



RESIDENCIA PROFESIONAL

**SISTEMA DIGITAL FONETICO PARA LA COMUNICACIÓN
DE PERSONAS CON DISCAPACIDADES EN EL HABLA**





Índice.

Capitulo 1.- Generalidades	3
Objetivos	3
Justificación proyecto	3
Caracterización área de trabajo	4
Introducción	5
Capitulo 2.- Marco Teórico	6
2.1 Producción del habla	6
2.1.1 Cavidades infraglólicas	7
2.1.2 Cavidad laríngea	7
2.1.3 Cavidades supraglólicas	9
2.2 Fonemas y Sonidos	11
2.2.1 Los fonemas vocálicos	12
2.2.2 Los fonemas consonánticos	14
2.2.3 Fonemas y letras	16
2.3. Memoria EEPROM isd25xx	18
2.3.1 Descripción de patillaje	19
2.3.2 Modos de Función	22
2.4 Descripción general del puerto paralelo	27
2.5 Descripción general del Pic16f877	29
2.5.1 Diagrama de bloques Pic16f877	33
2.5.2 El archivo de Registros	34
2.6 Descripción General del Teclado PS/2	35
2.6.1 Conexión eléctrica Teclado PS/2	37
2.6.2 Protocolo de Comunicación PS/2	40
Capitulo 3.- Desarrollo del Proyecto	43
3.1 Diseños de Diagramas y simulación	45



3.2Prueba Reales, Placas, Ensamble y Correcciones	55
Conclusiones y Sugerencias	60
Recomendaciones y Trabajos Futuros	62
Referencias	63
Anexo A (esquemáticos)	64
Anexo B (Pcb's)	68
Anexos C (programas)	69



Capítulo 1. Generalidades

Objetivos:

Generales:

Diseñar un sistema de reproducción de frases basadas en fonemas almacenadas en una memoria de audio y que pueda ser usado en la comunicación de personas con problemas de habla.

Específicos:

- Diseñar el dispositivo que permita grabar y reproducir el sonido.
- Desarrollar el programa que generará la selección de las frases a reproducir.
- Integrar el sistema para aplicar su funcionamiento.

Justificación del Proyecto:

Este proyecto está pensado para diferentes tipos de personas con problemas de habla, les permitirá aprender a pronunciar cada palabra o incluso comunicarse con otras personas. Beneficia a personas que han perdido la voz, permitiéndoles comunicarse mejor. Es útil porque ayuda a las personas afónicas en el desarrollo de su actividad social. Psicológicamente una persona mejora su autoestima. Además es de bajo costo, de fácil manejo y fácil mantenimiento.



Caracterización Área de Trabajo:

El lugar en que desarrollamos este proyecto fu en el área de investigación y desarrollo tecnologico contamos con todas las herramientas necesarias para desarrollar el proyecto como son: desarmadores de precisión, cautín, protoboard, osciloscopios, fuentes de voltaje conmutadas, PC's de estación, mesas de trabajo, etc.

Los problemas que pretendemos resolver son:

- Crear un sistema digital fonético que permita el desarrollo del habla en personas con discapacidad.
- Construir una biblioteca de todos los fonemas para poder construir con estas las frases o palabras que se quieran reproducir.
- Crear una interfaz ISD-PC que nos permita grabar dentro de la memoria ISD la librería de fonemas de manera fidedigna.



Introducción.

Este reporte contiene la información necesaria y básica para poder construir un grabador isd, así como un reproductor de frases fonéticas, que podrá ayudar a personas con discapacidades del habla, aunque antes de llegar a esto debemos tener un amplio conocimiento sobre fonética y fonología para de esta forma construir una librería de fonemas fidedigna.

Existen muchos sintetizadores de voz en el mercado la gran mayoría como software de PC pensados únicamente en la lectura para personas con problemas en la vista o incluso ciegas, estos sintetizadores son de un costo aproximado de 500 €. También existen implantes cerebrales para personas mudas o con disfunciones del habla sin embargo estos implantes son de alto riesgo y siempre existe la posibilidad de dañar de forma permanente al cerebro.

Nosotros escogimos la posibilidad de crear un reproductor de fonemas basándonos en la tecnología ISD y los microcontroladores Pic.

Electrónicamente podremos comprender el funcionamiento de una memoria isd2560 que permite la grabación y reproducción de sonido analógicamente así como la manera de conectarlo a un Pic, un LCD y un teclado Ps/2.



Capitulo 2.-MARCO TEORICO.

2.1 Producción del Habla.

El habla, como señal acústica, se produce a partir de las ondas de presión que salen de la boca y las fosas nasales de un locutor. El proceso comienza con la generación de la energía suficiente (flujo de aire) en los pulmones, la modificación de ese flujo de aire en las cuerdas vocales, y su posterior perturbación por algunas constricciones y configuraciones de los órganos superiores. Así, en el proceso fonador intervienen distintos órganos a lo largo del llamado tracto vocal, que en nuestro caso asumiremos que se restringe a la zona comprendida entre las cuerdas vocales y las aberturas finales: los labios y las fosas nasales.

En la figura 1 podemos observar el conjunto de órganos que intervienen en la fonación estos pueden dividirse en tres grupos bastante bien delimitados:

- 1) Cavidades infraglóticas (sistema sub-glotal) u órgano respiratorio
- 2) Cavidad laríngea u órgano fonador
- 3) Cavidades supraglóticas

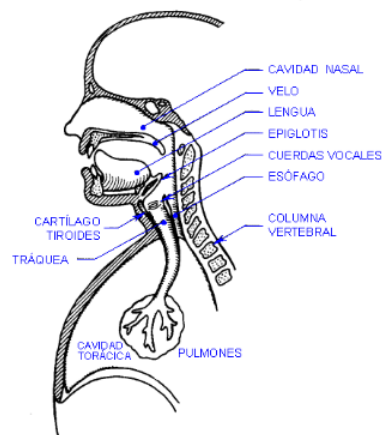


Figura1.1- partes del sistema fonético del cuerpo humano



2.1.1 Cavidades infraglólicas

Las cavidades infraglólicas constan de los *órganos propios de la respiración* (pulmones, bronquios, y tráquea), que son la fuente de energía para todo el proceso de producción de voz.

En el proceso de inspiración, los pulmones toman aire, bajando el diafragma y agrandando la cavidad torácica. En el momento de la fonación, la espiración, provocada por la contracción de los músculos intercostales y del diafragma, aporta la energía necesaria para generar la onda de presión acústica que atravesará los órganos fonadores superiores.

2.1.2 Cavidad laríngea

La cavidad laríngea es la responsable de modificar el flujo de aire generado por los pulmones y convertirlo (o no, como veremos), en una señal susceptible de excitar adecuadamente las posibles configuraciones de las cavidades supraglólicas.

El último cartílago de la tráquea, el *cricoides*, forma la base de la laringe, cuyo principal órgano son las cuerdas vocales que son dos pares de repliegues compuestos de ligamentos y músculos (figura 2). El par inferior son las llamadas cuerdas vocales verdaderas, que pueden juntarse o separarse mediante la acción de los músculos *crico-aritenoides* lateral y posterior, y que están protegidas en su parte anterior por el *cartílago tiroides*, el más importante de la laringe, abierto por su parte posterior. Finalmente, la parte superior de la laringe está unida al *hueso hioides*.

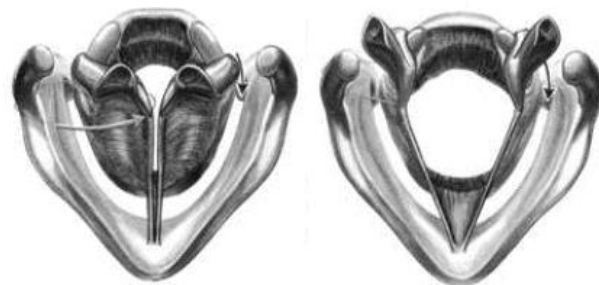


Figura1.2.- cuerdas vocales



En la figura 3 se muestra una vista transversal simplificada de la zona en la que se encuentran las cuerdas vocales, en sus posiciones extremas: abiertas y cerradas; y en la Ilustración siguiente puede verse una fotografía real de las mismas. A la apertura que queda entre las cuerdas vocales se le denomina glotis.

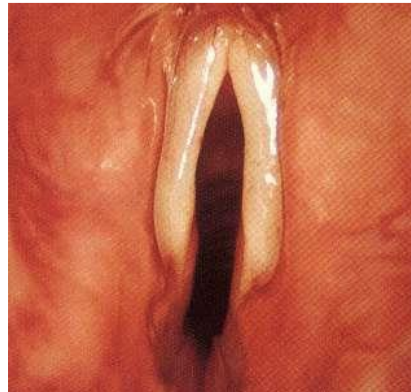


Figura1.3.- cavidad laríngea, cuerdas vocales

La cavidad laríngea está terminada por la *epiglotis*, un cartílago en forma de cuchara, que permite cerrar la apertura de la laringe en el acto de la deglución.

La distinción fundamental entre los sonidos se basa en su característica de sonoridad. En los sonidos sonoros, incluyendo las vocales, se observa un patrón regular tanto en su estructura temporal como en su estructura frecuencial patrón del que carecen los sonidos sordos la cualidad de sonoridad de los sonidos sonoros se produce por la acción vibradora de las cuerdas vocales.

El mecanismo de la vibración se produce de la siguiente forma: Si suponemos que inicialmente las cuerdas vocales están juntas, la presión subglotal se incrementa lo suficiente para forzar a las cuerdas vocales a separarse. Al separarse, el aire pasa a través de ellas y la presión subglotal disminuye, momento en el que la fuerza de los músculos hace que las cuerdas vocales vuelvan a juntarse. Cuando las cuerdas vocales se juntan, el flujo de aire disminuye y la presión subglotal aumenta de nuevo, con lo que se vuelve a reproducir el ciclo, y esta vibración de las cuerdas vocales produce pulsos casi periódicos de aire que excitan el sistema por encima de la laringe.



A esta frecuencia de vibración se la denomina frecuencia fundamental, y sus valores típicos oscilan entre los 60 Hz. para un hombre voluminoso, y los 300 Hz. para una mujer o un niño. La señal generada en las cuerdas vocales puede variar en frecuencia e intensidad según varíe la masa, la longitud y la tensión de las mismas.

2.1.3 Cavidades supraglóticas

Las cavidades supraglóticas están constituidas por la faringe, la cavidad bucal y la cavidad nasal. Su misión fundamental de cara a la fonación es perturbar adecuadamente el flujo de aire procedente de la laringe, para dar lugar finalmente a la señal acústica generada a la salida de la nariz y la boca.

La faringe es una cavidad en forma tubular que une la laringe con las cavidades bucal y nasal, y que suele dividirse en tres partes: faringe laríngea, faringe bucal (boca) y faringe nasal, las dos últimas separadas por el velo del paladar. El volumen de la faringe laríngea puede ser modificado por los movimientos de la laringe, la lengua y la epiglotis mientras que el volumen de la faringe bucal se modifica por el movimiento de la lengua.

La faringe nasal y las restantes cavidades nasales forman, desde el punto de vista de su acción sobre el flujo de aire procedente de la faringe, un *resonador* que puede o no conectarse al resonador bucal mediante la acción del velo del paladar. Según el resonador nasal esté o no conectado, el sonido será nasal u oral, respectivamente.

Si hacemos una descripción de la cavidad bucal (esquemática en la figura 4), podemos señalar las siguientes partes:

- □ Los labios en el extremo.
- □ Los dientes.
- □ La zona alveolar, entre los dientes y el paladar duro.
- □ El paladar, en el que a su vez, y de forma simplificada, podemos distinguir el paladar duro y el paladar blando o velo.



La raíz de la lengua forma la pared frontal de la faringe laríngea, y sus movimientos le permiten modificar la sección de la cavidad bucal (movimiento vertical), adelantar o retrasar su posición frente a la de reposo (movimiento horizontal), así como poner en contacto su ápice o la parte trasera con alguna zona del paladar.

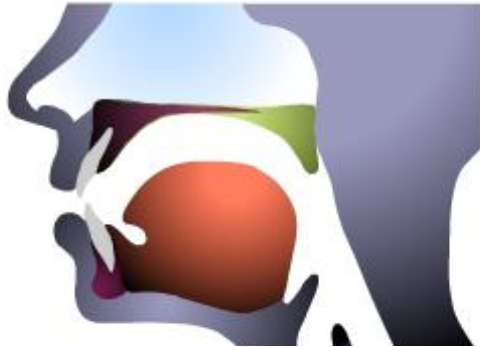


Figura 1.4.- cavidad vocal

El movimiento de los labios también interviene en la articulación, pudiendo ser de apertura o cierre y de protuberancia, alargando en este último caso la cavidad bucal.

De los movimientos de los órganos supraglotales surgen los distintos modos de articulación de los posibles sonidos emitidos por un locutor. En la mayor parte de los casos es un órgano el que se mueve (activo) y otro contra el que se efectúa la articulación (pasivo) y según la pareja de órganos activo/pasivo que tengamos, tenemos una serie de posibles articulaciones.



2.2.- Fonemas y sonidos.

Llamamos sonido a la realización física de un fonema.

Si varias personas pronuncian la palabra *tren*, por ejemplo, se notarán diferencias en la pronunciación más o menos marcadas. La *t* sonará más o menos enérgica; la *r* vibrará más o menos... Incluso si la misma persona pronuncia la palabra en situaciones diferentes, se notarán variaciones. Estas variaciones, perceptibles al oído, se notarán mucho más si utilizamos aparatos especiales. Esta realización física de la *t* o de la *r* es lo que llamamos sonido. Los sonidos de una lengua son innumerables, tantos como hablantes e, incluso tantos como empleos hace de ellos cada hablante.

Llamamos fonema a la imagen mental de un sonido.

En la mente de cualquier hablante no hay más que una *t* o una *r*, aunque después, en la realidad, haya tantísimas formas de pronunciarlas. Esa *t* ideal y única es a lo que llamamos fonema. Los fonemas son muy pocos.

Los fonemas y los sonidos carecen de significado.

La ciencia que estudia los fonemas se llama Fonología y la que estudia los sonidos, Fonética.



2.2.1 Los fonemas vocálicos

Cuando articulamos los sonidos vocálicos, el aire no encuentra obstáculos en su salida desde los pulmones al exterior. Para clasificar estos fonemas, tendremos en cuenta los siguientes factores:

- La localización (punto de articulación). Se refiere a la parte de la boca donde se articulan. Pueden ser anteriores (/e/, /i/), medio o central (/a/) o posteriores (/o/, /u/).
- La abertura (modo de articulación). Se refiere a la abertura de la boca al pronunciarlos. Pueden ser de abertura máxima o abierto (/a/), de abertura media o semiabiertos (/e/, /o/) y de abertura mínima o cerrados (i, u).



Tabla 2.1 CUADRO DE FONEMAS VOCALES (Triángulo de Hellwag)

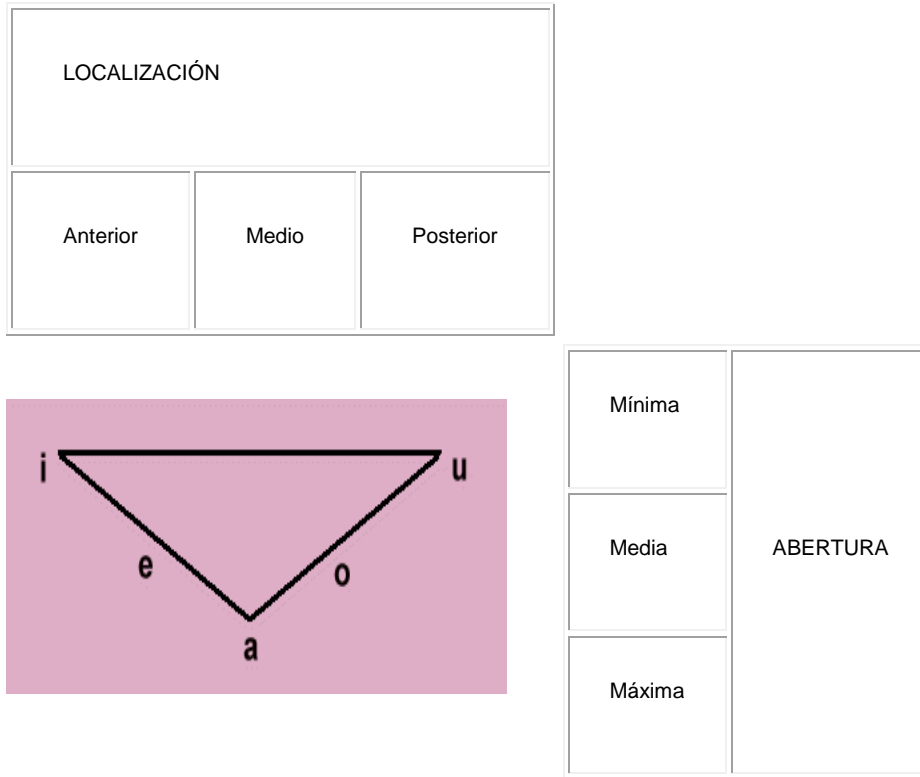


Tabla 2.2 RASGOS DE LOS FONEMAS VOCALES

- /a/ - Localización media y abertura máxima
- /e/ - Localización anterior y abertura media
- /i/ - Localización anterior y abertura mínima
- /o/ - Localización posterior y abertura media
- /u/ - Localización posterior y abertura mínima



2.2.2 Los fonemas consonánticos.

En la articulación de los sonidos consonánticos siempre hay un obstáculo más o menos grande que impide salir el aire desde los pulmones al exterior. Según las circunstancias que rodean esta salida del aire, existen ciertos factores que debemos tener en cuenta a la hora de clasificarlos:

- Zona o punto de articulación. Es el lugar donde toman contacto los órganos que intervienen en la producción del sonido. Por ejemplo, si para producir un sonido entran en contacto los dos labios, se crearán sonidos bilabiales como es el caso de las realizaciones de los fonemas /p/, /b/ y /m/.
- Modo de articulación. Es la postura que adoptan los órganos que producen los sonidos. Por ejemplo, si los órganos cierran total y momentáneamente la salida del aire, los sonidos serán oclusivos. Ese es el caso de los sonidos /p/, /t/ y /k/.
- Actividad de las cuerdas vocales. Cuando producimos sonidos, las cuerdas vocales pueden vibrar o no vibrar. Si las cuerdas vocales no vibran, los sonidos se llaman sordos. Así producimos /p/, /f/, /k/. Cuando, por el contrario, las cuerdas vocales vibran se llaman sonoros. Esto pasa al pronunciar /a/, /b/, /d/.
- Actividad de la cavidad nasal. Si al producir sonidos, parte del aire pasa por la cavidad nasal, los sonidos se llaman nasales. Son de esta clase /m/, /n/, /ñ/. Si todo el aire para por la cavidad bucal se llaman orales. De este tipo son /f/, /e/, /s/.



