



*Facultad
de Ingeniería*



Universidad Nacional
de Mar del Plata

PROYECTO FINAL

INSTRUMENTO ANALIZADOR DE IMPEDANCIA BIOELÉCTRICA

EMILIO MALDONADO

AGUSTÍN GIAI LEVRA

Director: Ing. Esteban L González

Facultad de Ingeniería – Universidad Nacional de
Mar del Plata

Septiembre de 2005



RINFI es desarrollado por la Biblioteca de la Facultad de Ingeniería de la Universidad Nacional de Mar del Plata.

Tiene como objetivo recopilar, organizar, gestionar, difundir y preservar documentos digitales en Ingeniería, Ciencia y Tecnología de Materiales y Ciencias Afines.

A través del Acceso Abierto, se pretende aumentar la visibilidad y el impacto de los resultados de la investigación, asumiendo las políticas y cumpliendo con los protocolos y estándares internacionales para la interoperabilidad entre repositorios



Esta obra está bajo una [Licencia Creative Commons Atribución- NoComercial-CompartirIgual 4.0 Internacional](https://creativecommons.org/licenses/by-nc-sa/4.0/).



*Facultad
de Ingeniería*



Universidad Nacional
de Mar del Plata

PROYECTO FINAL

INSTRUMENTO ANALIZADOR DE IMPEDANCIA BIOELÉCTRICA

EMILIO MALDONADO

AGUSTÍN GIAI LEVRA

Director: Ing. Esteban L González

Facultad de Ingeniería – Universidad Nacional de
Mar del Plata

Septiembre de 2005

INDICE

Resumen.....	2
Introducción.....	3
Anteproyecto.....	4
Proyecto	
Método empleado.....	7
Análisis de resolución.....	8
Descripción de las etapas que conforman el equipo	
Bloques digitales.....	10
Bloques analógicos.....	12
Bloques mixtos.....	14
Circuito general completo.....	16
Funcionamiento básico.....	17
Manual de Mantenimiento	
Especificaciones generales.....	22
Breve descripción del funcionamiento.....	22
Procedimiento para lo localización de fallas.....	26
Manual de operación.....	29
Mediciones.....	32
Conclusiones.....	34
Bibliografía.....	35
Anexo 1: Código de los programas	
Código en C del microcontrolador 1.....	37
Código en assembler del microcontrolador2.....	55
Anexo 2: Lista de componentes.....	80

RESUMEN

En este proyecto se llevó a cabo el diseño y construcción de un analizador BIA monofrecuencia (50KHZ) de medición distal (electrodos en mano y pie homolateral) destinado a medir la impedancia corporal de un ser humano, para deducir de este valor (y de otros que se ingresan al equipo por teclado) algunos parámetros que definen su composición corporal.

Su funcionamiento fue basado en una plataforma controlada por dos microcontroladores PIC de Microchip. Uno de ellos fue el encargado de sintetizar la señal inyectada sobre el cuerpo humano, luego de un filtrado y posterior transducción a corriente, y de generar una lógica de control para la adquisición de la onda de tensión. Su programación se llevó a cabo en lenguaje Assembler.

La corriente alterna utilizada fue de 0.8 mA rms (por debajo del nivel de percepción). La caída de tensión consecuente y su desfase respecto de la onda de corriente determinaron la impedancia corporal. La parte resistiva de dicha impedancia depende del paso de la corriente a través de las soluciones electrolíticas intra y extracelulares. La reactancia se relaciona con las propiedades dieléctricas de los tejidos, o bien por la acumulación temporal de cargas sobre las membranas celulares o sobre otras interfaces sumergidas en la solución electrolítica que se comportan capacitivamente [8],[9].

El segundo microcontrolador se encargó del procesamiento de los datos adquiridos, el manejo del display de LCD y los cálculos asociados a la obtención de los parámetros anteriormente citados. Este microcontrolador se programó en Lenguaje C.

La recuperación de la señal se realizó con electrodos de ECG conectados a un amplificador de instrumentación (INA114), que provee un gran rechazo a modo común (>120dB) y una elevadísima impedancia de entrada (>GΩ).

Se incluyó una circuitería de autocalibración, basada en relays y una resistencia patrón para corregir cualquier error en fase o amplitud que pueda aparecer previa a cada medición.

La conversión analógico-digital se llevó a cabo con un conversor Philips TDA 8703 que cuenta con un tiempo de muestreo sumamente corto (~2nseg) con lo que ayuda a detectar pequeños movimientos de fase.

Finalmente, para ingresar otros parámetros que se contemplan en los algoritmos que realiza el equipo (peso, sexo, edad y altura) se incluyó una pequeña botonera.

Las mediciones efectuadas con el instrumento presentaron gran concordancia al cotejar los valores medidos con el valor real de juegos de resistencias y capacitores. Recalcando que los valores de los mismos se eligieron dentro del rango de valores esperables para una impedancia corporal ($400\Omega < R < 800\Omega$, $30\text{nF} < C < 80\text{nF}$). Los errores hallados dentro de este rango se fueron menores al 1% para el cálculo de la resistencia y del orden del 2% para el ángulo de fase.

INTRODUCCIÓN

El análisis de la composición corporal es una herramienta de diagnóstico utilizada en diversas ramas de la medicina y en nutrición.

El analizador de impedancia bioeléctrica (BIA, *bioimpedance analyzer*) es un instrumento no invasivo que permite estimar la composición corporal de un individuo mediante la medición de su resistencia y reactancia. Los valores arrojados por este dispositivo son: TBW = agua corporal total (*total body water*), FFM=materia magra (*fat free mass*) y FM=materia grasa (*fat mass*). Estos parámetros son considerados de interés clínico debido a que permiten estudiar la evolución de los tejidos corporales como respuesta a diversas patologías y a sus tratamientos.

En la actualidad existen diversos equipos destinados a la medición de estos parámetros, y su complejidad varía en función de las variables que miden, su presentación al usuario y fundamentalmente por su costo. Un analizador standard, de medición resistiva únicamente, como el utilizado en la clínica Pueyrredón cuesta aproximadamente 1000 pesos pero los hay de hasta U\$S 5000.

Para este proyecto se plantea el desarrollo de analizador monofrecuencia que mide tanto resistencia como capacidad corporal. Esta elección se basó principalmente en la concordancia existente entre diversos autores acerca de la mejor estimación de los parámetros corporales que surgen del conocimiento de su capacidad [2],[3].

Dado que el ángulo de fase para un individuo sano varía entre 3 y 8 grados [8], la resolución del equipo se adopta en 0.1grado (51 posibles valores).

Para la resistencia, pueden esperarse valores de aproximadamente $500 \pm 100 \Omega$. En este caso la resolución adoptada fue de 1ohm.

