 3m away the location 12 will be non-zero ( $3000/352 = 8$ ) ( $8+4=12$ ). Arranging the data like this should be better for a neural net than the other formats. The input to your network should be 0 if the byte is zero and 1 if its non-zero. I have a SOFM (Self Organizing Feature Map) in mind for the neural net, but will hopefully be useful for any type.

Location 4	Location 5	Location 6	Location 7	Locations 8 - 35
0 – 352mm	353 – 705mm	706 – 1057mm	1058 – 1410mm	and so on

Locations 2,3 contain the range of the nearest object converted to inches, cm or uS and is the same as for Ranging Mode.

#### Checking for Completion of Ranging

You do not have to use a timer on your own controller to wait for ranging to finish. You can take advantage of the fact that the SRF08 will not respond to any I2C activity whilst ranging. Therefore, if you try to read from the SRF08 (we use the software revision number a location 0) then you will get 255 (0xFF) whilst ranging. This is because the I2C data line (SDA) is pulled high if nothing is driving it. As soon as the ranging is complete the SRF08 will again respond to the I2C bus, so just keep reading the register until its not 255 (0xFF) anymore. You can then read the sonar data. Your controller can take advantage of this to perform other tasks while the SRF08 is ranging.

#### Changing the Range

The maximum range of the SRF08 is set by an internal timer. By default, this is 65mS or the equivalent of 11 metres of range. This is much further than the 6 metres the SRF08 is actually capable of. It is possible to reduce the time the SRF08 listens for an echo, and hence the range, by writing to the range register at location 2. The range can be set in steps of about 43mm (0.043m or 1.68 inches) up to 11 metres.

The range is ((Range Register x 43mm) + 43mm) so setting the Range Register to 0 (0x00) gives a maximum range of 43mm. Setting the Range Register to 1 (0x01) gives a maximum range of 86mm. More usefully, 24 (0x18) gives a range of 1 metre and 140 (0x8C) is 6 metres. Setting 255 (0xFF) gives the original 11 metres (255 x 43 + 43 is 11008mm). There are two reasons you may wish to reduce the range.

1. To get at the range information quicker
2. To be able to fire the SRF08 at a faster rate.

If you only wish to get at the range information a bit sooner and will continue to fire the SRF08 at 65ms or slower, then all will be well. However if you wish to fire the SRF08 at a faster rate than 65mS, you will definitely need to reduce the gain – see next section.

The range is set to maximum every time the SRF08 is powered-up. If you need a different range, change it once as part of your system initialization code.

#### Analogue Gain

The analogue gain register sets the *Maximum* gain of the analogue stages. To set the maximum gain, just write one of these values to the gain register at location 1. During a ranging, the analogue gain starts off at its minimum value of 94. This is increased at approx. 70uS intervals up to the maximum gain setting, set by register 1. Maximum possible gain is reached after about 390mm of range. The purpose of providing a limit to the maximum gain is to allow you to fire the sonar more rapidly than 65mS. Since the ranging can be very short, a new ranging can be initiated as soon as the previous range data has been read. A potential hazard with this is that the second ranging may pick up a distant echo returning from the previous "ping", give a false result of a close by object when there is none. To reduce this possibility, the maximum gain can be reduced to limit the modules sensitivity to the weaker distant echo, whilst still able to detect close by objects. The maximum gain setting is stored only in the CPU's RAM and is initialized to maximum on power-up, so if you only want do a ranging every 65mS, or longer, you can ignore the Range and Gain Registers.

**Note** – Effective in Ranging Mode only, in ANN mode, gain is controlled automatically.

Gain Register		Maximum Analogue Gain
Decimal	Hex	
0	0x00	Set Maximum Analogue Gain to 94
1	0x01	Set Maximum Analogue Gain to 97
2	0x02	Set Maximum Analogue Gain to 100
3	0x03	Set Maximum Analogue Gain to 103
4	0x04	Set Maximum Analogue Gain to 107
5	0x05	Set Maximum Analogue Gain to 110
6	0x06	Set Maximum Analogue Gain to 114
7	0x07	Set Maximum Analogue Gain to 118
8	0x08	Set Maximum Analogue Gain to 123
9	0x09	Set Maximum Analogue Gain to 128
10	0x0A	Set Maximum Analogue Gain to 133
11	0x0B	Set Maximum Analogue Gain to 139
12	0x0C	Set Maximum Analogue Gain to 145
13	0x0D	Set Maximum Analogue Gain to 152
14	0x0E	Set Maximum Analogue Gain to 159
15	0x0F	Set Maximum Analogue Gain to 168
16	0x10	Set Maximum Analogue Gain to 177
17	0x11	Set Maximum Analogue Gain to 187
18	0x12	Set Maximum Analogue Gain to 199
19	0x13	Set Maximum Analogue Gain to 212
20	0x14	Set Maximum Analogue Gain to 227