Tabla 2.3 CUADRO DE LOS FONEMAS CONSONANTES

Bilabial		Labiodental		Interdental		Dental		Alveolar		Palatal		Velar		
-	-	-	-	-	-	-	-	-	-	-	-	-	-	
Sordo	Sonoro	Sordo	Sonoro	Sordo	Sonoro	Sordo	Sonoro	Sordo	Sonoro	Sordo	Sonoro	Sordo	Sonoro	
-	-	-	-	-	-	-	-	-	-	-	-	-	-	
P	B					t	d					k	G	Oclusivos
										ch				Africados
		f		z				S			y	j		Fricativos
									l		ll			Laterales
									r, rr					Vibrantes
	M								n		ñ			Nasales

Tabla 2.4 LISTADO DE LOS RASGOS DE LOS FONEMAS CONSONANTES

/p/ - bilabial, oclusivo, sordo	/ch/ - palatal, africado, sordo
/b/ - bilabial, oclusivo, sonoro	/r/ - alveolar, vibrante, sonoro
/t/ - Dental, oclusivo, sordo	/rr/ - alveolar, vibrante, sonoro
/d/ - Dental, oclusivo, sonoro	/l/ - alveolar, lateral, sonoro
/k/ - velar, oclusivo, sordo	/ll/ - palatal, lateral, sonoro
/g/ - velar, oclusivo, sonoro	/m/ - bilabial, nasal, sonoro
/f/ - labiodental, fricativo, sordo	/n/ - alveolar, nasal, sonoro
/z/ - interdental, fricativo, sordo	/ñ/ - palatal, nasal, sonoro
/s/ - alveolar, fricativo, sordo	/y/ - fricativo, palatal, sonoro
/j/ - velar, fricativo, sordo	



2.2.3 Fonemas y letras

En castellano hay veinticuatro fonemas y veintinueve letras, pero es preciso considerar que su correspondencia con ellas no es total; por lo que hay que tener en cuenta lo siguiente:

Tabla 2.5 de fonemas y representacion

Fonemas	Representación
/b/	Letras B y V
/k/	Letras K y C (delante de A, O, U) y Qu (delante de E, I)
/g/	Letra G (delante de A, O, U)
/z/	Letras Z y C (delante de E, I)
/j/	Letras J y G (delante de E, I)
/r/	Letra R (entre vocales)
/rr/	Letra RR y R (a comienzo de palabra y detrás de consonante)



EL ALFABETO FONÉTICO INTERNACIONAL (actualizado en 2005)

CONSONANTES (INFRAGLOTALES)

	LABIAL		CORONAL				DORSAL			RADICAL		GLOTA
	BILABIAL	LABIODENTAL	DENTAL	ALVEOLAR	POSTALVEOLAR	RETROFLEJA	PALATAL	VELAR	UVULAR	FARÍNGEA	EPIGLOTA	
NASAL	m	ɱ	n				ɳ	ɲ	ŋ	ɴ		
OCLUSIVA	p b	ɸ β	t d				ʈ ɖ	c ɟ	k g	q ɢ		ʔ
FRICATIVA	ɸ β	f v	θ ð	s z	ʃ ʒ	ʂ ʐ	ç ʝ	x ɣ	χ ʁ	ħ ʕ	ħ ʕ	h ɦ
AFROXIMANTE		ʋ	ɹ				ɻ	j	ɰ			
VIBRANTE MÚLTIPLE	B		r						R		ʀ	
VIBRANTE SIMPLE		ʋ	ɾ				ɽ					
FRICATIVA LATERAL			ɬ ɮ			ɮ	ɬ	ɮ				
AFROXIMANTE LATERAL			l				ɭ	ʎ	ʟ			
VIBR. SIMPLE LATERAL			ɭ				ɮ					

Las consonantes alineadas a la izquierda son sordas, las alineadas a la derecha sonoras. Las casillas en gris son articulaciones consideradas imposibles.

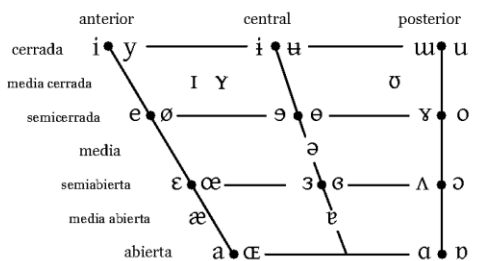
CONSONANTES (SUPRAGLOTALES)

CLIC	IMPLOSIVA	EYECTIVA
◌̥ bilabial	ɓ bilabial	ʼ como en:
◌̦ dental	ɗ dental / alveolar	pʼ bilabial
◌̧ (post)alveolar	f palatal	tʼ dental / alveolar
◌̨ palatoalveolar	ɟ velar	kʼ velar
◌̩ lateral alveolar	ɠ uvular	sʼ fricativa alveolar

CONSONANTES (COARTICULADAS)

- ɱ fricativa labiovelar sorda
- ʋ aproximante labiovelar sonora
- ɰ aproximante labioalveolar sonora
- ç fricativa alveopalatal sorda
- ʝ fricativa alveopalatal sonora
- ɰ y x simultáneas
- kp̚ ts̚ Africadas y dobles articulaciones pueden representarse con dos símbolos atados con una cuña

VOCALES



Las vocales a la izquierda del punto son no labializadas, las de la derecha son labializadas

SUPRASEGMENTALES

- ◌́ acento principal
- ◌̂ acento extra
- ◌̃ acento secundario
- ◌̄ larga
- ◌̅ semilarga
- ◌̆ corta
- ◌̇ breve
- ◌̈ rotura silábica
- ◌̉ enlace
- ◌̊ ENTONACIÓN
- ◌̋ grupo menor (pie)
- ◌̌ grupo mayor (entonación)
- ↗ ascenso global
- ↘ descenso global

TONO

- NIVEL
- ◌̎ extra alto
- ◌̏ alto
- ◌̐ medio
- ◌̑ bajo
- ◌̒ extra bajo
- NIVELACIÓN
- ◌̓ ascendente
- ◌̔ descendente
- ◌̕ ascendente alto
- ◌̖ ascendente bajo
- ◌̗ descendente alto
- ◌̘ descendente bajo
- ◌̙ ascendente descendente
- ◌̚ descendente
- ◌̛ ascendente

DIACRÍTICOS En algunos pueden aparecer arriba: ̂. En superíndice: ̣ (tendencia fricativa), ̤ (sonora mate), ̥ (ataque glotal), ̦ (schwa epentético), ̧ (diptongación)

SILABICIDAD Y TENDENCIA		FONACIÓN		ARTICULACIÓN PRIMARIA		ARTICULACIÓN SECUNDARIA			
ɹ ɳ	silábica	ɳ̥ ɳ̤	ensordecida	ɬ̥ ɬ̤	dental	tʰ ɖʰ	labializada	ɕ ɟ̣	más labializada
ç ɰ	no silábica	ʃ̥ ʃ̤	sonorizada	ʈ̥ ʈ̤	apical	tʰ̣ ɖ̣	palatalizada	ɕ̣ ɟ̣ʰ	menos labializada
tʰ ɖʰ	aspirada	ɓ̣ ɓ̤	sonora mate	ʈ̥ ʈ̤	laminal	tʰ̣ ɖ̣	velarizada	ẽ ẹ̃	nasalizada
ḍ	tendencia nasal	ɓ̣ ɓ̤	sonora estridente	ɹ̣ ɹ̤	avanzada	tʰ̣ ɖ̣	faringizada	ɕ̣ ɟ̣	rotacismo
ḍ	tendencia lateral	ɓ̣ ɓ̤	estridente	ɹ̣ ɹ̤	retraída	ɬ̣ ɮ̣	velarizada o faringizada	ɕ̣ ɟ̣	base de la lengua avanzada
ḍ	tendencia no audible	ɬ̣ ɮ̣	linguolabial	ẹ̃ ạ̈	centralizada	ẹ̃ ụ̃	medio centralizada	ɕ̣ ɟ̣	base de la lengua retraída
ç̣ β̣	descenso lingual (ç es aproximante bilabial)	ç̣ ɹ̣	ascenso lingual (ɹ es fricativa alveolar sonora no silbante)						



2.3.- La memoria EEPROM de ISD25XX

Merece la pena entretenerse en este punto por lo peculiar de esta memoria. El chip de **ISD** ilustrada en el diagrama a bloques de la figura 5 almacena en esta memoria el dato en formato analógico directamente, por lo que se ahorra en el proceso dos conversiones, la A/D al grabar y la D/A al reproducir. El método de grabación es denominado por **ISD** como “break through” (abrir paso) y permite almacenar cada muestra en una celda en formato analógico. La grabación de cada celda es un proceso en bucle cerrado donde se compara el valor de la muestra con el contenido en la celda de memoria y se van inyectando electrones en la misma hasta que se iguala la tensión de la muestra. El dato almacenado tiene una cuantificación equivalente de 256 niveles (ocho bits). El proceso se basa en la tecnología EEPROM floating gate.

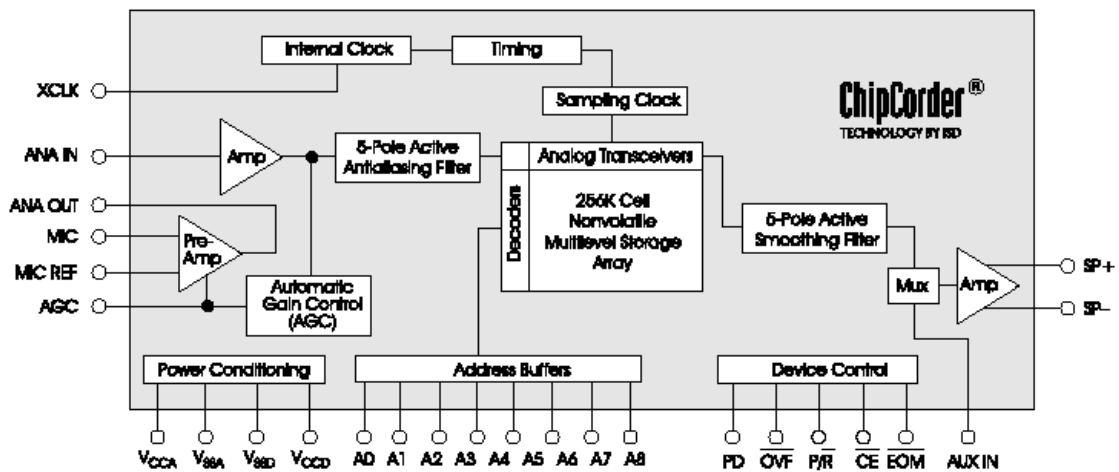


Figura3.1.- Diagrama de Bloques ISD25XX



2.3.1 Descripción de las señales y del patillaje

2.3.1.1 Alimentación (V_{CCA} , V_{CCD} , V_{SSA} , V_{SSD})

La tensión de alimentación es de 5 voltios. Para reducir al mínimo el ruido, las partes digital y analógica de los circuitos disponen de entradas de alimentación independientes.

2.3.1.2 Entrada de corte de corriente (**PD - Power Down**)

Cuando la señal **PD** se pone a nivel alto el circuito entra en el modo de muy bajo consumo. Cuando se produce la condición de desbordamiento de la memoria, **PD** se utiliza para llevar el puntero al inicio de la zona de grabación o reproducción. Además tiene una funcionalidad adicional cuando se trabaja en el *Modo de Función M6 (Push-Button)* que se describirá en la siguiente sección.

2.3.1.3 Entrada para habilitar circuito (**CE – chip enable**)

La entrada **CE** debe ponerse a nivel bajo para habilitar la operación del circuito. Las direcciones de entrada y la señal de grabación/reproducción (**P/R**) son capturadas durante el flanco de bajada de la señal. Además tiene una funcionalidad adicional cuando se trabaja en el *Modo de Función M6 (Push-Button)* que se describirá en la siguiente sección.

2.3.1.4 Entrada de Reproducción/Grabación (**P/R – playback/record**)

La entrada **P/R** permite seleccionar entre el modo de reproducción (nivel alto) y el de grabación (nivel bajo). Durante un ciclo de grabación la dirección de comienzo se proporciona a través de las correspondientes entradas y se detiene cuando se pone a nivel alto cualquiera de las señales **PD** o **CE**, o cuando se produce un desbordamiento.



Cada vez que se termina un ciclo de grabación mediante las señales **PD** o **CE**, se pone una marca de fin de mensaje (**EOM**) en la dirección de memoria donde se finalizó.

Durante un ciclo de reproducción se proporciona la dirección de comienzo y la grabación se reproduce de manera continua hasta que se encuentra una marca de fin de mensaje (**EOM**). El dispositivo puede seguir reproduciendo a continuación de una marca de fin de mensaje (**EOM**) cuando se encuentra en el *Modo de Función*.

2.3.1.5 Señal de fin de mensaje y de funcionamiento (**EOM**)

La salida **EOM** genera un pulso a nivel bajo de duración T_{EOM} al final de cada mensaje. Cuando el dispositivo se configura en el *Modo de Función M6*, esta salida se mantiene a nivel alto mientras el dispositivo está grabando o reproduciendo.

2.3.1.6 Señal de desbordamiento (**OVF**)

Se genera un pulso a nivel bajo cuando se agota el espacio de memoria. La salida **OVF** sigue a la entrada **CE** hasta que se reinicia el puntero de memoria con la señal **PD**. Esta salida puede utilizarse para la conexión de varios dispositivos en cascada.

2.3.1.7 Entrada de micrófono (**MIC**)

La entrada va conectada a un preamplificador integrado junto con un control automático de ganancia (**AGC**) que regula la ganancia de éste desde -15 hasta 24 dB. La conexión de un micrófono externo deberá hacerse a través de un condensador serie de desacoplo que, junto con la resistencia interna de $10\text{ K}\Omega$ de la entrada, determina la frecuencia de corte inferior.



2.3.1.8 Entrada de referencia de micrófono (MIC REF)

Es la entrada no inversora del preamplificador para la conexión del micrófono y proporciona una cancelación de ruido gracias a su elevado grado de rechazo al modo común.

2.3.1.9 Entrada del Control Automático de Ganancia (AGC)

El control automático de ganancia permite cubrir un amplio margen de niveles de entrada procedentes del micrófono, que van desde un pequeño susurro hasta sonidos muy altos. El tiempo de *activación* viene determinado por la constante de tiempo que conforma un condensador externo conectado entre **AGC** y V_{SSA} y la resistencia interna de 5 K Ω . El tiempo de *desactivación* viene determinado por la constante de tiempo asociada al condensador externo anterior junto con una resistencia externa conectada en paralelo con él. Los valores de 4.7 μ F y 470 K Ω para el condensador y la resistencia respectivamente dan en la mayoría de los casos un resultado satisfactorio.

2.3.1.10 Salida analógica (ANA OUT)

La salida del conjunto *preamplificador – control automático de ganancia* se conduce a través de este *pin*.

2.3.1.11 Entrada analógica (ANA IN)

A través de esta entrada se conduce la señal al *chip* para su grabación. La señal de entrada debe desacoplarse con un condensador externo, incluso si procede de la salida **ANA OUT** (caso de un micrófono), debiendo tenerse en cuenta que este condensador



junto con la resistencia interna de $3K\Omega$ de la entrada conforma un filtro paso bajo que pudiera recortar la señal.