Respecto del método de medición se opta por la medición tetrapolar distal homolateral. En ella se ubican cuatro electrodos (uno en la mano, uno en la muñeca, uno en el tobillo y el otro sobre el pie, todos del mismo lado). En [7] se plantea la medición "*de pie a pie*" aunque dicha técnica no se encuentra validada, con lo cual se empleó la metodología previamente citada cuyo uso es generalizado.

Respecto a la frecuencia de trabajo, existen en el mercado diferentes equipos con frecuencias variables o fijas. La elección es nuevamente conservadora y si bien no existe una estandarización para este tipo de análisis [11] se adopta la medición monofrecuencia a 50KHz dado que permite obtener el máximo desfase de la onda en su paso por el cuerpo [8].

Además cumplir con todas las especificaciones citadas, se plantea un diseño portátil, de alimentación baterías, sencillo e intuitivo para usar.

ANTEPROYECTO

En este proyecto se comenzó por diseñar y poner en funcionamiento el microcontrolador y la circuitería asociada que realizan el control del equipo. Para ello se optó por un PIC16F873, ya que se trata de un dispositivo sumamente versátil, económico (\$27) y puede encontrarse en las casas de electrónica de la ciudad. Con microcontrolador como pieza fundamental de la plataforma, comenzó a desarrollarse el software en assembler para el manejo del display de LCD. Este último es un Hitachi compatible de 16x2 caracteres. Si bien hubiera alcanzado con un display de 16x1 caracteres, dado que estos no son de tan amplio uso y en consecuencia, se importan al país en menor cantidad, su precio es sensiblemente mayor a pesar de ser mas pequeños.

Dada la necesidad de realizar operaciones matemáticas que complican en gran medida la programación en assembler, se optó por "traducir" el programa hecho hasta el momento a lenguaje C y continuar programando en este lenguaje para luego generar el código que se graba en el microcontrolador con el programa PIC-C.

Como el software del microcontrolador creció más de lo previsto en un primer momento, hubo que reemplazar el microcontrolador PIC16F873 por el PIC16F876 que es pin-compatible con su predecesor pero cuenta con el doble de memoria de programa.

A continuación se comenzaron a desarrollar las pruebas de adquisición de señales por parte del microcontrolador a través de un conversor externo. Dada la considerable precisión que se había planteado para el equipo, se necesitaba un conversor capaz de tomar la muestra en un tiempo inferior a los 5,5nseg. No podía conseguirse en la ciudad un conversor con dicha especificación. Finalmente se implementó un conversor Philips TDA8703 flash con tiempo de adquisición de 2nseg, adquirido en Buenos Aires. Este dispositivo cumplió perfectamente con las especificaciones aunque dado su valor (35 u\$s) aumentó sensiblemente el costo del equipo.

El siguiente paso fue la generación de la señal inyectada. Se comenzó por plantear un oscilador que trabaje a la frecuencia deseada (50KHz). Si bien el diseño generaba la onda correctamente, resultaba dificultosa la lógica para adquirir la señal en instantes sumamente precisos de tiempo. La alternativa elegida para hacer frente a este inconveniente fue la generación de la señal en forma digital por parte de un segundo microcontrolador que genere a su vez la lógica de adquisición. Se optó por un microcontrolador PIC16F873, dado que se contaba ya con buen manejo de estos dispositivos y todos los elementos para programarlo (se disponía de programador y software del microcontrolador principal).

El paso siguiente fue entonces, hacer trabajar en forma coordinada ambos microcontroladores. Para ello hubo que hacer uso de un buffer tri-state DM74LS126 para evitar colisiones en el bus que comparten los dos PIC, el conversor ADC y el Display de LCD.

Una vez funcional la plataforma con sus dos microcontroladores se trabajó en el pasaje de la señal generada en forma digital a forma analógica y libre de espurias. Se implemento un conversor digital-analógico DAC800, que es sumamente económico y

cumplió efectivamente con su tarea. La etapa de filtrado de armónicas siguiente se llevó a cabo con un Butterworth activo de segundo orden. Es remarcable que la etapa de filtrado no fue crítica, dado que en la síntesis de la señal digital se usó un gran número de muestras, con lo cual las armónicas espurias quedaron bien alejadas en frecuencia y por lo tanto fácilmente filtrables.

Una versión preliminar de la etapa de filtrado se llevó a cabo con el filtro mostrado en la figura 1.

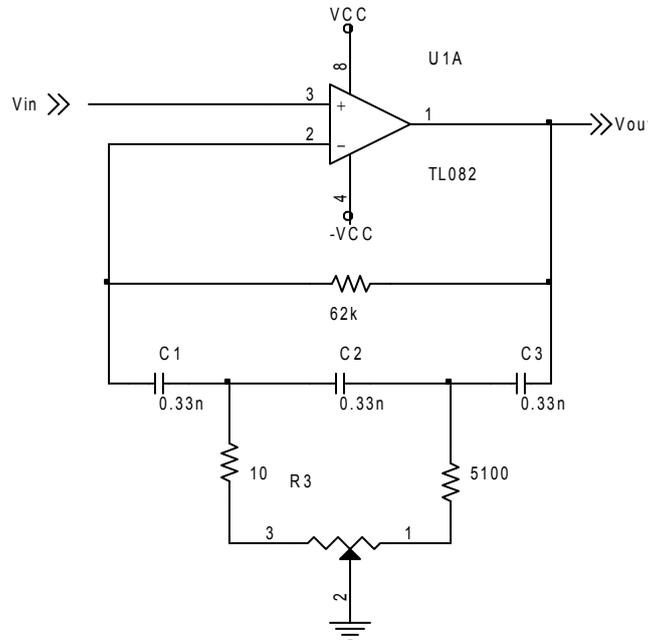


Figura 1

Este arreglo presenta una pendiente mayor a 20dB/dec del Butterworth, sin embargo si se examina el gráfico de Bode simulado en Spice (gráficos 1 y 2), se observa que la variación que tiene en fase es muy abrupta para la frecuencia de trabajo.