21	0x15	Set Maximum Analogue Gain to 245
22	0x16	Set Maximum Analogue Gain to 265
23	0x17	Set Maximum Analogue Gain to 288
24	0x18	Set Maximum Analogue Gain to 317
25	0x19	Set Maximum Analogue Gain to 352
26	0x1A	Set Maximum Analogue Gain to 395
27	0x1B	Set Maximum Analogue Gain to 450
28	0x1C	Set Maximum Analogue Gain to 524
29	0x1D	Set Maximum Analogue Gain to 626
30	0x1E	Set Maximum Analogue Gain to 777
31	0x1F	Set Maximum Analogue Gain to 1025

Note that the relationship between the Gain Register setting and the actual gain is not a linear one. Also there is no magic formula to say "use this gain setting with that range setting". It depends on the size, shape and material of the object and what else is around in the room. Try playing with different settings until you get the result you want. If you appear to get false readings, it may be echo's from previous "pings", try going back to firing the SRF08 every 65mS or longer (slower).

If you are in any doubt about the Range and Gain Registers, remember they are automatically set by the SRF08 to their default values when it is powered-up. You can ignore and forget about them and the SRF08 will work fine, detecting objects up to 6 metres away every 65mS or slower.

#### Light Sensor

The SRF08 has a light sensor on-board. A reading of the light intensity is made by the SRF08 each time a ranging takes place in either Ranging or ANN Modes ( The A/D conversion is actually done just before the "ping" whilst the +/- 10v generator is stabilizing). The reading increases as the brightness increases, so you will get a maximum value in bright light and minimum value in darkness. It should get close to 2-3 in complete darkness and up to about 248 (0xF8) in bright light. The light intensity can be read from the Light Sensor Register at location 1 at the same time that you are reading the range data.

#### LED

The red LED is used to flash out a code for the I2C address on power-up (see below). It also gives a brief flash during the "ping" whilst ranging.

#### Changing the I2C Bus Address

To change the I2C address of the SRF08 you must have only one sonar on the bus. Write the 3 sequence commands in the correct order followed by the address. Example; to change the address of a sonar currently at 0xE0 (the default shipped address) to 0xF2, write the following to address 0xE0; (0xA0, 0xAA, 0xA5, 0xF2 ). These commands must be sent in the correct sequence to change the I2C address, additionally, No other command may be issued in the middle of the sequence. The sequence must be sent to the command register at location 0, which means 4 separate write transactions on the I2C bus. When done, you should label the sonar with its address, however if you do forget, just power it up without sending any commands. The SRF08 will flash its address out on the LED. One long flash followed by a number of shorter flashes indicating its address. The flashing is terminated immediately on sending a command the SRF08.

Address		Long Flash	Short flashes
Decimal	Hex		
224	E0	1	0
226	E2	1	1
228	E4	1	2
230	E6	1	3
232	E8	1	4
234	EA	1	5
236	EC	1	6
238	EE	1	7
240	F0	1	8
242	F2	1	9

244	F4	1	10
246	F6	1	11
248	F8	1	12
250	FA	1	13
252	FC	1	14
254	FE	1	15

Take care not to set more than one sonar to the same address, there will be a bus collision and very unpredictable results.

#### Current Consumption

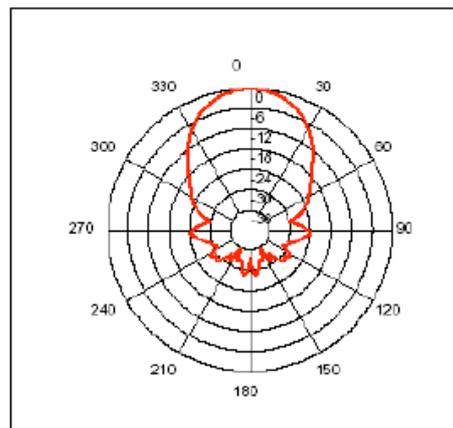
Average current consumption measured on our prototype is around 12mA during ranging, and 3mA standby. The module will automatically go to standby mode after a ranging, whilst waiting for a new command on the I2C bus. The actual measured current profile is as follows;

Operation	Current	Duration
Ranging command received - Power on	275mA	3uS
+/- 10v generator Stabilization	25mA	600uS
8 cycles of 40kHz "ping"	40mA	200uS
Ranging	11mA	65mS max
Standby	3mA	indefinite

The above values are for guidance only, they are not tested on production units.

#### Changing beam pattern and beam width

You can't! This is a question which crops up regularly, however there is no easy way to reduce or change the beam width that I'm aware of. The beam pattern of the SRF08 is conical with the width of the beam being a function of the surface area of the transducers and is fixed. The beam pattern of the transducers used on the SRF08, taken from the manufacturers data sheet, is shown below.



There is more information in the [sonar faq](#).

Here you can have a look at the [schematic](#) and [software](#)

Your feedback/comments/criticisms and wish lists are very welcome, as always.