2.3.1.12 Entrada de reloj externo (XCLK)

La tolerancia del reloj interno es de $\pm 5\%$. En caso de que se requiera una mayor precisión, puede controlarse con una señal de reloj externa cuyo valor dependerá del dispositivo de la familia que estemos utilizando. De este modo deberán usarse los siguientes valores 1024 Khz., 819.2 Khz., 682.7 Khz. y 512 Khz. para las frecuencias de muestreo de 8 Khz., 6.4 Khz., 5.3 Khz. y 4 Khz. que usan los dispositivos **ISD 2532**, **ISD 2540**, **ISD 2548** e **ISD 2564** respectivamente. En caso de no usarse, esta entrada deberá conectarse a masa.

2.3.1.13 Salidas de altavoz (SP+ y SP-)

Todos los dispositivos de esta familia disponen de una salida diferencial para altavoces capaz de proporcionar 50 mW sobre un altavoz con impedancia de 16Ω si la señal procede de **AUX IN** o 12.2 mW si procede de la memoria.

Nota: Cuando se usan varios dispositivos no se deben conectar en paralelo las salidas de altavoz, ya que podría dañar los dispositivos. Nunca se deben poner a masa las salidas de altavoz.

2.3.1.14 Entrada auxiliar (AUX IN)

La entrada auxiliar permite multiplexar una señal al amplificador de salida y el altavoz cuando las señales **CE** y **P/R** están a nivel alto o cuando se produce un desbordamiento



durante la reproducción. Esto permite conectar la señal de reproducción al siguiente dispositivo de una cascada.

2.3.1.15 Entradas de dirección y modo (AX/MX)

Las entradas de dirección y modo tienen dos funciones dependiendo de cómo se conecten los dos bits más significativos (**A7** y **A8**).

Si cualquiera de estos dos bits o ambos se conectan a masa, las entradas **A0...A8** se interpretan como bits de dirección. Cuando ambos bits (**A7** y **A8**) se conectan a nivel alto, las entradas (**M0...M6**) se interpretan como bits indicativos de cada *modo de función*.

De los siete modos posibles sólo seis están operativos. Es posible combinar varios modos a la vez.

2.3.2 Modos de función

Hay dos consideraciones importantes que hacer sobre el uso de los modos de función.

En primer lugar, todas las acciones comienzan en la dirección cero y las posteriores pueden hacerlo en cualquier posición, dependiendo del modo en cuestión. Además el puntero de direcciones se pone a cero cada vez que se cambia del modo reproducción al modo grabación o viceversa y cuando se ejecuta un ciclo de corte de corriente (*Power-Down*).

En segundo lugar, los Modos de Función se ejecutan cuando la señal **CE** pasa a nivel bajo y los dos bits de dirección más significativos (**A7-A8**) están a nivel alto,



manteniéndose en el modo en cuestión hasta que se produce una nueva transición de nivel alto a bajo en la señal **CE**.

Descripción de los modos de función

M0 – Búsqueda de mensaje

Esta función permite al usuario pasar de mensaje en mensaje sin conocer la dirección de inicio de cada uno. Cada transición a nivel bajo de la señal **CE** hace que el puntero salte al inicio del siguiente mensaje. Este modo inhabilita el amplificador de salida y reproduce el mensaje a una velocidad 800 veces superior a la normal. Funciona como si de un avance rápido se tratase.

M1 – Borrado de los marcadores de fin de mensaje

Permite reproducir los mensajes grabados de modo continuo como si de un solo mensaje se tratara.

M2 – Sin uso

M2 debe ponerse a nivel bajo en el Modo de Función.

M3 – Repetición de mensaje

La función **M3** permite la repetición continua del mensaje situado en la primera posición de la memoria.



M4 – Direccionamiento consecutivo

Durante el funcionamiento normal, el puntero de direcciones se pone a cero cuando se llega al final del mensaje (**EOM**). Este modo inhibe la puesta a cero del puntero y los mensajes son reproducidos consecutivamente.

M5 – Activación por nivel de la entrada CE

Por defecto la activación del chip se realiza con el flanco de bajada de la señal **CE** durante la reproducción y por el nivel detectado de la señal **CE** durante la grabación. El modo de función **M5** provoca que la activación mediante la señal **CE** se realice siempre por la detección del nivel.

M6 – Pulsar Botón

El modo Pulsar Botón se usa, ante todo, en aplicaciones de muy bajo coste y ha sido diseñado para reducir al máximo los circuitos y componentes externos. En este modo el dispositivo siempre entra en desconexión (*power-down*) al final de cada reproducción. En este modo varios de los terminales de entrada-salida tienen otra función diferente que se describe a continuación.

Entrada CE (*START/PAUSE*)

En el modo Pulsar Botón, **CE** actúa como una señal **START/PAUSE** cuando se lleva a nivel bajo. Un pulso a nivel bajo iniciará la grabación o reproducción y el siguiente pulso aplicado generará una pausa. El contador de direcciones no es puesto a cero y otro pulso más provocará que la operación continúe en el punto donde se quedó.



Entrada PD (*STOP/RESET*)

En el modo Pulsar Botón, cuando **PD** se lleva a nivel alto se termina el ciclo en progreso y se pone a cero el contador de direcciones.

Salida EOM (*RUN*)

En este modo, **EOM** se convierte en una señal activa a nivel alto mientras el dispositivo se encuentra funcionando, lo que permite alimentar un LED o activar cualquier otro dispositivo externo.

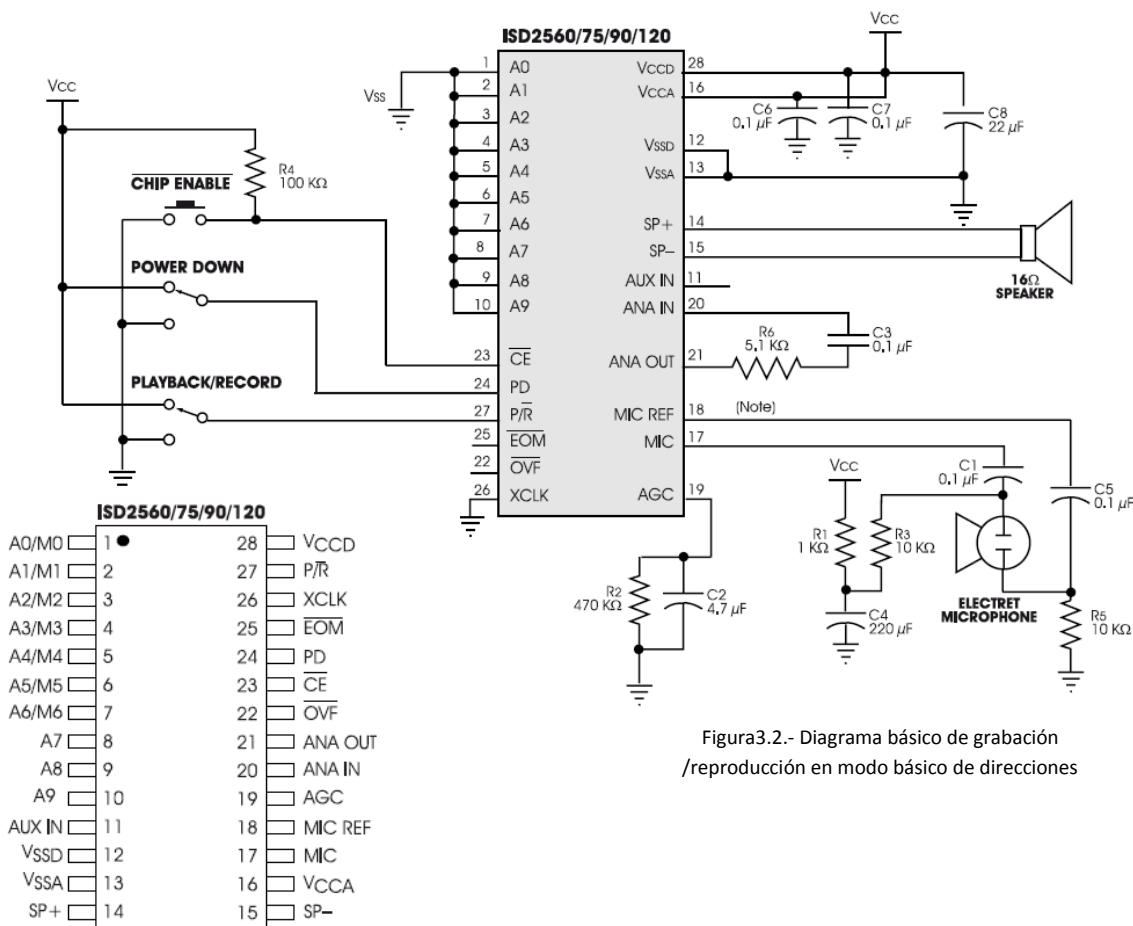


Figura3.2.- Diagrama básico de grabación /reproducción en modo básico de direcciones

Figura 3.4.- patillaje del isd25xx (dip/soic)



2.4.- Descripción General Puerto paralelo.

Aquí tenemos una imagen (figura 7(a)) de los conectores

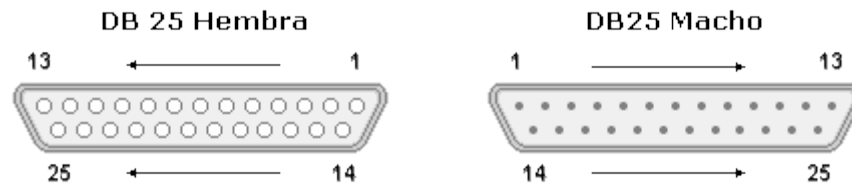


Figura 4.1(a)- conectores DB25

Cada pin de los conectores tiene un número asignado por lo cual es muy importante que tengamos bien identificados los cables que vamos a conectar ya que esto nos podría costar un retraso grande.

Este puerto dispone de tres registros de 8 bit cada uno (un byte). cada uno de estos registros se denominan puertos o PORT., y cada uno de sus bits, representa un pin determinado del puerto. Los pin's que van del 18 al 25 (ambos inclusive): Son para masa, y sirven para conectar las descargas de los circuitos.

Veamos ahora los tres registros Puerto de datos (Pin 2 al 9): Es el PORT 888 y es de solo escritura, por este registro enviaremos los datos al exterior de la pc.

Puerto de estado (Pin 15, 13, 12, 10 y 11): Es el PORT 889 y es de solo lectura, por aquí enviaremos señales eléctricas al ordenador, de este registro solo se utilizan los cinco bits de más peso, que son el bit 7, 6, 5, 4 y 3 teniendo en cuenta que el bit 7 funciona en modo invertido.



Puerto de control (Pin 1, 14, 16 y 17): Es el correspondiente al PORT 890, y es de lectura/escritura, es decir, podremos enviar o recibir señales eléctricas, según nuestras necesidades. De los 8 bits de este registro solo se utilizan los cuatro de menor peso o sea el 0, 1, 2 y 3, con un pequeño detalle, los bits 0, 1, y 3 están invertidos.

En esta imagen (figura(b)) podemos ver los tres registros, sus bits y los pines asignados a cada uno de ellos. La imagen corresponde a un conector DB-25 (Hembra).

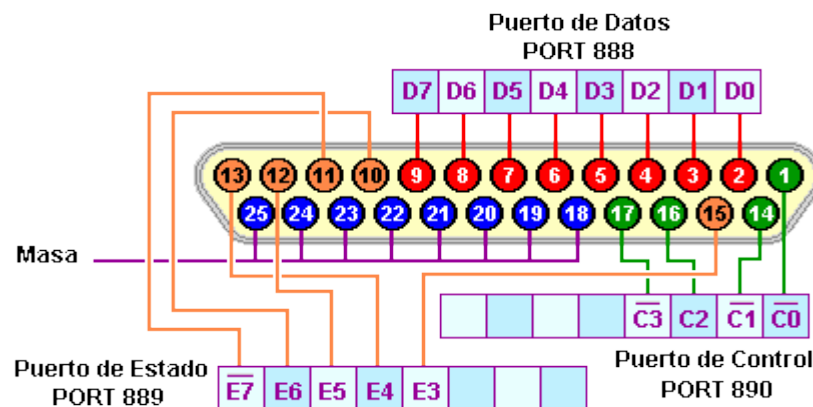


Figura 4.1(b).- configuración del puerto paralelo.

La tensión de trabajo del puerto es de 5 voltios, por lo que necesitamos una fuente estabilizada o regulada de tensión, esto es importante tenerlo en cuenta, ya que estaremos enviando señales al puerto. Por otro lado, si bien puedes utilizar la PC para enviar señales al exterior sin necesidad de una fuente externa, es recomendable utilizarla. Ahora bien, si activas un bit de salida por el puerto, este permanecerá así hasta que lo cambies, es decir que estaremos enviando 5V de forma continua hasta que lo pongamos a 0.



2.5.- Descripción General del PIC16F877

La Familia del PIC16F877

El microcontrolador PIC16F877 de Microchip pertenece a una gran familia de microcontroladores de 8 bits (bus de datos) que tienen las siguientes características generales que los distinguen de otras familias:

- Arquitectura Harvard
- Tecnología RISC
- Tecnología CMOS

Estas características se conjugan para lograr un dispositivo altamente eficiente en el uso de la memoria de datos y programa y por lo tanto en la velocidad de ejecución.

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

Tabla 5.1 bus de instrucciones.

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

- Existen algunas excepciones, como el PIC16C5X que maneja 33 instrucciones de 12 bits (posee empaquetados de 18 y 28 pines y se energiza con 2.5 volts)
- Algunos autores manejan una “*gama enana*” consistente en los PIC12C508 y PIC12C509 en empaque de 8 patitas y con un bus de instrucciones de 12 o de 14 bits.



Variantes principales

Los microcontroladores que produce Microchip cubren una amplio rango de dispositivos cuyas características pueden variar como sigue:

- Empaquetado (desde 8 patitas hasta 68 patitas)
- Tecnología de la memoria incluida (EPROM, ROM, Flash)
- Voltajes de operación (desde 2.5 v. Hasta 6v)
- Frecuencia de operación (Hasta 20 MHz)

Empaquetados

Aunque cada empaquetado tiene variantes, especialmente en lo relativo a las dimensiones del espesor del paquete, en general se pueden encontrar paquetes tipo PDIP (Plastic Dual In Line Package), PLCC (Plastic Leaded Chip Carrier), QFP (Quad Flat Package) y SOIC (Small Outline I.C.) los cuales se muestran en la figura 8:

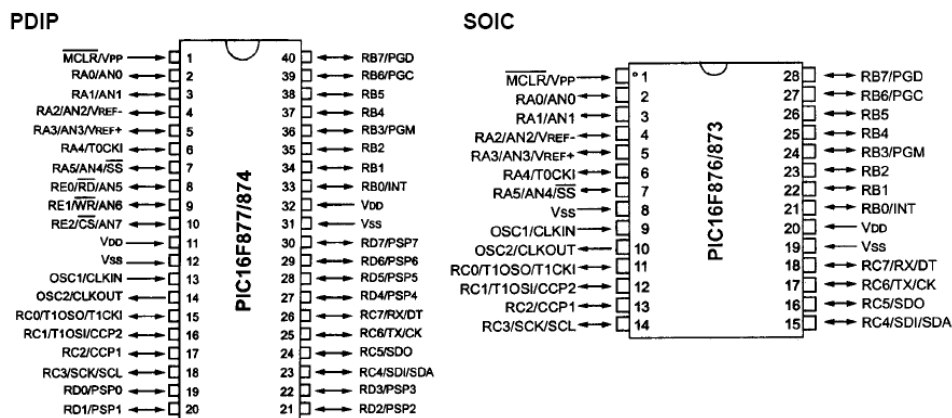


Figura 5.1.- patillaje para familia Pic 16f87XX



La siguiente es una lista de las características que comparte el PIC16F877 con los dispositivos más cercanos de su familia:

Tabla 5.2 características compartidas de microcontrolador pic 16f877

PIC16F873	PIC16F874	PIC16F876	PIC16F877
-----------	-----------	-----------	-----------

CPU:

- Tecnología RISC
- Sólo 35 instrucciones que aprender
- Todas las instrucciones se ejecutan en un ciclo de reloj, excepto los saltos que requieren dos
- Frecuencia de operación de 0 a 20 MHz (200 nseg de ciclo de instrucción)
- Opciones de selección del oscilador

Memoria:

- Hasta 8k x 14 bits de memoria Flash de programa
- Hasta 368 bytes de memoria de datos (RAM)
- Hasta 256 bytes de memoria de datos EEPROM
- Lectura/escritura de la CPU a la memoria flash de programa
- Protección programable de código
- Stack de hardware de 8 niveles



Reset e interrupciones:

- Hasta 14 fuentes de interrupción
- Reset de encendido (POR)
- Timer de encendido (PWRT)
- Timer de arranque del oscilador (OST)
- Sistema de vigilancia Watchdog timer.