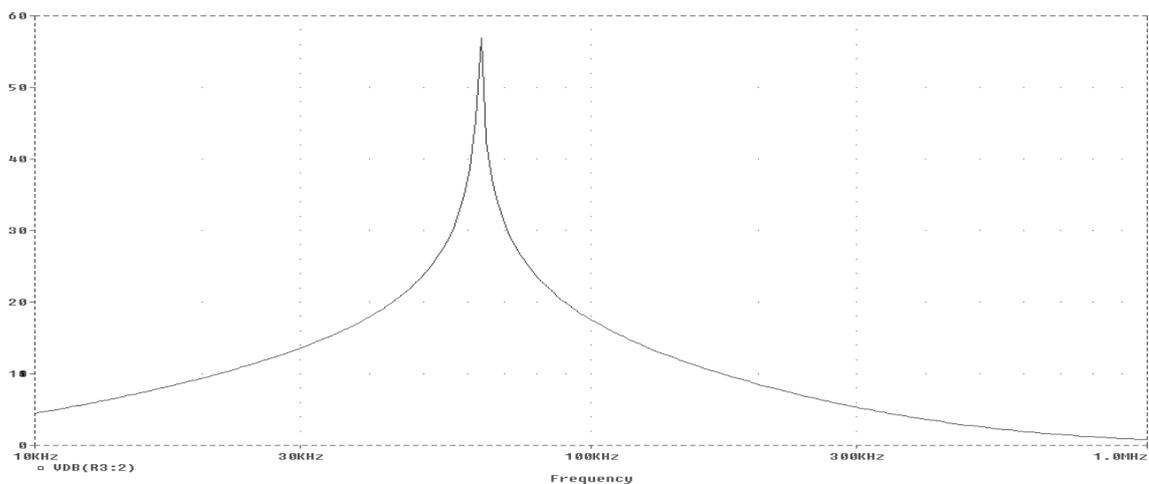


Gráfico 1

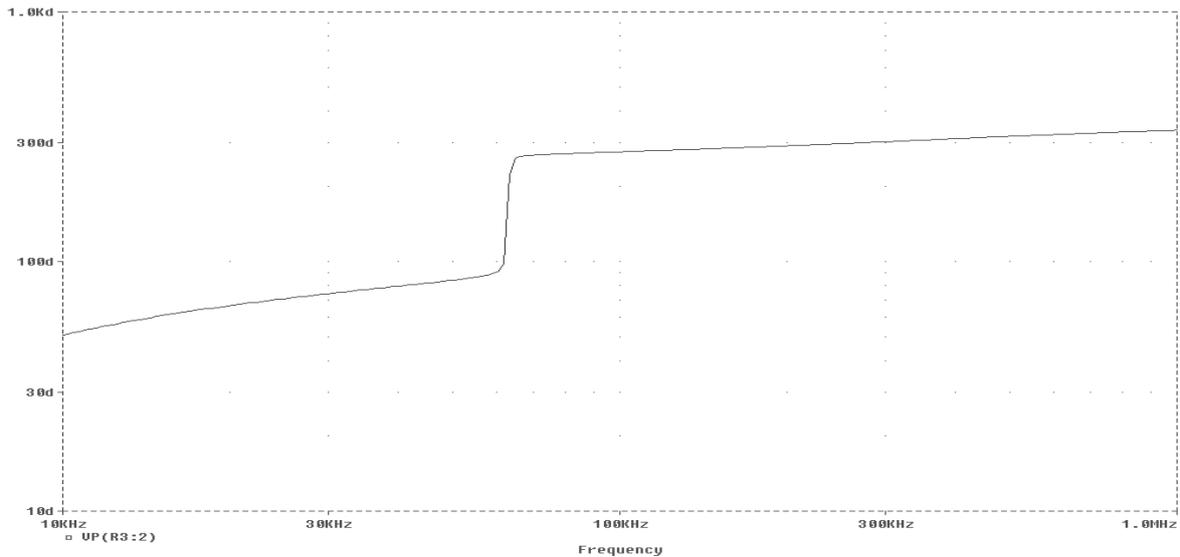


Gráfico 2

Prácticamente, se observaron los grandes cambios de fase del filtro para condiciones casi idénticas.

Dada esta gran variabilidad en su funcionamiento, se descartó este diseño, reemplazándolo por el Butterworth.

Como la señal que debe inyectarse en el cuerpo debe ser de corriente, hubo que implementar en el diseño un amplificador de transconductancia. Se usó el OTA3080 que es sencillo, económico y mostró excelente desempeño al ser cargado con un amplio rango de impedancias diferentes.

Para la recuperación de la señal de tensión que se genera en el cuerpo, se utilizaron electrodos descartables de cardiología, que pueden conseguirse con facilidad en farmacias y comercios del rubro médico. Los mismos se conectaron al dispositivo a través de cables blindados (símil cable de micrófono pero de mayor calibre) y a un amplificador de instrumentación INA 114. Este dispositivo es costoso y no se encuentra disponible en el país, pero se implementó debido a la disponibilidad de los mismos en el laboratorio de comunicaciones. Presenta excelente desempeño en rechazo a modo común e impedancia de entrada, aunque podría llegar a reemplazarse para una futura producción a mayor escala, por un arreglo de operacionales que imiten la estructura interna del INA y de esta forma reducir los costos.

Por último se decidió agregarle al equipo un sistema de auto calibración llevado a cabo por dos relés los cuales conectan una carga patrón bien conocida, miden la amplitud y la fase producida por el equipo mismo, luego desconectan dicha carga y conectan al paciente. Con los valores medidos previamente y los obtenidos del paciente obtienen la amplitud y el desfasaje buscados. Este sistema hace innecesaria la calibración periódica y manual que requieren otros instrumentos similares y mejora la reproducibilidad de las mediciones.

PROYECTO

Método empleado

El método empleado por el equipo para medir la impedancia del cuerpo humano se basa en tomar dos muestras de la tensión generada sobre el mismo al inyectarle la corriente de características ya mencionadas, para luego obtener, de un sistema de dos ecuaciones con dos incógnitas, la amplitud y fase de dicha tensión respecto de la corriente inyectada. Conociendo estos dos valores puede conocerse el valor de la amplitud y fase de la impedancia corporal:

$$v(t) = A * \text{sen}(wt - \theta) + V_m$$

donde $v(t)$ = tensión sobre el cuerpo

V_m = Valor medio expresado en valor digital (0-255)

$w = 2 * \pi * 50000$ rad/seg

A = Amplitud expresada en valor digital (incógnita)

θ = Fase (incógnita)

X_c = reactancia capacitiva expresada en Ω (incógnita)

Tomando dos muestras en tiempos conocidos:

$$\text{En } t = t_1 = 0.8 \mu\text{seg} \Rightarrow N_1 = A * \text{sen}(w * t_1 - \theta) + V_m$$

$$\text{En } t = t_2 = 10.4 \mu\text{seg} \Rightarrow N_2 = A * \text{sen}(w * t_2 - \theta) + V_m$$