Otros:

- Modo SLEEP de bajo consumo de energía
- Programación y depuración serie “In-Circuit” (ICSP) a través de dos patitas
- Rango de voltaje de operación de 2.0 a 5.5 volts
- Alta disipación de corriente de la fuente: 25mA
- Rangos de temperatura: Comercial, Industrial y Extendido
- Bajo consumo de potencia:
 - Menos de 0.6mA a 3V, 4 MHz
 - 20 μ A a 3V, 32 KHz.
 - menos de 1 μ A corriente de standby (modo SLEEP).

Tabla 5.3 de periféricos pic16f877

Periféricos:

Periférico	PIC16F873 PIC16F876	PIC16F874 PIC16F877	Características
3 a 5 Puertos paralelos	PortA,B,C	PortA, B,C,D,E	con líneas digitales programables individualmente
3 Timers	Timer0	Timer0	Contador/Temporizador de 8 bits con pre-escalador de 8 bits
	Timer1	Timer1	Contador/Temporizador de 16 bits con pre-escalador
	Timer2	Timer2	Contador/Temporizador de 8 bits con pre-escalador y post-escalador de 8 bits y registro de periodo
2 módulos CCP	Captura	Captura	16 bits, 1.5 nseg de resolución máxima
	Comparación	Comparación	16 bits, 200 nseg de resolución máxima
	PWM	PWM	10 bits
1 Convertidor A/D	AN0,...,AN4	AN0,...,AN7	de 10 bits, hasta 8 canales
Puertos Serie	SSP	SSP	Puerto Serie Sincrono
	USART/SCI	USART/SCI	Puerto Serie Universal
	ICSP	ICSP	Puerto serie para programación y depuración “in circuit”
Puerto Paralelo Esclavo	PSP	PSP	Puerto de 8 bits con líneas de protocolo



2.5.1 Diagrama de Bloques del PIC16F877

En la figura 9 se muestra a manera de bloques la organización interna del PIC16F877, Se muestra también junto a este diagrama su diagrama de patitas, para tener una visión conjunta del interior y exterior del Chip.

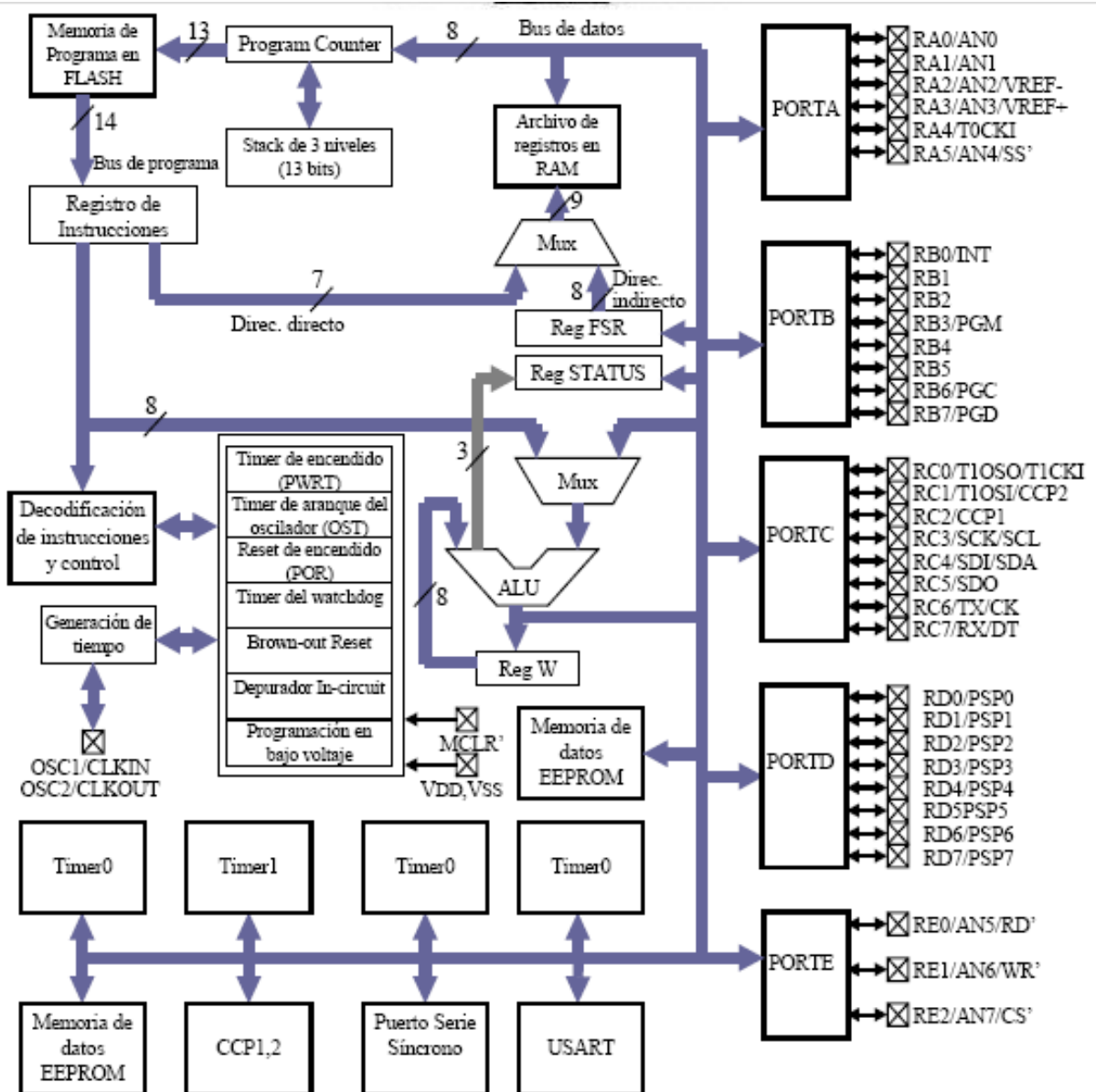


Figura 5.2-diagrama de bloque Pic 16f877



2.5.2 El Archivo de Registros

Aunque el archivo de registros en RAM puede variar de un PIC a otro, la familia del PIC16F87x coincide casi en su totalidad. En la siguiente figura se muestra a detalle el mapa de este archivo de registros y su organización en los cuatro bancos.

Tabla 5.4 Registros

Dirección	registro	Dirección	registro	Dirección	registro	Dirección	registro
00h	INDF(*)	80h	INDF(*)	100h	INDF(*)	180h	INDF(*)
01h	TMR0	81h	OPTION_REG	101h	TMR0	181h	OPTION_REG
02h	PCL	82h	PCL	102h	PCL	182h	PCL
03h	STATUS	83h	STATUS	103h	STATUS	183h	STATUS
04h	FSR	84h	FSR	104h	FSR	184h	FSR
05h	PORTA	85h	TRISA	105h		185h	
06h	PORTB	86h	TRISB	106h	PORTB	186h	TRISB
07h	PORTC	87h	TRISC	107h		187h	
08h	PORTD(1)	88h	TRISD(1)	108h		188h	
09h	PORTE(1)	89h	TRISE(1)	109h		189h	
0Ah	PCLATH	8Ah	PCLATH	10Ah	PCLATH	18Ah	PCLATH
0Bh	INTCON	8Bh	INTCON	10Bh	INTCON	18Bh	INTCON
0Ch	PIR1	8Ch	PIE1	10Ch	EEDATA	18Ch	EECON1
0Dh	PIR2	8Dh	PIE2	10Dh	EEADR	18Dh	EECON2
0Eh	TMR1L	8Eh	PCON	10Eh	EEDATH	18Eh	RESERVADO(2)
0Fh	TMR1H	8Fh		10Fh	EEADRH	18Fh	RESERVADO(2)
10h	T1CON	90h		110h		190h	
11h	TMR2	91h	SSPCON2	111h		191h	
12h	T2CON	92h	PR2	112h		192h	
13h	SSPBUF	93h	SSPADD	113h		193h	
14h	SSPCON	94h	SSPSTAT	114h		194h	
15h	CCPR1L	95h		115h		195h	
16h	CCPR1H	96h		116h	Registros de propósito general (16 bytes)	196h	Registros de propósito general (16 bytes)
17h	CCP1CON	97h		117h		197h	
18h	RCSTA	98h	TXSTA	118h		198h	
19h	TXREG	99h	SPBRG	119h		199h	
1Ah	RCREG	9Ah		11Ah		19Ah	
1Bh	CCPR2L	9Bh		11Bh		19Bh	
1Ch	CCPR2H	9Ch		11Ch	19Ch		
1Dh	CCP2CON	9Dh		11Dh	19Dh		
1Eh	ADRESH	9Eh	ADRESL	11Eh	19Eh		
1Fh	ADCON0	9Fh	ADCON1	11Fh	19Fh		
20h		A0h		120h		1A0h	
	Registros de propósito General (96 bytes)		Registros de propósito General (80 bytes)		Registros de propósito General (80 bytes)		Registros de propósito General (80 bytes)
		EFh				16Fh	
		F0h	Acceso a regs 70h-7Fh		Acceso a regs 70h-7Fh	1F0h	Acceso a regs 70h-7Fh
7Fh		FFh		17Fh		1FFh	



2.6.- DESCRIPCION GENERAL TECLADO PS/2

Un **teclado PS/2** de 104 teclas para PC es un dispositivo razonablemente *inteligente*. Esto quiere decir que él mismo se encarga de la parte hardware del rastreo de teclas, de evitar los insufribles rebotes y de transmitirnos, mediante una trama perfectamente establecida, de los resultados de nuestras manipulaciones sobre sus teclas.

El teclado PS/2 tiene un **mapa de teclas** (figura 10) a las que asigna un **código**, de uno o dos bytes, para cada una de ellas y que son los códigos que nos va a transmitir para indicarnos que se está pulsando una tecla determinada. A estos códigos les vamos a llamar **códigos de rastreo de teclado**. Más abajo se muestra una imagen de un teclado estándar (figura 10 (a), 10(b)) donde se muestran los códigos de rastreo de todas y cada una de las teclas:

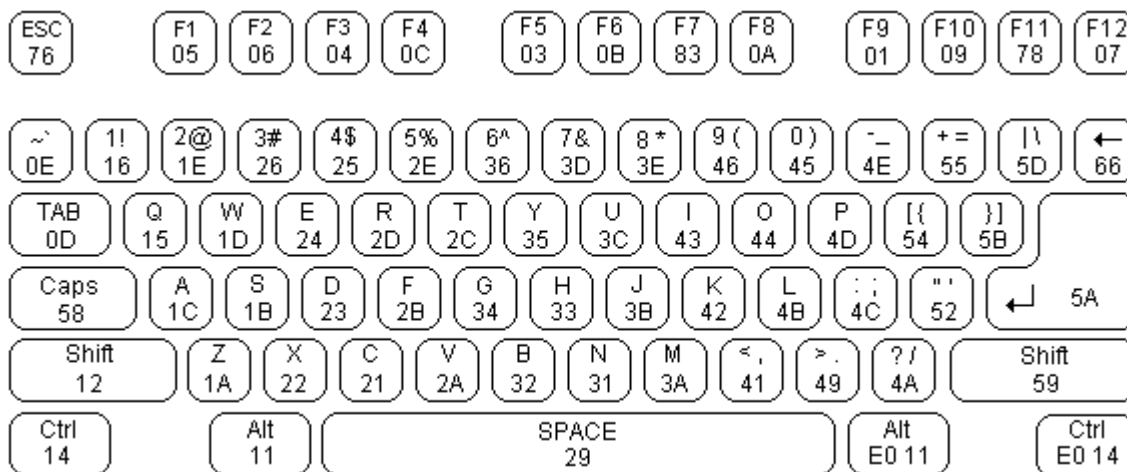


Figura 6.1(a).- Teclado ps/2

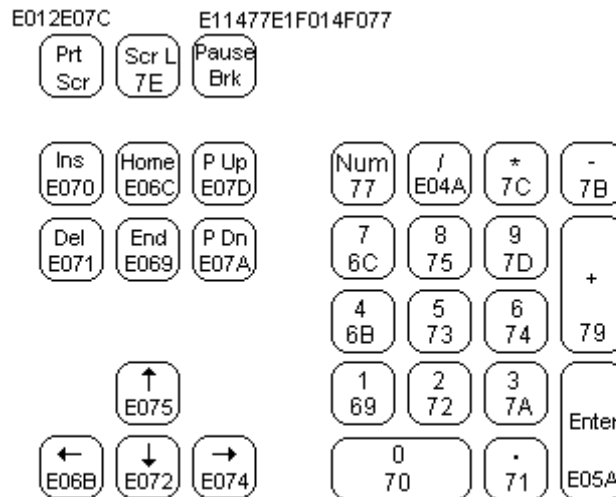


Figura 6.1(b).- Teclado numérico del teclado ps/2

El teclado PS/2 envía el **código de rastreo** asociado a una tecla al ser ésta pulsada, tantas veces como sea necesario si se mantiene pulsada con una cadencia tal como indique su tiempo de repetición, que es programable, y el mismo código de rastreo con el prefijo del byte **F0h** al ser soltada, también conocido como **Break Code**.

Por ejemplo: Para conseguir la letra **G Mayúscula** debemos pulsar la tecla Shift y mientras la mantenemos pulsada, pulsar la tecla G, soltar la tecla G y soltar la tecla Shift. Esa secuencia de pulsaciones nos va a hacer que el teclado transmita la siguiente secuencia de bytes:

Pulsar Shift -> 12h,

Pulsar "G" -> 34h,

Soltar "G" -> F0h 34h

y soltar Shift -> F0h 12h



o escribiendo solo los bytes que vamos a recibir:

1h, 34h, F0h, 34h, F0h, 12h

Nuestro cometido va a ser recibir e interpretar esta secuencia de bytes y actuar en consecuencia.

El teclado PS/2 también admite comandos. Dispone de funciones build-in que podemos disparar mediante el envío del comando correspondiente. Abajo muestro una tabla con los comandos disponibles que podemos enviarle a nuestro teclado. (Nosotros, en este proyecto no vamos a utilizarlos, pero ahí quedan por si alguno de mis amables visitantes desea ampliar la funcionalidad del mismo).

TABLA 6.1 DE COMANDOS DEL TECLADO PS/2

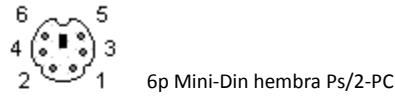
Comando	Valor Hexadecimal
Set/Reset	ED
Echo	EE
Invalid Command	EF
Selec Alternate Scan Codes	F0
Invalid Command	F1
Read ID	F2
Set Typematic Rate/Delay	F3
Eneable	F4
Default disable	F5
Set Default	F6
Set all Keys	
Typematic	F7
Make/break	F8
Make	F9
Typematic/Make/break	FA
Set Key Type:	
Typematic	FB
Make/break	FC
Make	FD
Resend	FE
Reset	FF



2.6.1 Conexión eléctrica:

El teclado PS/2 se conecta mediante cuatro hilos. Dos de ellos son para alimentación **Vcc** a **5V** y **GND**, y otros dos para las señales **Data** y **Clock**. El pinout de los conectores Mini-DIN PS/2 tanto hembra, los del PC, como machos, los del Teclado, es tal como se muestra en la imagen (FIGURA 12)inferior:

Keyboard 6 Pin (PS/2)



Pin	Name	Dir	Description
1	DATA	↔	Key Data
2	n/c		Not connected
3	GND	—	Gnd
4	VCC	→	Power , +5 VDC
5	CLK	→	Clock
6	n/c		Not connected

Figura 6.2.- configuración de pines del teclado (ps/2)



En el Teclado PS/2 las señales Data y Clock con de "**colector abierto**". Esto quiere decir que para establecer una comunicación eléctricamente correcta debemos nosotros suministrar voltaje para el nivel lógico alto, y es él el encargado de dar los correspondientes niveles lógicos bajos, tirando nuestra señal a **GND** cuando así sea necesario. Este tema se soluciona conectando dos resistencias de 10K entre dichas líneas y Vcc por lo que siempre tendrán nivel lógico alto, salvo cuando el teclado disponga lo contrario y lo tire a GND para dar lo correspondientes niveles lógicos bajos. Esto es lo que se llama conectar unas resistencias **Pull-Up**. El esquema de más abajo (FIGURA 13) muestra como conectar las resistencias Pull-Up entre el teclado y el PIC:

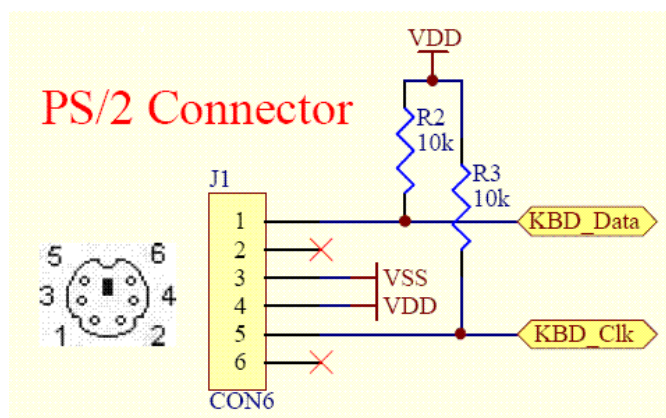


Figura 6.3.- conexión del Ps/" hacia el Pic, se ilustran las resistencias Pull-Up

Nota: Generalmente nuestros PIC's y en concreto el utilizado para este proyecto, el PIC **16F877A**, tiene todo el puerto B con la capacidad de conectar, mediante la correspondiente configuración, una batería de **resistencias Pull-Up a todos sus pines**. Esto hace innecesario la conexión de las resistencias externas tal como se presenta en el esquema anterior, pero se indico porque podría tener que utilizar cualquier otro puerto que no disponga de esta característica, y en ese caso se debería conectarlas para su correcto funcionamiento.



2.6.2 Protocolo de comunicación PS/2:

El teclado PS/2 se comunica mediante un **Protocolo Serie Síncrono**. Utiliza, por lo tanto, una señal de **Clock** que indica cuando están disponibles los correspondientes bits en la señal de **Data**.

En reposo la señal de Clock está a nivel alto; a cada pulso a nivel bajo corresponde un pulso a nivel alto o bajo en la señal de Data, que se traducen respectivamente como bits 0 ó 1 del dato a transmitir.

La **trama** completa se compone de **11 bits**. Siendo el primero un bit de **Start**, a continuación los 8 bits del **Dato** a transmitir enviándose primero el LSB (ó bit menos significativo), el décimo es el de **paridad** (usa la Impar, u Odd en Inglés) y por último un bit de ACK o **Stop**.

Abajo (FIGURA 14) puede verse un cronograma de esta trama de comunicación PS/2 Teclado (Keyboard) -> PC (host):

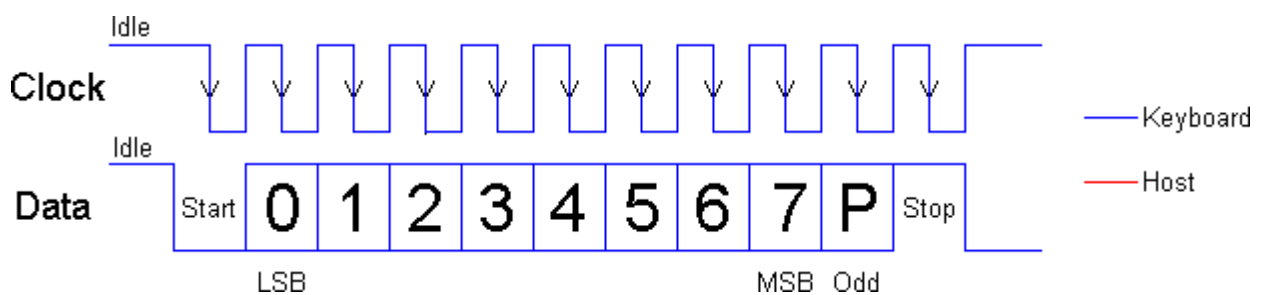


Figura 6.4.- cronograma de la trama de comunicación Ps/2.



Este protocolo de comunicación es **bidireccional**. El teclado PS/2 admite también comandos enviados desde el PC con el mismo formato que estamos estudiando, tal como adelantamos en el primer apartado de este proyecto. Y tal como decíamos allí, nosotros no vamos a implementar el envío de comandos al teclado, pero creo conveniente discutir también su procedimiento de envío, a nivel de protocolo por si alguien desea implementarlos en su propio trabajo.

Para poder habilitar la comunicación inversa, del PC (host) al Teclado PS/2 (Keyboard) debemos primero indicárselo así al Teclado mediante la señal de Clock. Para ello debemos poner a nivel bajo el Clock durante unos 160 uS, y la señal de Data a bajo unos 35 uS después de haber lanzado la del Clock. A partir de ahí debemos esperar la señal del Clock generada por el Teclado. Esto nos indicará que el teclado está dispuesto para recibir nuestro comando. Detectamos dicha señal como primer pulso de Clock, y a partir del siguiente podemos comenzar a enviarle nuestro byte.

Le enviaremos entonces los ocho bits de nuestro comando, cada uno de ellos cuando el correspondiente pulso en bajo del Clock del teclado así nos lo indique, empezando por el LSB, a continuación el bit de paridad impar (El numero de unos en los datos más el de paridad deber ser impar o sea 1 si el numero de unos es par y cero si el total de unos es impar)



Y entonces debemos esperar el ACK del teclado (FIGURA 15), que debe venir tras dos pulsos de reloj, en nivel bajo, indicándonos de este modo que el teclado ha recibido correctamente nuestro comando. en caso contrario debemos volver a repetir nuestra secuencia de envío.

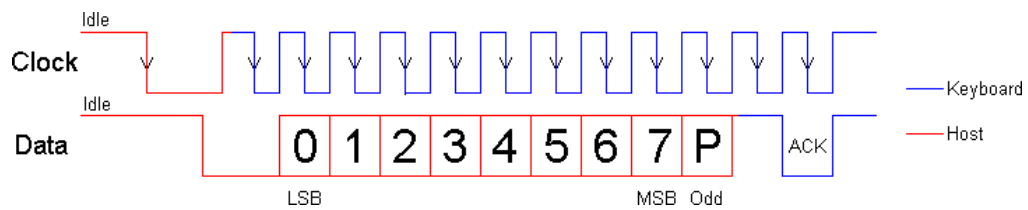


Figura 6.5.- Repetición de secuencia de envío



Capítulo 3.- DESARROLLO DEL PROYECTO

Descripción cronológica de actividades.

La duración aproximada de investigación fue de dos meses iniciando en 14 de julio hasta el día 2 septiembre.

Durante el proceso de investigación se logro identificar los diferentes tipos de fonemas que existen, así como también se planeo que componentes se utilizarian para la construcción del sistema de reproducción fonético.

En forma general se puntualizara lo que se necesita:

Una librería de fonemas, librería que se creó en un programa de grabación y edición de audio de nombre audio galaxy este programa versátil propicio un ambiente cómodo de trabajo y rapidez debido a que se pudo de manera práctica cambiar la frecuencia de las grabaciones que hicimos, y también poner a 100us todos los fonemas para que fueran del mismo tamaño.

Un componente de almacenamiento para la librería de fonemas, para esto se utiliza una memoria isd2560 que puede almacenar 60seg. de sonido guardados analógicamente en direcciones especificas, por esto y para poder controlar el tiempo y las direcciones desarrollamos una interfaz PC-isd la cual se expone más adelante.



Elegir una pantalla de LCD para visualizar los fonemas esta pantalla es la 16-4b en su hoja de datos podemos ver todas sus características, nosotros la elegimos simplemente por su tamaño y precio en el mercado.

Un teclado, se tuvo la brillante idea de construir uno pero el Pic solo detectaba el teclado matricial de 4x4, aunque se logro hacer uno de 8x4 los costos eran altos porque presentaba demasiados rebotes debido a esto se prefirió utilizar un teclado de PC ps/2 el cual se configuro con MikroC ya que contaba con toda la librería.

Utilizamos para el proyecto una fuente Vcc de 5volts.

Todo este tiempo planearon el camino para los siguientes meses en que fue realizado el proyecto figura 16.

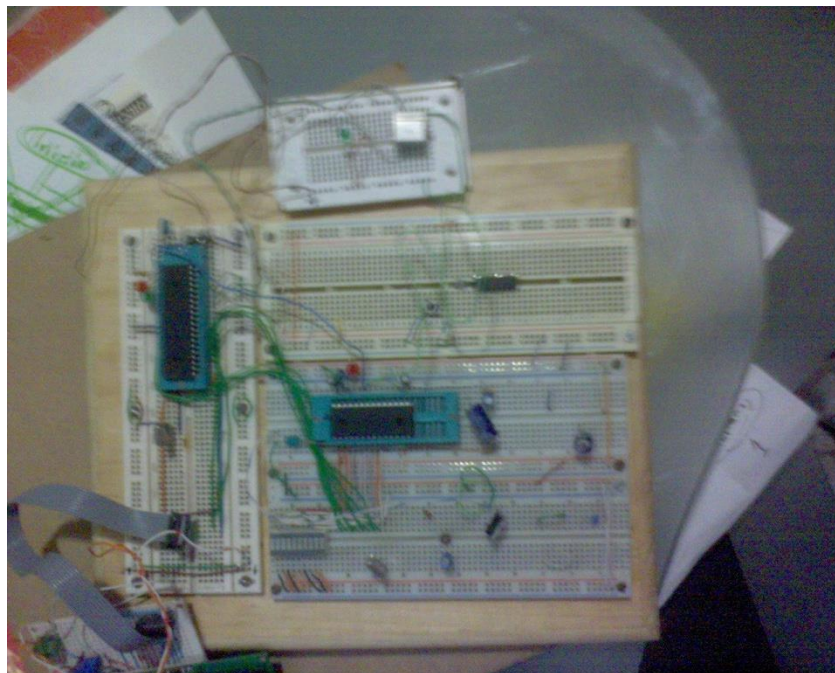


Figura 7.1.- proyecto terminado en el protoboard



3.1 Diseño de diagramas y simulación.

La duración de este punto fue de un mes aproximadamente partiendo el día 4 de septiembre hasta 13 de octubre podría decirse pero siempre que se pudo mejorar algo se regreso a la simulación y al rediseño.

Para poder explicar esta parte del proyecto lo dividiremos en cinco partes que son

1. Fuentes de voltaje
2. Forma de: Grabación / Reproducción de fonemas
3. Diseño de Interfaz PC-isd y su software
4. Conexión comunicación isd- Pic
5. Conexión comunicación Pic- Teclado Ps/2

Fuentes de voltaje

La fuente de voltaje es la que se va a encargar de suministrar todo el voltaje y la corriente necesaria para el funcionamiento estable de el reproductor de fonemas, de la misma forma servirá para alimentar a todos los elementos de entrada y salida como son la pantalla LCD, el teclado Ps/2 al cerebro del Microcontrolador.

En una fuente de voltaje, el voltaje va disminuyendo mientras en medida de la corriente que necesita.

La fuente de voltaje que se construyo de 5Vcc. Cuenta con un transformador de 110vca, a 30 Vca Y 3 amperes junto con dos reguladores de voltaje el 7815 que convierte los 30 vcc en 15 vcc este mismo conectado aun 7805 que convierte los 15vcc a 5vcc con un nivel de corriente de 1 ampere. Se observo fuera una fuente muy dinámica y resistente



realizamos distintas pruebas hasta que resulto una fuente bastante estable; este es el diagrama (figura 17) y la simulación.

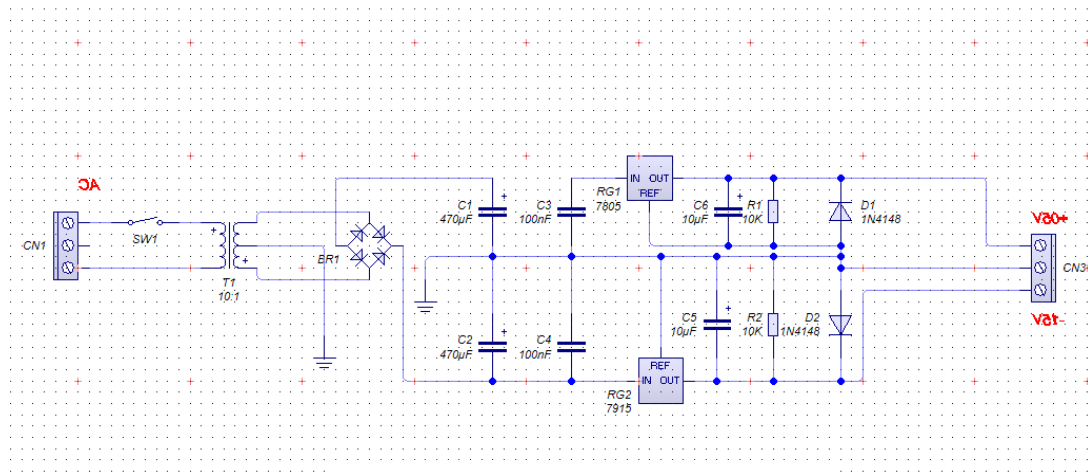


Figura 7.2.- diagrama para fuente simetrica +5 -5 Vcc.

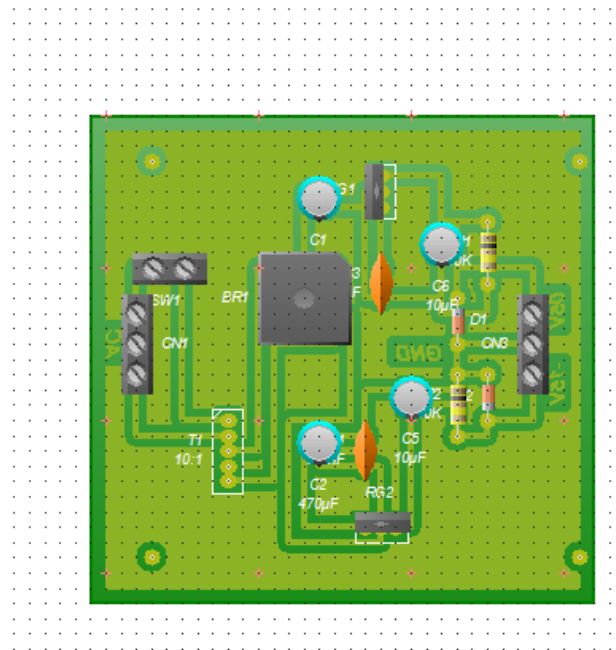


Figura 7.3.- diseño de placa terminado en 3D.



Forma de: Grabación / Reproducción fonemas

Para poder grabar y reproducir los fonemas se utiliza el siguiente diagrama (Figura 19), aunque grabar de esta forma atrajo demasiado ruido y los fonemas variaban demasiado su tamaño.

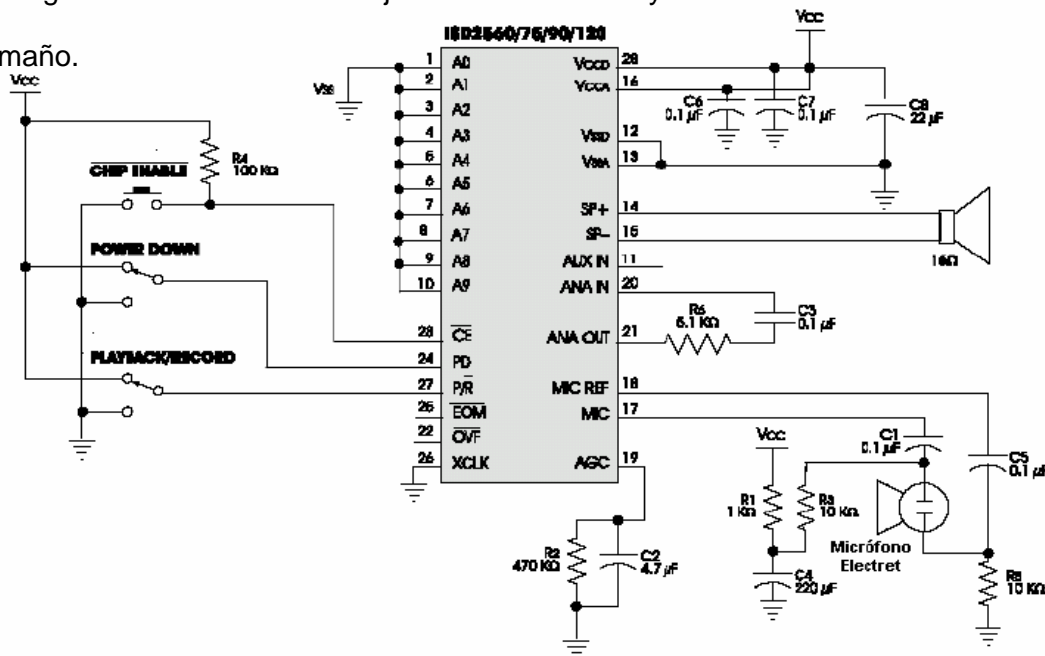


Figura 7.4.- Modo básico para grabación reproducción de fonemas.

Este es el circuito básico para grabación y reproducción de mensajes se le conoce como “modo básico de direccionamiento” ya puede utilizar un direccionamiento de memoria.

Para activar la memoria isd en este modo al igual que las entradas de A0 – A9 necesitamos poner los dos bits más significativos en este caso A8-A9 en estado bajo para que las entradas sean identificadas como de direccionamiento.



Modo grabación

- 1) Colocar el pin PD (Power Down) en estado “BAJO”. Normalmente se utiliza un pull-down
- 2) El pin P/R (play/record) debe estar en estado BAJO.
- 3) Asignar la dirección de memoria en los pines A9-A0.
- 4) El pin CE (chip-enable) es conmutado BAJO durante toda la grabación de un determinado mensaje. EOM (End of message) va a estado ALTO, indicando que la operación está en proceso.
- 5) Una marca EOM es insertada al final del mensaje, que indica la terminación del mismo.

Modo reproducción

- 1) Colocar el pin PD en estado BAJO
- 2) El pin P/R debe estar en ALTO
- 3) Asignar la dirección en memoria de un determinado mensaje
- 4) El pin CE es conmutado BAJO y la reproducción dura hasta encontrar una marca EOM.