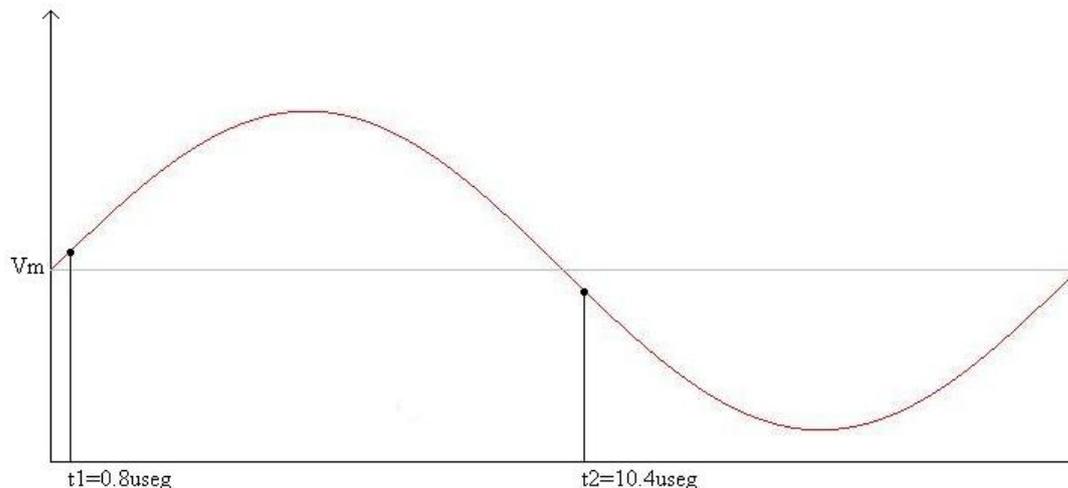


Gráfico 3

Despejando:

$$A = \frac{(N_1 - V_m)}{\text{sen}(w * t_1 - \theta)}$$

$$\theta = \text{arctg} \left(\frac{(N_2 - V_m) * \text{sen}(w * t_1) - (N_1 - V_m) * \text{sen}(w * t_2)}{(N_2 - V_m) * \text{cos}(w * t_1) - (N_1 - V_m) * \text{cos}(w * t_2)} \right)$$

Con lo que se obtiene:

$$R = \frac{A * \frac{(3.26 - 1.55)}{255} + 1.55}{0.8mA}$$

donde el numerador es el

valor digital de A pasado a Volts, ya que 3.26 y 1.55 son los valores de tensión máximo y mínimo admisibles a la entrada del ADC

$$X_C = R * \text{tg}(\theta)$$

Se observa en las ecuaciones que existe un valor medio presente en la señal de tensión generada sobre el paciente a pesar de que el valor medio de la corriente a inyectar es cero. Esto se debe a que el conversor analógico-digital utilizado (TDA8703) opera con un rango de tensiones de entrada que va desde 1.55 V a 3.26 V, por lo que debió sumarse a la señal que entra a este último un valor de $(3.26 - 1.55)/2 = 0.85$ V para obtener (conjuntamente con un ajuste de ganancia) una excursión máxima sobre el rango del ADC. Más allá de que este valor es ajustado mediante un preset dispuesto a tal fin, el equipo lo computa antes de cada examen para evitar derivas por temperatura, por envejecimiento de componentes, etc.

Análisis de resolución

El siguiente análisis pretende obtener la resolución en fase aproximada del equipo con el Conversor analógico-digital disponible (n=8 bits)

Se plantea una señal sobre el cuerpo de la forma:

$$v(t) = A * \text{sen}(wt + \theta)$$

Se intenta muestrear a la señal en los puntos donde la misma posee máxima pendiente, ya que de esa forma un cambio en un bit del valor muestreado corresponderá al mínimo desplazamiento de fase detectable. Bajo esta premisa y considerando en θ pequeño, se asume que se muestrea en $t=0$ para simplificar los cálculos (ver figura 3):

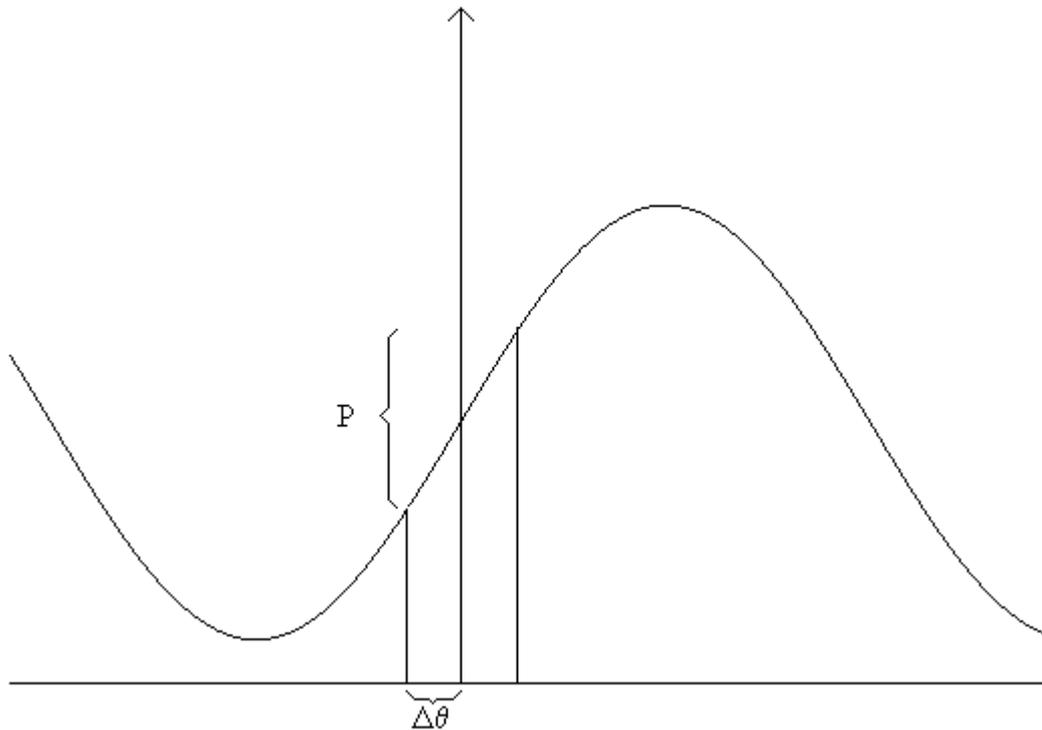


Gráfico 4

$$v(t) = A \cdot \text{sen}(\theta) \Rightarrow \frac{dv}{dt}(t) = A \cdot \cos(\theta) \approx \frac{\Delta v}{\Delta \theta}$$

$$A \cdot \cos(\theta) \leq \frac{P/2}{\Delta \theta} \Rightarrow \Delta \theta \leq \frac{P}{2 \cdot A \cdot \cos(\theta)}$$

$$P = \frac{3.26 - 1.55}{255}$$

$$A = 0.008 \cdot \sqrt{2} \cdot 500 \text{ , asumiendo } R=500\Omega$$

Operando con las últimas ecuaciones:

$$\theta \leq \arccos\left(\frac{0.00592}{\Delta \theta}\right)$$

Con lo cual se obtiene una resolución de $\Delta \theta \geq 3.5$ grados para $\theta \leq 14$ grados.

Cabe recalcar que este análisis esta hecho bajo la premisa de tomar una muestra sola de cada t_i , sin considerar ningún tipo de promediación.