NOTA: para la reproducción de un mensaje se requiere un pulso de CE, de 10ms de duración como mínimo.



Diseño de Interfaz PC-isd y su software.

Para poder grabar los fonemas dentro de la isd se diseño una interfaz, esto requirió conocimientos básicos de Borland C++ builder para crear el software. En el diseño del hardware de esta tarjeta se hizo por el puerto paralelo debido a lo práctico de sus salidas y la facilidad de conexión con las direcciones de entrada en el isd. Aquí se presenta el diseño del diagrama (figura 20).

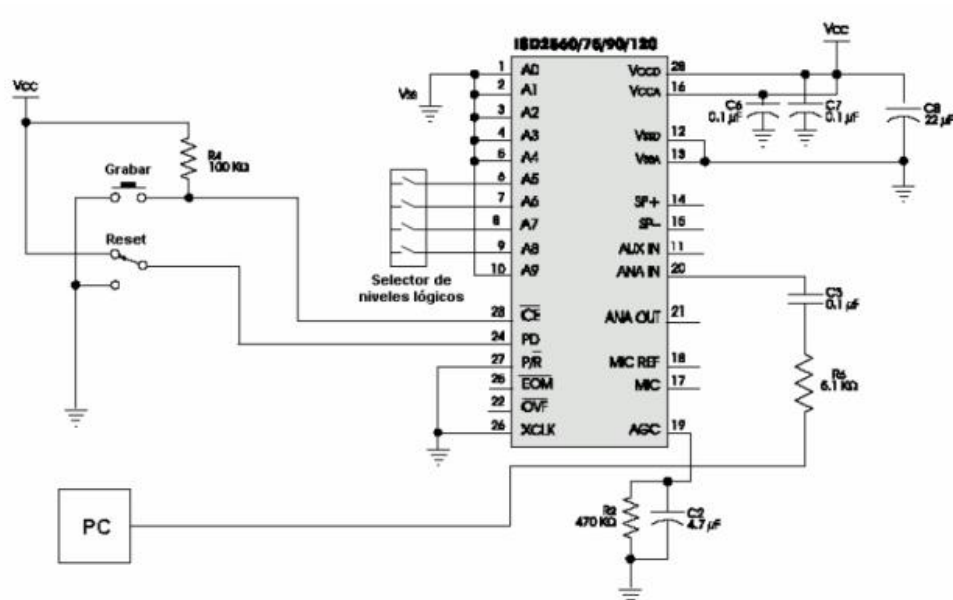


Figura 7.5.- diseño de interfaz Pc/Isd.

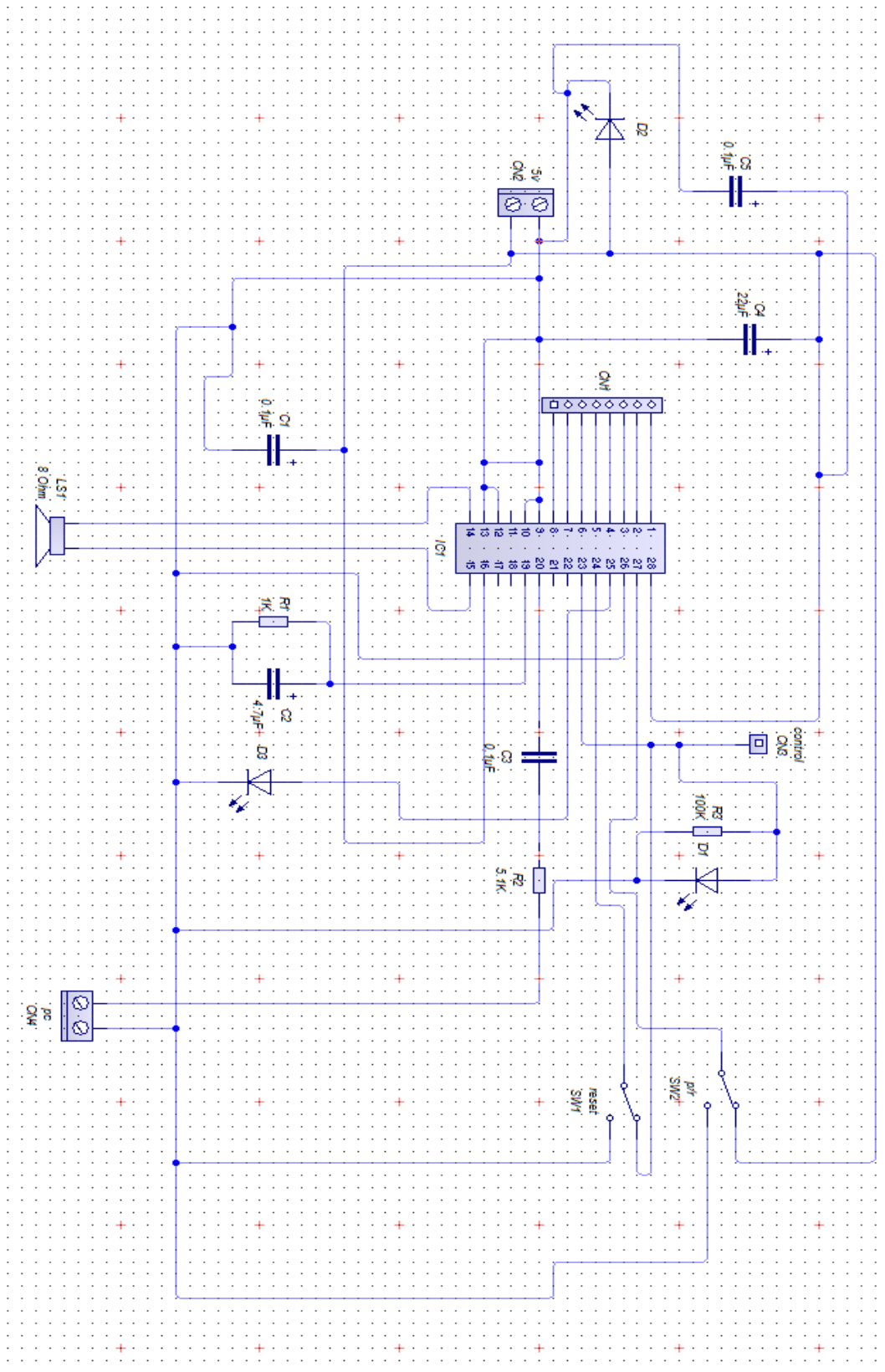


Figura 7.5.- diseño de interfaz Pc/Isd. Con control del CE.



Aquí presentamos una tabla de direcciones en que fueron grabados los fonemas dentro de la isd con el software.

7.1.-Tablas de direcciones

Tabla de VOCALES DIRECCIONES isd

A	1
E	4
I	6
O	8
U	10

(a)

Tabla CONSONANTES DIRECCIONES isd

B	14
CH	16
D	18
F	20
G	22
H	24
J con cola	25
K	27
L	29
LL / Y	31
M	33
N	35
P	37
R	39
RR	41
S	43
T	45
X	47
X como jota	49
Y / LL	51

(b)



Esta es una imagen (figura 21) del software de interfaz se anexa también todo el programa en borland c++ builder. Para grabar un fonema desde la interfaz lo primero que deberán hacer es ingresar una dirección después se da un click en abrir y grabar y el fonema será grabado en la isd, el funcionamiento de este pequeño software y el tiempo de grabación se basa en el tamaño del fonema esto debido a que en cuanto empieza la reproducción se manda a estado bajo el pin CE del ISD y en cuanto termina se pone en estado alto ingresando así una señal EOM justo al término del fonema.

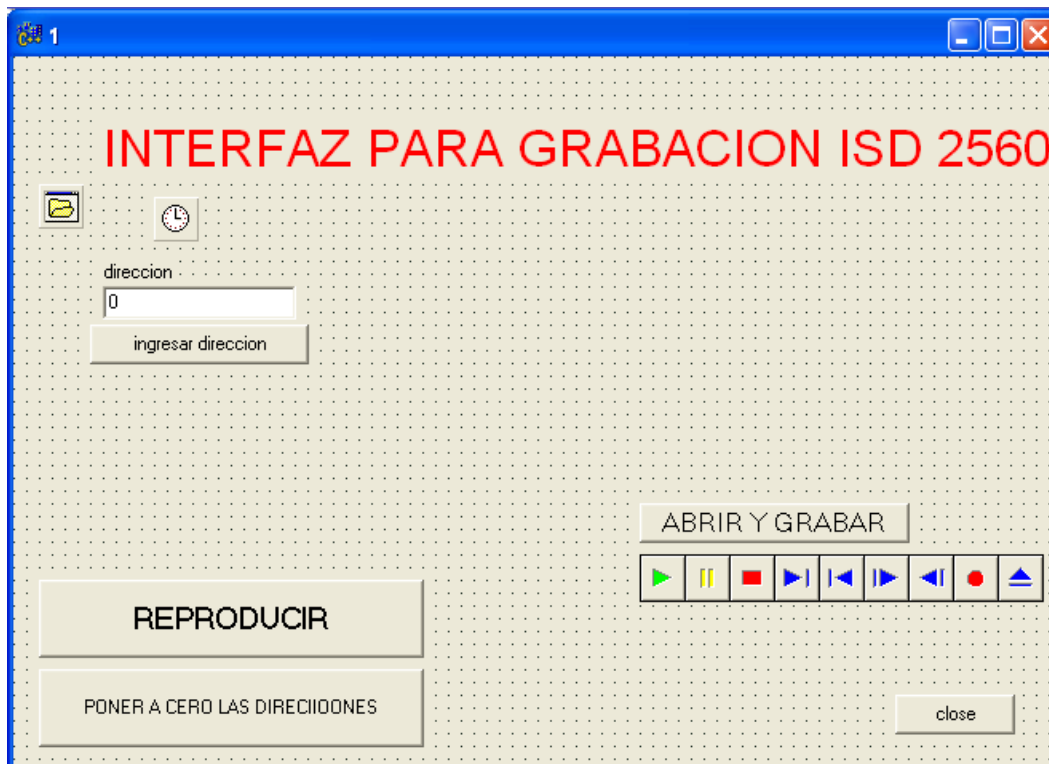


Figura 7.6.- diseño de interfaz Pc/Isd. Con control por software



Conexión comunicación isd Puerto Paralelo

Las pruebas para conectar el isd con el puerto paralelo se realiza en protoboard de esta manera se diseña el cable de conexión al igual el diagrama de conexión, la conexión que escogida fue muy sencilla dado que el puerto paralelo consta de 8 bits al igual que el isd. Se Asigna la entrada A0 del isd con la salida D0 del puerto paralelo, en seguida A1 con D1 y así consecutivamente hasta A7 con D7 como se puede observar en la Figura 21(a).

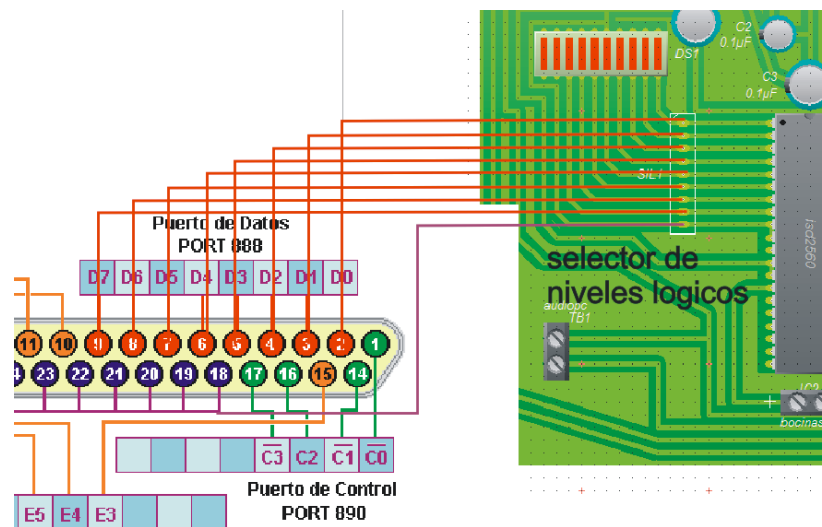


Figura 7.6(a).- conexión virtual de isd

Conexión comunicación Pic- Teclado ps/2.

Para la conexión del teclado se utiliza la librería de MikroC aunque no traía diagramas de conexión durante la investigación se muestra el modo de conectar y aquí también están las imágenes (Figura 22, Figura 23, Figura 24) en el protoboard.

para que el teclado funcione es necesario que el oscilador sea de 8.000 MHz como mínimo.

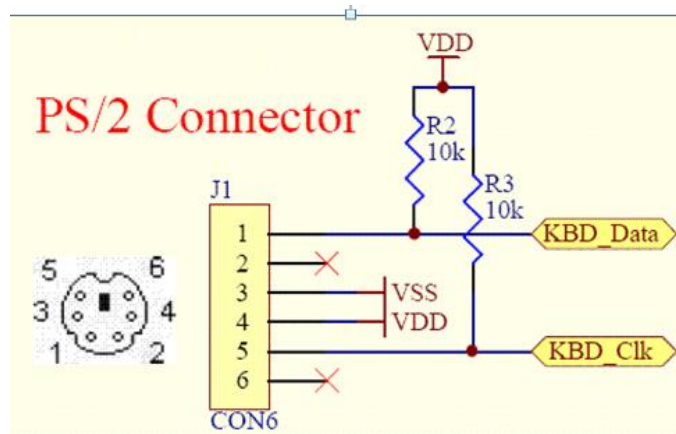


Figura 7.7.- Conexión ps/2 al pic enfocando sistema de resistencias pull up



Figura 7.9.- Teclado Ps/2

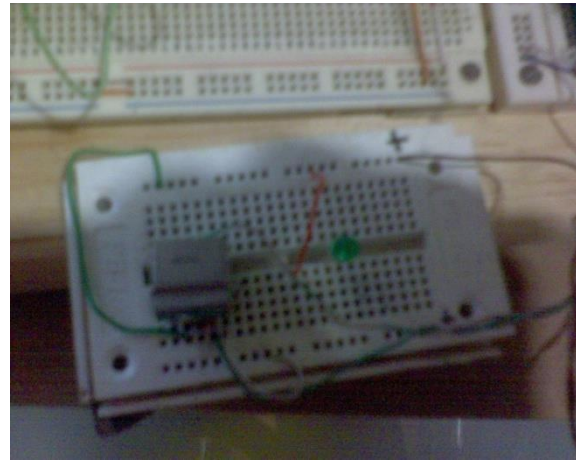


Figura 7.8.- Conexión Ps/2 dentro del protoboard



3.2 Pruebas REALES, Placas de Circuitos, Ensamble y Correcciones

Esta etapa duro del 20 de octubre al 1 de diciembre contemplando casi dos meses. Aquí es donde se presentan los resultados de los cinco puntos tocados anteriormente.

Fuentes de voltaje

La fuente fue construida en una protoboard le realizaron pruebas de estabilidad sin embargo el protoboard introducía mucho ruido así que se construyo sobre una placa de baquelita después de terminar de soldar todos sus componentes se le recubrió con un barniz aislante para reducir el ruido, el resultado es el siguiente (Figura 25).



Figura 7.10.- Fuente de Voltaje

Los voltajes obtenidos son de 5 Vcc en la salida con una corriente de 1 ampere.



Forma de: Grabación / Reproducción de fonemas

El circuito de grabación en modo de direcciones fue construido sobre una protoboard, para realizar pruebas pertinentes en lo que concierne a grabación y reproducción para su sorpresa el resultado fue ser una memoria bastante práctica y robusta como pensábamos pues no necesito ningún amplificador para que fuera audible y con buen sonido, sin embargo como este circuito solo sirvió de muestra para ver el funcionamiento de la memoria se tuvo que conectar el micrófono y todos sus componentes tal cual el diagrama que se presenta en la sección anterior, este circuito fue desmantelado al hacerle las modificaciones necesarias para la interfaz como podrán observar más adelante.



Terminación de Interfaz isd. – PC

Se Logra Finalizar en un 100% la interfaz, esta es una imagen virtual (Figura 26) del hardware .

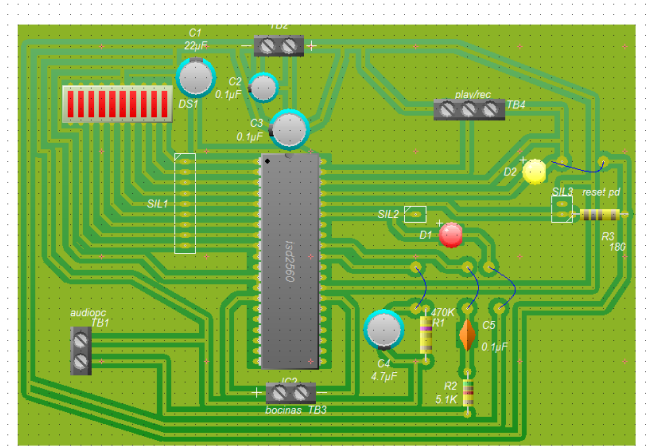


Figura 7.11.- Interfaz ISD-PC, imagen virtual.