NOTA: Con un valor de $\theta \approx 5$ grados, los tiempos óptimos para muestrear serían

$t_1 \approx 277 \text{ nseg}$ y $t_2 \approx 10.277 \mu\text{seg}$, sin embargo se eligieron $t_1 \approx 0.8 \mu\text{seg}$ y $t_2 \approx 10.4 \mu\text{seg}$ debido a que estos son los valores más cercanos que se pueden conseguir con la máxima resolución temporal del PIC utilizado en el DDS.

Descripción de las etapas que conforman el equipo

El equipo está formado básicamente por cuatro bloques digitales, cuatro bloques analógicos y dos mixtos. Ver Figura 3.

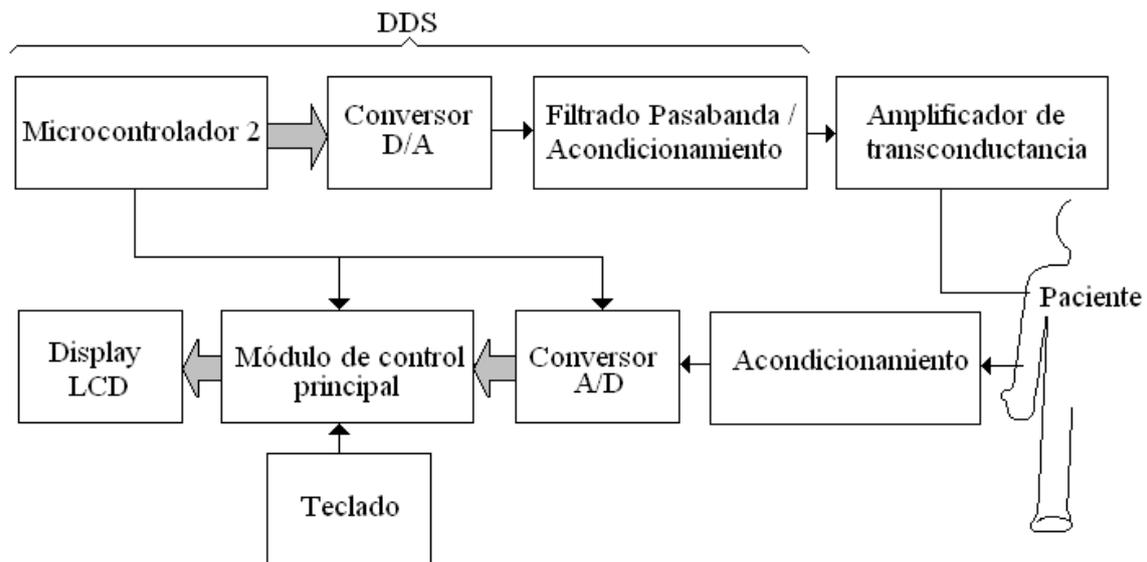


Figura 2

Bloques digitales

1. **Módulo de control principal:** implementado a partir de un microcontrolador PIC16F876 de Microchip. Tiene la función de controlar a los restantes bloques digitales que conforman el equipo y de realizar los cálculos para obtener los resultados del examen (TBW, FFM, FM). Debido a que dichos cálculos son en punto flotante, se optó por desarrollar el software bajo una distribución del lenguaje de programación "C" para microcontroladores PIC perteneciente a la empresa CCS®, denominada PIC-C. Ver circuito general completo.
2. **Microcontrolador II:** Esta etapa, en asociación con un DAC (DAC0800) y otra etapa de acondicionamiento, forman un DDS que tiene la función de sintetizar la señal de tensión senoidal a partir de la cual se obtendrá la señal de corriente que se inyectará al paciente, así como también de generar la señal de CLK para el ADC y la de interrupción para el *Módulo de control principal*. Se basa en un microcontrolador PIC16F873 de Microchip que recorre una tabla grabada en su

memoria ROM donde se encuentran los valores de la senoidal a generar. Se optó por este método de síntesis por dos razones:

- a) Se tiene un conocimiento preciso del valor de la onda sintetizada en determinados instantes.
- b) Gran versatilidad ante la eventual necesidad de realizar cambios de frecuencia, forma o amplitud de la señal generada (basta con modificar el software y la etapa de filtrado)

Debido a que el tiempo entre muestras debe ser perfectamente conocido y permanecer constante, se optó por desarrollar un software en *Assembler* bajo el entorno de programación MpLab®, el cual recorre cíclicamente una tabla de 25 posiciones grabada en memoria de programa para establecer en el puerto los valores que se muestran a continuación:

Nro _{muestra}	$F(x)=30*\text{SENO}(6,28*\text{Nro}_{\text{muestra}}/25)+127$	Round(F(x))
0	127	127
1	134,456994	134
2	141,445911	141
3	147,528053	148
4	152,321642	152
5	155,525784	156
6	156,939353	157
7	156,473619	156
8	154,157817	154
9	150,13731	150
10	144,664467	145
11	138,082819	138
12	130,805499	131
13	123,289307	123
14	116,006036	116
15	109,41286	109
16	103,923634	104
17	99,8829193	100
18	97,5443523	98
19	97,0547257	97
20	98,4447736	98
21	101,627242	102
22	106,402367	106
23	112,470412	112
24	119,450484	119

El factor de amplitud 30 es el mínimo valor que proporciona una simetría aceptable en la senoidal; valores mayores son innecesarios ya que saturan al

OTA y por ende deben atenuarse antes de llegar a este. El microcontrolador se encuentra trabajando a su velocidad máxima: un clock de 20 Mhz que implica un ciclo de instrucción de 200 nseg, ya que la frecuencia de instrucción es igual a la de clock dividida cuatro. Teniendo en mente esto y sabiendo también que el número de instrucciones necesarios para sacar un valor digital por el puerto es de 4:

$$\text{Periodo de la señal generada} = 25 \text{ Muestras} * 4 * 200 \frac{\text{nseg}}{\text{Muestra}} = 20 \text{ useg}$$

3. Recordemos que la frecuencia de la señal a sintetizar es de 50 KHz, con lo cual su periodo es de 20 μ seg. Ver Circuito general completo
4. **LCD:** Posibilita el ingreso de los datos del paciente (edad, sexo, altura y peso) así como también informa los resultados del examen. Ver Circuito general completo
5. **Buffer TriState:** Se utiliza para evitar colisiones sobre el bus de datos del instrumento. Implementado con un 74LS126. Ver Circuito general completo

Bloques Analógicos

1. **Filtrado / Acondicionamiento:** Conformado por un seguidor y control de amplitud (TL082), y un filtro pasabajos activo de orden 2 del tipo Butterworth con frecuencia de corte = 100 KHz implementado con un TL081, del cual también se utiliza su control de offset. Cumple la función de acondicionar la señal que se obtiene del DAC, con la cual posteriormente se ataca al OTA. Ver figura 4.

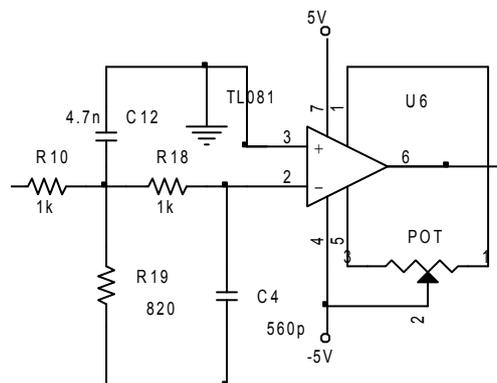


Figura 3

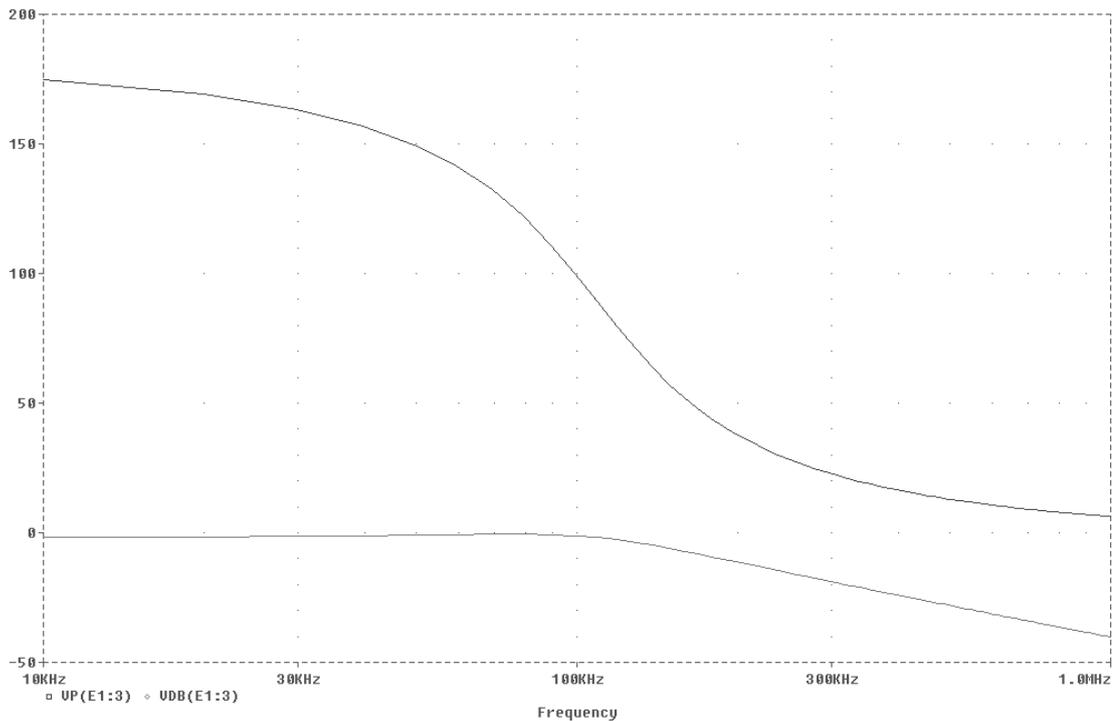


Gráfico 5: Diagrama de bode del Filtro Butterworth utilizado.

- 2. Amplificador de transconductancia (OTA):** Genera la señal de corriente que se inyecta al paciente, a partir de la de tensión entregada por el DDS. Está implementado en base a un LM3080, el cual posee un control de ganancia.

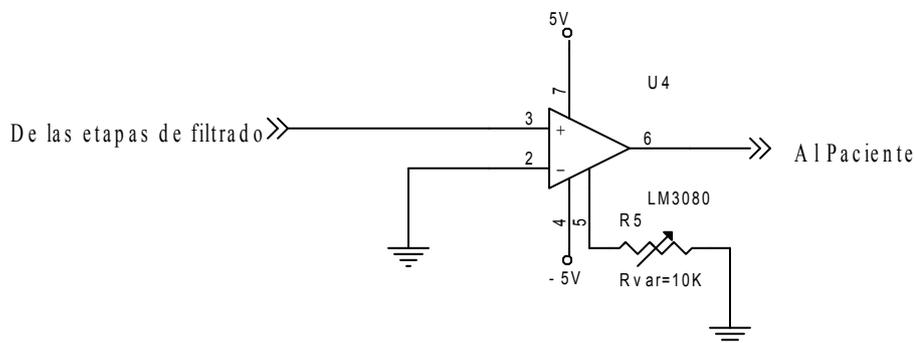


Figura 4

- 3. Relés de calibración:** Previo al comienzo de cada examen, el equipo lleva a cabo una rutina de autocalibración, la cual consiste en la medición de una resistencia patrón para obtener del mismo la fase propia que introduce la el sub-sistema *DDS + OTA* y tenerla en cuenta en la medición del paciente, disminuyendo así errores por posibles derivas por temperatura o envejecimiento de los componentes. Para ello consta de un arreglo de dos relés que son comandados por el *Módulo de control principal*, y que primeramente conectan la resistencia patrón para obtener a través de ella la fase intrínseca del sistema, para luego sí conectar al paciente para realizar el análisis.

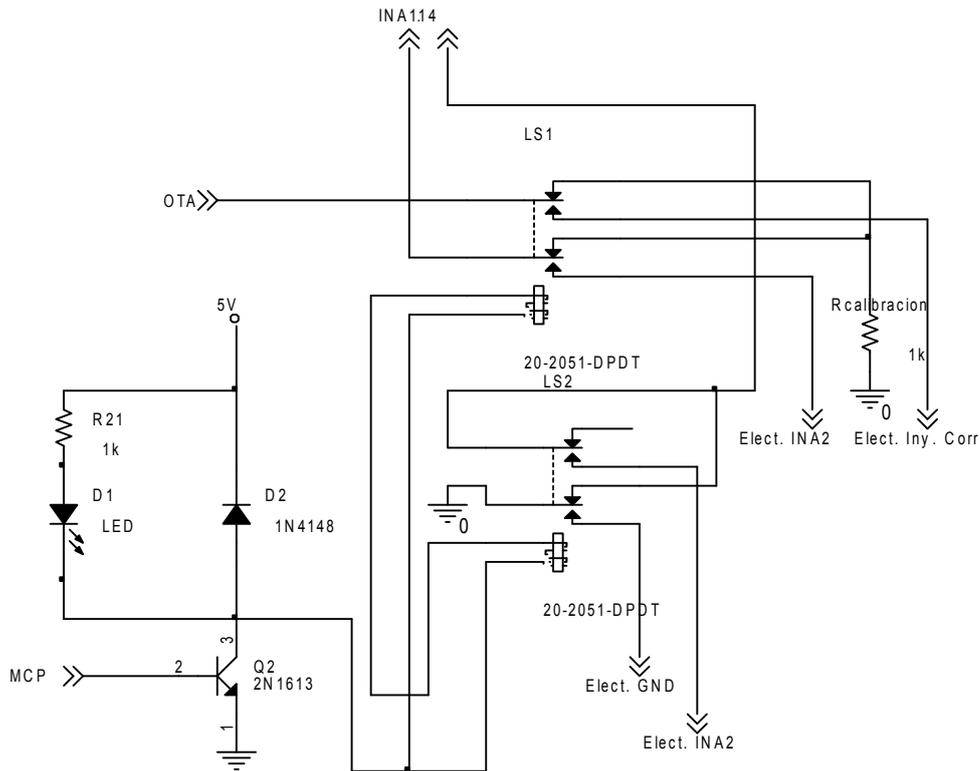


Figura 5

4. **Acondicionamiento:** Esta etapa es la encargada de:

- a) levantar la señal de tensión generada en el cuerpo mediante un amplificador de instrumentación *INA114*.
- b) amplificar dicha señal y sumarle a la misma un nivel de continua de 0.85 V, ambas acciones con el propósito de obtener máxima excursión sobre el rango de tensiones permitido a la entrada del convertor analógico – digital. Está implementada en base a un doble AOV TL082, uno de los cuales se utiliza como sumador y el otro como amplificador.

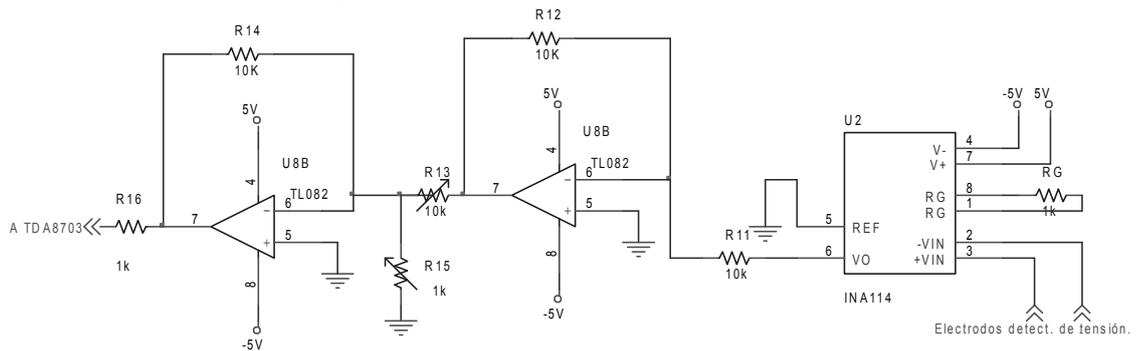


Figura 6: Adquisición de la señal y acondicionamiento

Bloques mixtos

1. **Convertor analógico-digital:** Cumple la función de digitalizar la señal entregada por la etapa de acondicionamiento para ingresarla al *Módulo de Control Principal*. Se utilizó un CI *TDA8703* (*flash*), debido fundamentalmente a su muy pequeña ventana de muestreo ($t_{ds} = 2$ nseg). Ver figura 5.

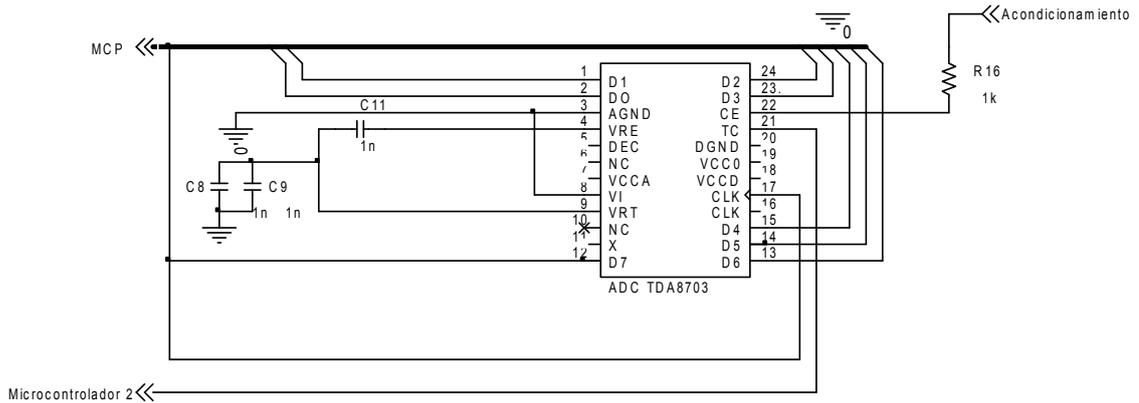


Figura 7

2. **Convertor digital-analógico:** Esta etapa del DDS convierte los valores digitales que le entrega el *Microcontrolador II* en una señal analógica que, luego de ser debidamente filtrada se ingresa al *OTA*. Se utilizó un CI *DAC800*.