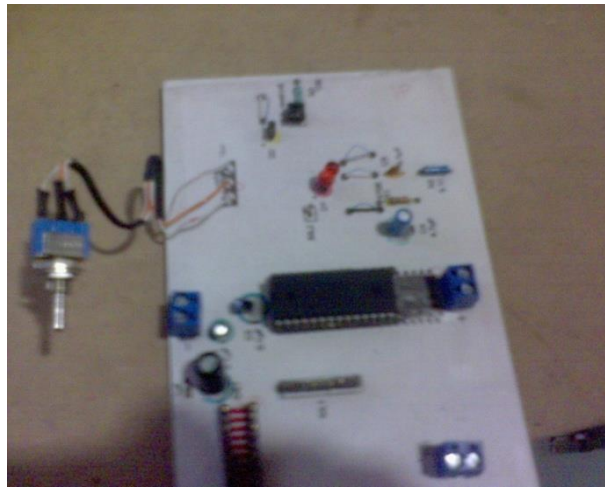


Figura 7.12.- Interfaz ISD-PC, imagen Real del Cto.

Como se puede observar en la imagen se agrega una barra de led's (Figura 27) que indican la dirección en código binario del isd esta interfaz permite tanto grabar el fonema, como reproducirlo. Las pruebas revelaron que con esta interfaz el tiempo de grabación es el mismo, siempre y cuando se haya realizado la librería de fonemas al mismo tiempo y frecuencia.



Conexión Comunicación Pic – isd

En esta etapa se completo al 70% debido a que supera las pruebas de conexión y función realizando completamente el trabajo.

pero no se logra completar el programa que pudiera enlazar el teclado con el Pic al igual que no se pudo emplacar, debido a que se agoto el tiempo para la entrega de este, aun así se logra reproducir algunas palabras aquí pongo la imagen de conexión al Pic (Figura28, Figura 29) al igual que su forma real en el protoboard. (se anexa el programa del Pic)

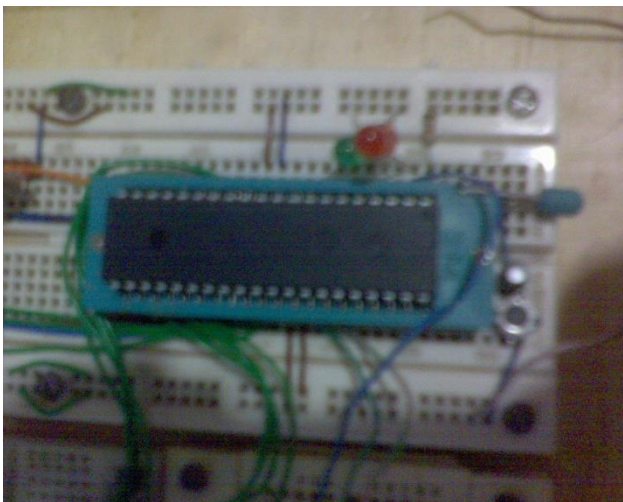


Figura 7.13.- conexión del PIC-ISD 1.

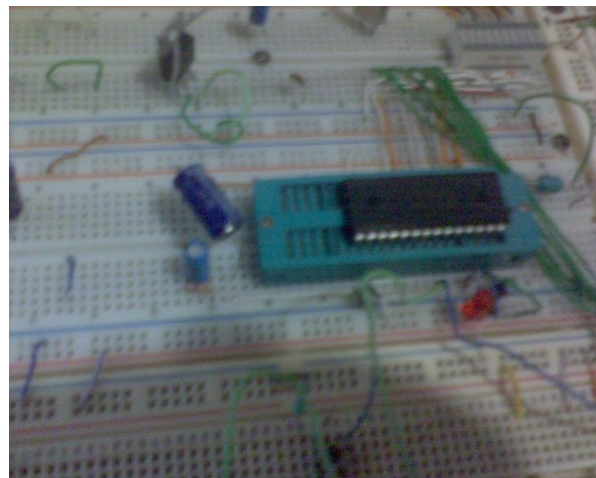


Figura 7.14.- conexión PIC-ISD 2



Diseño Conexión comunicacion Pic-Teclado Ps/2

Esta etapa se completo al 80% se logra establecer completamente la conexión PIC-Teclado/Ps2-LCD también por falta de tiempo no se logra completar el programa para que comparara los datos introducidos al teclado y los reflejara en las direcciones de la isd. De esta parte del proyecto no se hicieron simulaciones sino simplemente pruebas reales en el protoboard pero el esquema es el mismo que se muestra en la imagen anterior, aquí se muestran unas imagenes reales (Figura30, Figura 31) del protoboard y su conexión.



Figura 7.15.- conexión PIC-LCD a 4 bits.

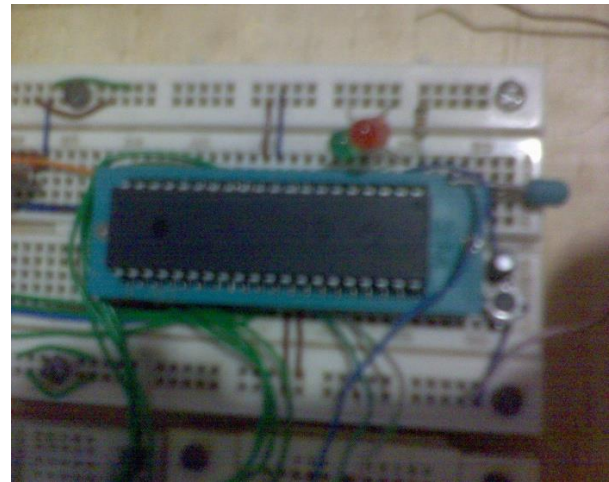


Figura 7.16.- conexión PIC-LCD-TecladoPs/2

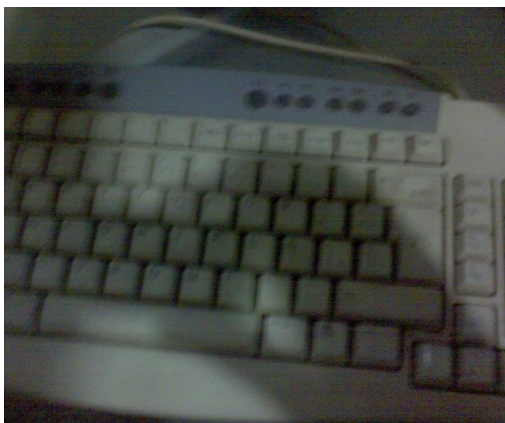


Figura 7.17.- Teclado Ps/2

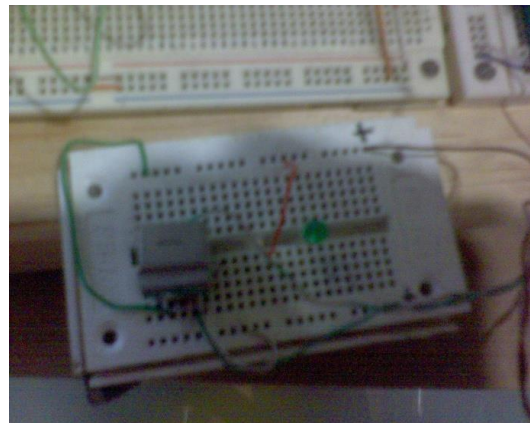


Figura 7.18.- Ps/2



Conclusiones y sugerencias.

En conclusión el proyecto no pudo ser terminado por falta de tiempo y no de conocimientos ya que estos en el transcurso del proyecto siempre logramos obtenerlos, para seguir dando pasos hacia adelante.

Podemos puntualizar algunas cosas como son:

Que el funcionamiento de la memoria isd y todo el proyecto en si no serviría de nada si no logramos obtener una librería de fonemas fidedigna (nos referimos a una serie de grabación de fonemas que cuente con una claridad de audio excelente).

Para que obtengan una librería Fidedigna es necesario controlar su tiempo de reproducción logrando sea el mismo tiempo en todos los archivos de audio, controlar que su frecuencia sea la misma nosotros recomendaríamos de 48 Khz.

Podran observar que al grabar los fonemas se puede lograr una mejor calidad de reproducción grabación bajo todas las condiciones descritas que el descargarlo de la red..

Debe existir un tamaño minimo del fonema para que pueda ser audible este tamaño puede ser aproximadamente 100ms mínimo.

También tendremos que ajustar la fuente a más de un ampere debido a que la fuente que diseñamos aunque contaba con la corriente suficiente los reguladores que utilice solo soportaban 1 ampere debido a esto la fuente se calentaba en un periodo muy corto.



En cuanto a la conexión del teclado aunque fue un éxito pensamos que quizá sería mejor buscar la forma de adaptarle un teclado de USB aunque quizá esto signifique ocupar gran parte de la memoria del Pic al tener que construir todo el protocolo de comunicación.

Otro punto que nos costó bastante construir fue la respuesta de las direcciones en que grabamos y reproducíamos puesto que cada espacio de la memoria isd es con incrementos pequeños por esto es bueno dejar suficiente espacio entre una grabación y otra al igual que definir bien el pulso del CE debido a que si lo oprimimos demasiado tiempo este reproduce todo lo que hay en la memoria isd hasta causar un desbordamiento en la memoria.

Los conocimientos adquiridos en la realización de este proyecto me han llevado a pensar la forma en que se podría mejorar la idea raíz que tuvimos implementado un segundo isd que funcione desfasado con el primero para poder así crear un buffer y un efecto conocido como de titubeo en fonética obteniendo así que la voz sea un poco más suave y no tan cuadrada.



Recomendaciones y trabajos futuros.

Que el funcionamiento de la memoria isd y todo el proyecto en si no serviría de nada si no logramos obtener una librería de fonemas fidedigna (nos referimos a una serie de grabación de fonemas que cuente con una claridad de audio excelente).

Tendran que ajustar la fuente a más de un ampere debido a que la fuente que diseñamos aunque contaba con la corriente suficiente los reguladores que utilice solo soportaban 1 ampere debido a esto la fuente se calentaba en un periodo muy corto

Buscar la forma de adaptarle un teclado de USB aunque quizá esto signifique ocupar gran parte de la memoria del Pic al tener que construir todo el protocolo de comunicación.

Los conocimientos adquiridos en la realización de este proyecto han llevado a pensar la forma en que se podría mejorar la idea raíz que tuvimos implementado un segundo isd que funcione desfasado con el primero para poder así crear un buffer y un efecto conocido como de titubeo en fonética obteniendo así que la voz sea un poco más suave y no tan cuadrada.



Referencias

Libros.

[**ISD, 1996**]: "Voice Record and Playback ICs". ISD Data Book, 1996.

[**Kelly-Bootle, 1989**]: S. Kelly. Bootle B. Fowler, "68000/68010/68020 Arquitectura y programación en ensamblador" Madrid: Anaya Multimedia, 1989.

[**Mata, 1999**]: N.G. Mata, Filtros de Onda Acústica Superficial "Saw". Universidad Tecnológica Nacional – F.R. Bahía Blanca – Departamento Electrónica – Cátedra de Electrónica Aplicada III.

[**Mikro-e, 2008**]: "tutorial de MikroC"

Artículos.

[**Sol, 2007**]: Álvaro Hdez. Sol "Curso de Micro controladores Pic 16f877A" Instituto Tecnológico de Tuxtla Gutiérrez – Tuxtla Gutiérrez- Departamento de electrónica – Cátedra de Micro controladores.

[**Romano, 1984**]: Jorge Romano "un sistema automático de síntesis del habla mediante semisilabas" Múnich- universidad técnica de Múnich Cátedra de Procesos de Datos.

[**Barbosa, A.**] Desarrollo de una nueva voz en el Español de México para el sistema de texto a voz Festival. Tesis de Maestría, Universidad de las Américas, México Otoño 1997.

[**Beutnagel, M. & Conkie, A.**]: Interaction of units in a Unit Selection Database. NJ, USA: AT&T Labs - Research, 1999.

[**Flores, L.**]: Síntesis de Voz con Unit Selection. Tesis de Licenciatura, Universidad de las Américas, México Primavera 2001.

[**Hunt, A. & Black, A.**]: Unit Selection in a concatenative speech synthesis system using a large speech database. In Proceedings of ICASSP-96, Atlanta, GA, 1996.

[**Lester, J.C; Converse, S.A.; Stone, B.A., Kahler, S.E. y Barlow, S.T.**]: "Animated pedagogical agents and problem solving effectiveness: A large scale empirical evaluation". In *Proceedings of the Eighth World Conference on Artificial Intelligence in Education*, pp. 23-30, IOS Press, 1997b.

[**Lester, J.C.; Voerman, J.R.; Towns, S.G. y Callaway, C.B.**]: "Deictic believability: Coordinating gesture, locomotion and speech in life-like pedagogical agents". *Applied Artificial Intelligence*, 13: 383-414, 1999.

[**Lester, J.C.; Stone, B.A. y Stelling, G.D.**]: "Life-like pedagogical agents for mixed-initiative problem solving in constructivist learning environments". *User Modeling and User-Adapted Interaction*, 9: 1-44, 1999.

[**Shaw, E.; Ganeshan, R.; Johnston, W.L. y Millar, D.**]: "Building a case for agent-assisted learning as a catalyst for curriculum reform in medical education". In *Proceedings of the Ninth World Conference on Artificial Intelligence in Education*, IOS Press, 1999.

[**Meza, H.**]: Modelos estadísticos de Duraciones de los Fonemas en el Español Mexicano. Tesis de Licenciatura, Universidad de las Américas, Puebla, México Primavera 1999.

Página Internet.

<http://www.isd.com>

<http://www.uiowa.edu/~acadtech/phonetics/spanish/frameset.html>

<http://www.superrobotica.com/S310230.htm>

<http://electrotips.wordpress.com/2007/11/21/isd1420seq/>

http://catarina.udlap.mx/u_dl_a/tales/documentos/lis/guzman_a_ma/capitulo5.pdf

<http://www.todoarquitectura.com/>



ANEXOS A esquemáticos

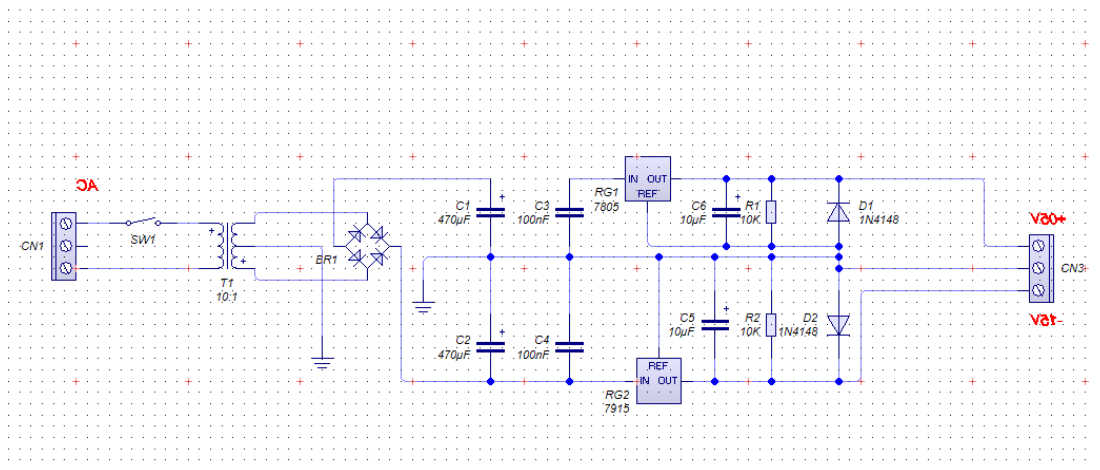
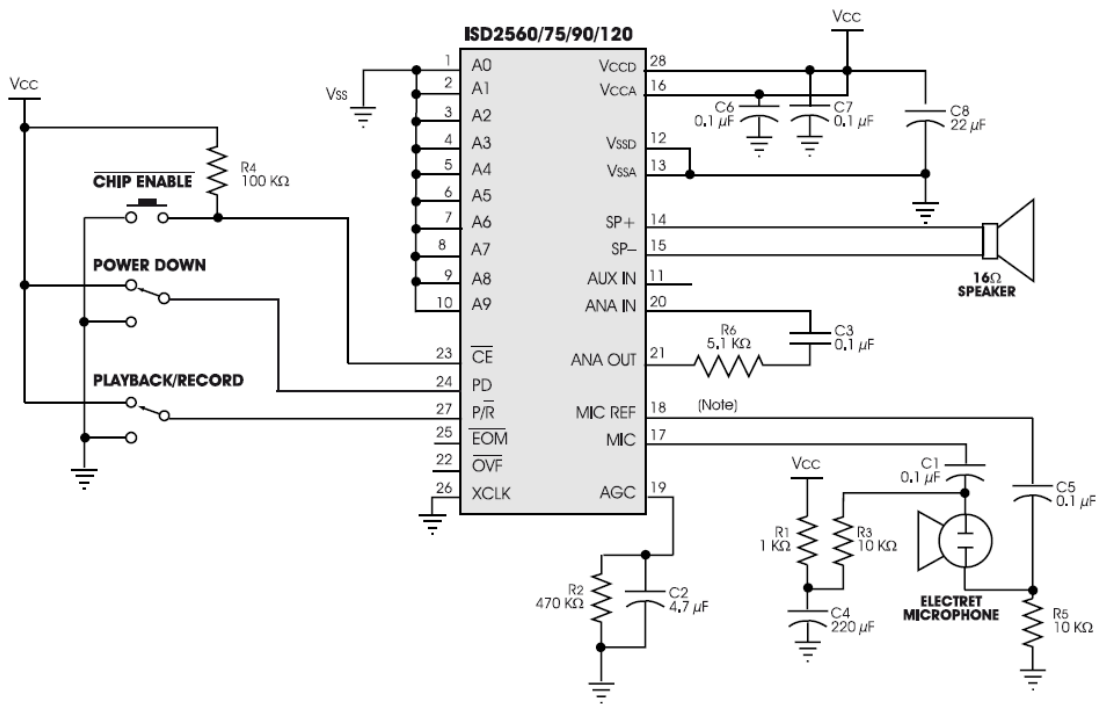
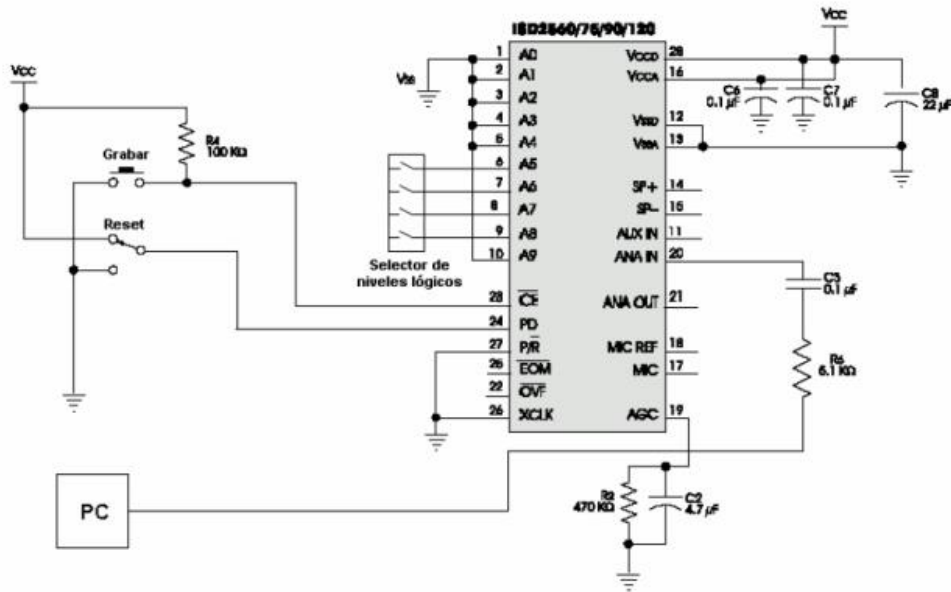
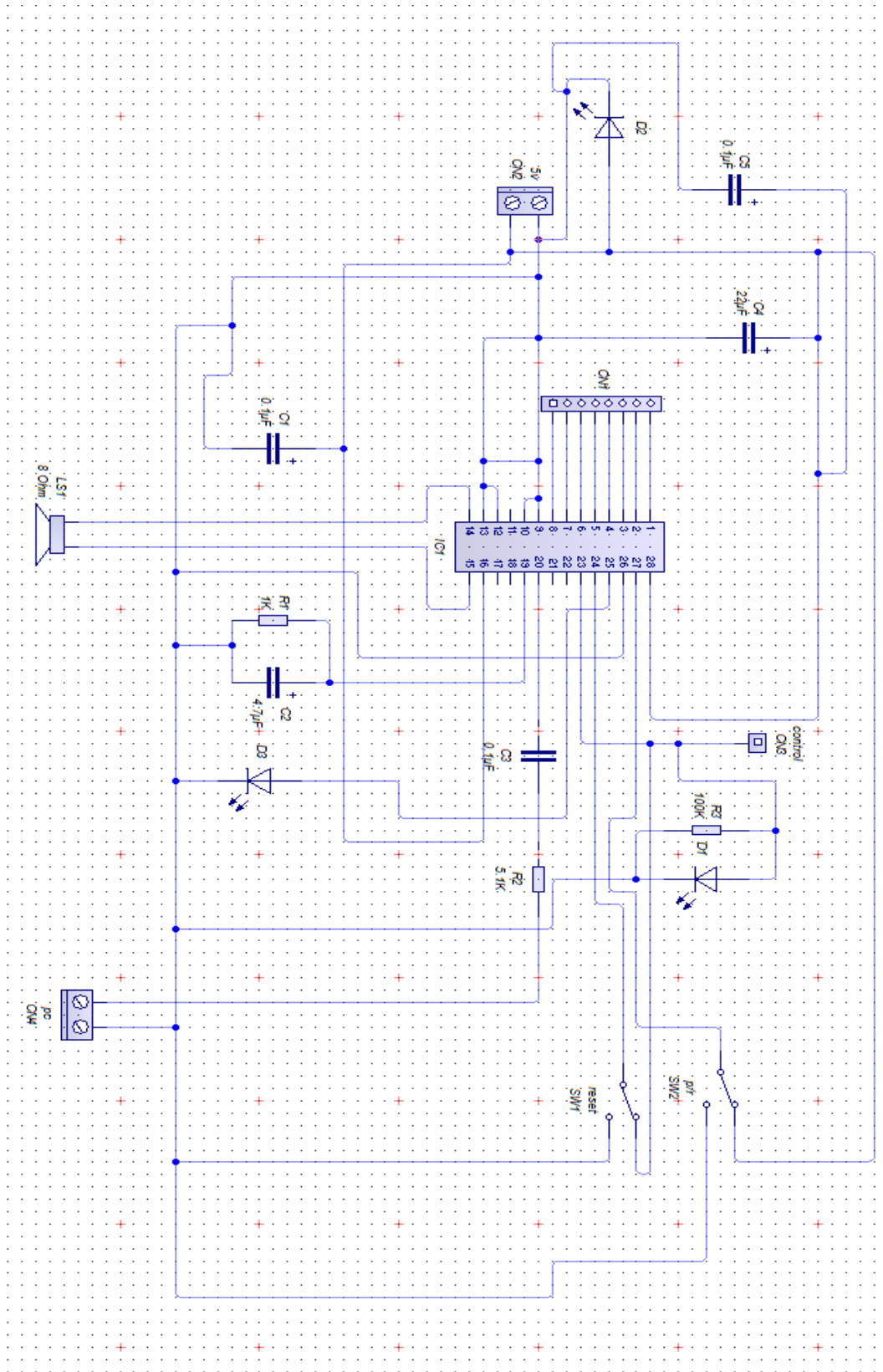


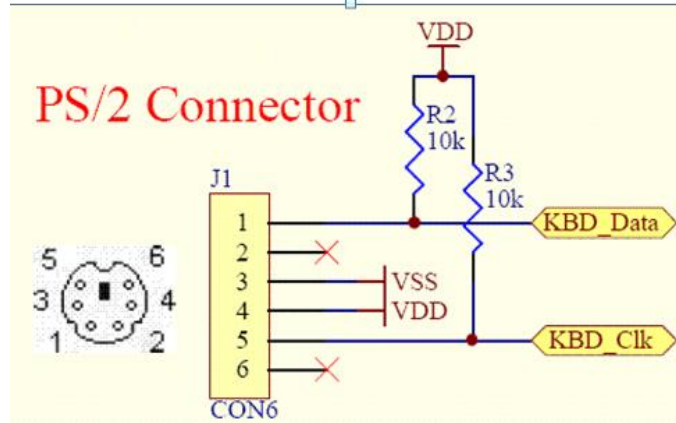
diagrama para fuente simetrica +5 -5 Vcc.



diseño de interfaz Pc/Isd.



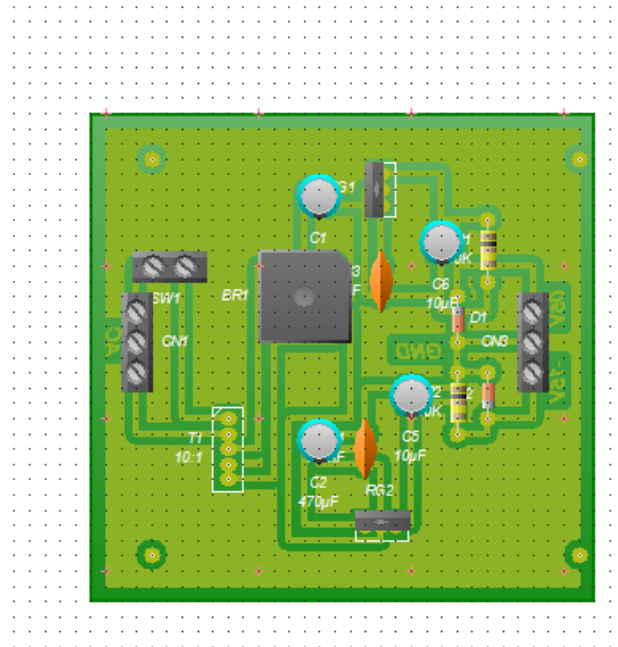
diseño de interfaz Pc/Isd. Con control del CE.



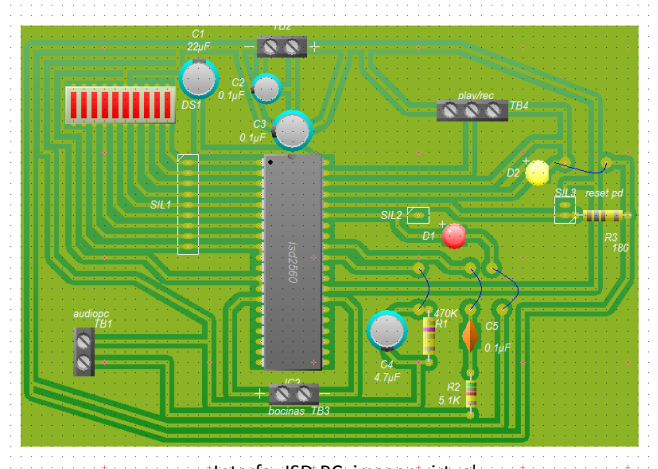
Conexión ps/2 al pic enfocando sistema de resistencias pull up



ANEXOS B pcb's



diseño de placa terminado en 3D.



Interfaz ISD-PC, imagen virtual.



ANEXOS C

- PROGRAMA BUILDER DE INTERFAZ

```
//-----  
  
#include <vcl.h>  
  
#pragma hdrstop  
  
#include "Unit1.h"  
  
//-----  
  
#pragma package(smart_init)  
  
#pragma link "HWINTERFACELib_OCX" // libreria ocx para que funcione el programa  
  
#pragma resource "*.dfm"  
  
TForm1 *Form1;  
  
//-----  
  
__fastcall TForm1::TForm1(TComponent* Owner)  
    : TForm(Owner)  
{  
}  
  
//-----  
  
void __fastcall TForm1::Button1Click(TObject *Sender)  
{  
    Close();    // boton cerrar  
}  
  
//-----
```



```
void __fastcall TForm1::Button2Click(TObject *Sender)
{

int direccionptl; // envia los datos escritos en el edit hacia el puerto paralelo
direccionptl = StrToInt(Edit1 -> Text);
Hwinterface1-> OutPort(0x378,direccionptl);
}

//-----

void __fastcall TForm1::FormCreate(TObject *Sender)
{
Hwinterface1-> OutPort(0x37A,200); //enciende los pines de control
Hwinterface1-> OutPort(0x378,0);
}

//-----

void __fastcall TForm1::Button3Click(TObject *Sender)
{
Hwinterface1-> OutPort(0x378,0);
Hwinterface1-> OutPort(0x37A,200); //boton que limpia pines de control
}

//-----

void __fastcall TForm1::Button4Click(TObject *Sender)
{
OpenDialog1->Title = "abrir fichero de sonido"; //configuración de caja de dialogo
OpenDialog1->Filter = "fichero de sonido(*.wav)|*.WAV";
OpenDialog1->InitialDir = "E:\\ROBERTO";
OpenDialog1->Execute();
}
```



```
Hwinterface1-> OutPort(0x37A,202);

    int direccionptl;

    direccionptl = StrToInt(Edit1 -> Text);

    Hwinterface1-> OutPort(0x378,direccionptl); // coloca direccion en el Puerto paralelo

MediaPlayer1->FileName=OpenDialog1->FileName; // active la caja de media player

MediaPlayer1->Open();

MediaPlayer1->Wait=true;

MediaPlayer1->Play();

MediaPlayer1->Close();

Hwinterface1-> OutPort(0x37A,200); //señal de fin de grabacion
```

```
}
```

```
//-----
```

```
void __fastcall TForm1::Edit1KeyPress(TObject *Sender, char &Key)
```

```
{
```

```
if (( Key < '0' || Key > '9') &&Key != 8 ) //establecemos que la caja de edicion solo
```

```
Key = 0; //admita valores de 0 a nueve
```

```
}
```

```
//-----
```

```
void __fastcall TForm1::Button5Click(TObject *Sender)
```

```
{
```




```
Hwinterface1-> OutPort(0x37A,202); //establecemos la activacion de un control del  
//puerto paralelo en un tiempo determinado de 10ms
```

```
Timer1->Enabled = true;
```

```
}
```

```
//-----
```

```
void __fastcall TForm1::Timer1Timer(TObject *Sender)
```

```
{
```

```
float tiempo= .33; //retardo de tiempo para activación de grabacion
```

```
tiempo--; //disminución del tiempo
```

```
Hwinterface1-> OutPort(0x37A,200);
```

```
if (tiempo == 0)
```

```
{
```

```
Timer1->Enabled = false;
```

```
}
```

```
}
```

```
//-----
```



- PROGRAMA MIKROC PIC-ISD-LCD

```
char *text="HOLA";
```

```
void main()
```

```
{
```

```
    TRISD = 0;          // PORTB is output
```

```
    Lcd_Init(&PORTD);  // Initialize LCD connected to PORTB
```

```
    Lcd_Cmd(Lcd_CLEAR); // Clear display
```

```
    Lcd_Cmd(Lcd_CURSOR_OFF); // Turn cursor off
```

```
    Lcd_Out(1, 1, text); // Print text to LCD, 2nd row, 1st column
```

```
TRISC = 0;                                // CONFIGURACION COMO PUERTO DE SALIDA
```

```
ADCON1=6;
```

```
TRISA=0;
```

```
while ( 1 )                                // CICLO INFINITO
```

```
{
```

```
    PORTA.F0=1 ;
```

```
    PORTA.F1=0 ;
```

```
    Delay_ms(100);
```

```
    PORTC = 20;
```

```
    Delay_ms( 300);          // ENVIA PRIMER DATO
```

```
    PORTA.F0= 0;
```

```
    Delay_ms(10);
```



```
PORTA.F0=1;
Delay_ms(200);

PORTC = 1 ;
Delay_ms( 300 );           // ENVIA SEGUNDO DATO
PORTA.F0= 0;
Delay_ms(10);
PORTA.F0=1;
Delay_ms(200);

PORTC = 43;
Delay_ms( 300 );           // ENVIA TERCER DATO
PORTA.F0= 0;
Delay_ms(10);
PORTA.F0=1;
Delay_ms(200);

PORTC = 6;
Delay_ms( 300 );           // ENVIA CUARTO DATO
PORTA.F0= 0;
Delay_ms(10);
PORTA.F0=1;
Delay_ms(200);

PORTC = 29;
Delay_ms( 300 );           // ENVIA 5 DATO
PORTA.F0= 0;
Delay_ms(10);
PORTA.F0=1;
```



```
Delay_ms(200);

// PORTC = 43;
// Delay_ms( 200 );           // ENVIA 6 DATO
// PORTA.F0= 0;
// Delay_ms(10);
// PORTA.F0=1;
//Delay_ms(100);

PORTA.F1=1 ;
Delay_ms(1000);
// break;

}
```



- **PROGRAMA MIKROC PIC-TECLADO/PS2-LCD**

```
unsigned short
keydata, special, down;

void main() {
    //CMCON = 0x07;      // Disable analog comparators
    // INTCON = 0;      // Disable all interrupts
    TRISD=0;
    Lcd_Init(&PORTD);    // Init LCD on PORTB
    Lcd_Cmd(Lcd_CLEAR); // Clear display
    Lcd_Cmd(Lcd_CURSOR_OFF);
    // Ps2_Init(&PORTB); // Init PS/2 Keyboard on PORTD
    Ps2_Config(&PORTB,1,0);
    Lcd_Out(1,1,"TEXT0");
    Delay_ms(100);      // Wait for keyboard to finish

    do {
        if (Ps2_Key_Read(&keydata, &special, &down)) {
            if (down && (keydata == 16)) { // Backspace
                Lcd_Cmd(LCD_MOVE_CURSOR_LEFT);
            }
            else if (down && (keydata == 13)) { // Enter
                Lcd_Cmd(Lcd_Clear);
            }
            else if (down && !special && keydata) {

                Lcd_Chr_CP(keydata);
            }
        }
    }
```



```
}  
  
    Delay_ms(10);    // debounce  
} while (1);  
}~-
```