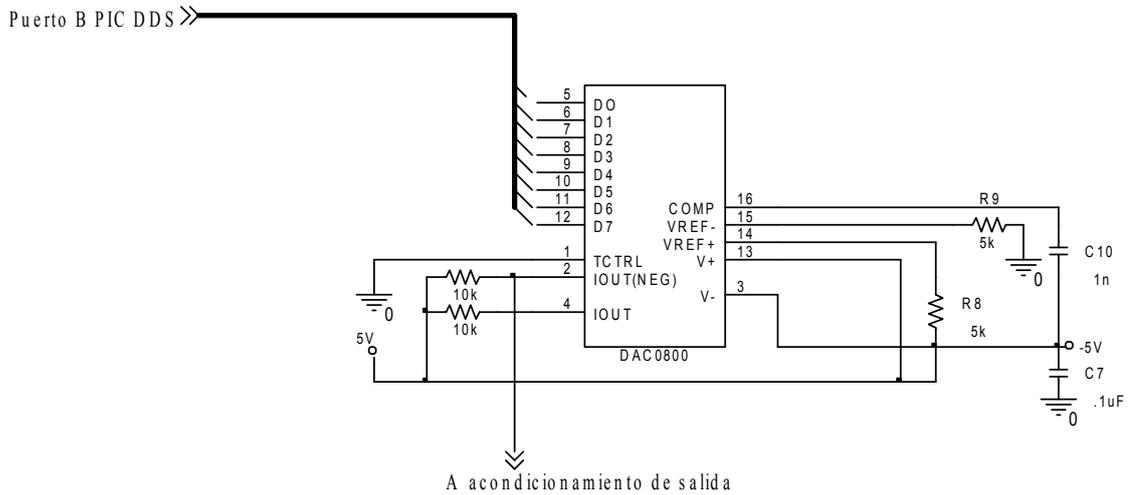
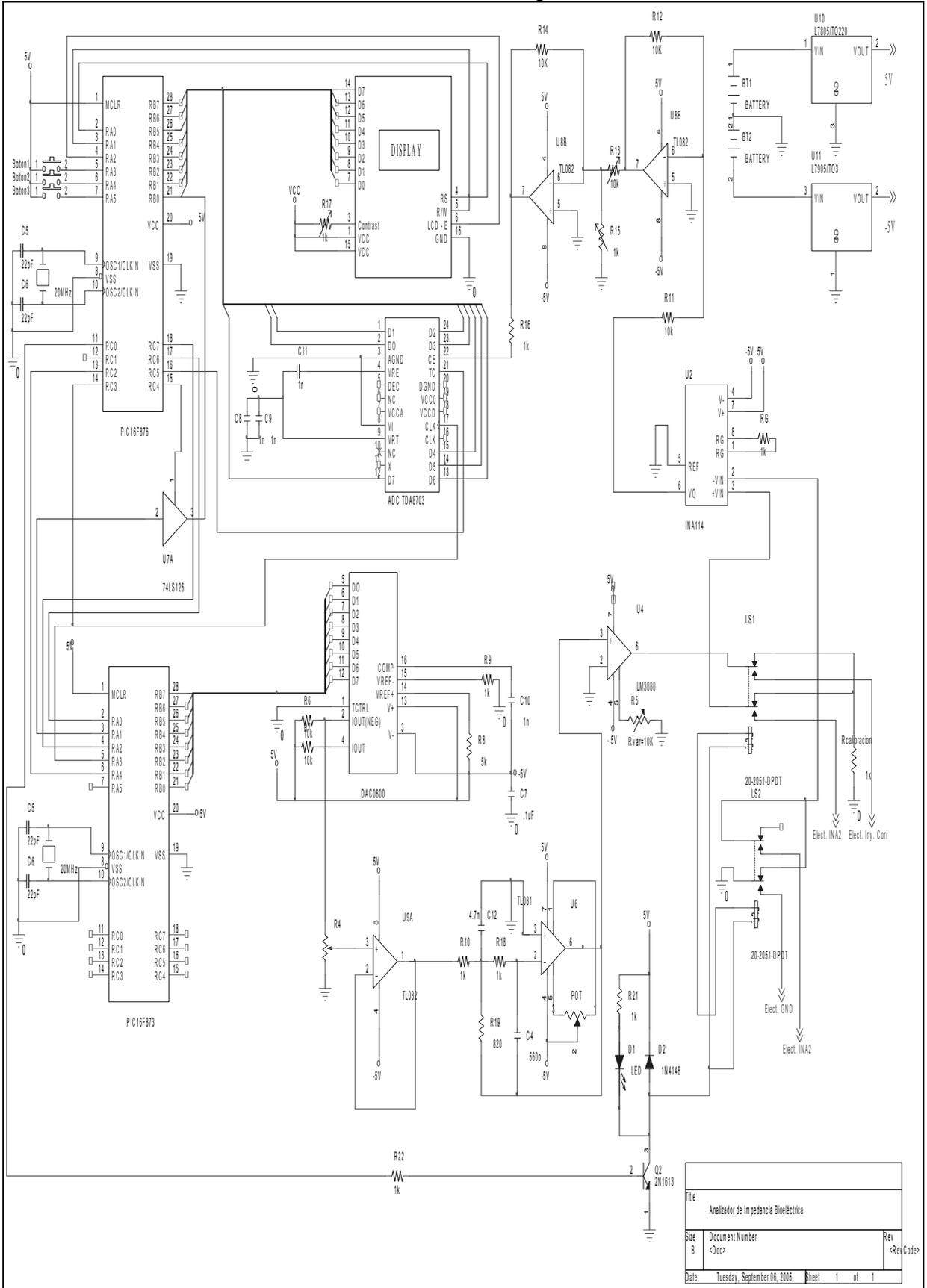


Figura 8

Circuito General Completo



Title		
Analizador de Impedancia Bioeléctrica		
Size	Document Number	Rev
B	<Doc>	<Rev Code>
Date:	Tuesday, September 06, 2005	Sheet 1 of 1

Funcionamiento básico

A continuación se enumera la secuencia de pasos que describen el funcionamiento básico del equipo. Para el manejo de las líneas de control y buses de datos de los microcontroladores se recomienda ver la figura 9.

1. Al encender el equipo, el *Módulo de control principal* envía los comandos de inicialización al *LCD*.
2. El usuario ingresa datos de peso, edad, altura y sexo del paciente a través de los tres botones dispuestos a tal fin y luego espera la confirmación del usuario para iniciar el estudio. En esta etapa el *DDS* se encuentra deshabilitado (no está generando señal), el arreglo de relés tiene conectada a la salida del *OTA* la resistencia patrón, y el conversor analógico-digital no está tomando muestras.
3. El usuario confirma el inicio del examen, con lo cual el *Módulo de control principal*:

- a) Setea la línea *CtrlBuffer* de forma tal que la línea *Clk_{DDS-INT}* que maneja el *Microcontrolador II* llegue a la pata que genera la interrupción externa sobre el *Módulo de control principal*, y pone la línea *Clk_{BIA}* en estado de alta impedancia ya que en esta etapa los pulsos de muestreo para el conversor analógico-digital serán proporcionados por el *Microcontrolador II*.
- b) Pone en “uno” la línea *GenerarINT* mediante la cual le comunica al *Microcontrolador II* que debe generar los pulsos de interrupción por la pata *Clk_{DDS-INT}* en instantes determinados, además de sacar por el puerto los valores digitales que representan la senoidal a sintetizar. Estos valores que sintetizan la senoidal así como los pulsos de interrupción (ambos generados por el *Microcontrolador II*) no se comienzan a generar hasta que el *Módulo de control principal* no lo indica mediante un valor alto en la línea *Habilitación*.
- c) El *Módulo de control principal* pone en alto la línea *Habilitación* con lo cual el *Microcontrolador II* comienza a generar la senoidal con la que ataca al DAC y también genera la señal de interrupción externa para el *Módulo de control principal*. Esta señal no solo se usa para generar una interrupción, sino también es la que produce el flanco necesario en la pata de *CLK* del conversor analógico-digital para que este adquiera, en determinados instantes, el valor de tensión presente sobre la carga, originado a su vez por la inyección de corriente debida al amplificador de transconductancia.

Cabe acotar que la rutina de atención de la interrupción externa del *Módulo de control principal*, lee el valor digital presente en el bus de datos del conversor analógico-digital y lo acumula en una de dos variables en memoria RAM, dependiendo del valor de la línea *MuestraPar* informado por el *Microcontrolador II*. El hecho que se genere la señal de adquisición en el conversor analógico-digital al mismo tiempo que la señal de interrupción externa sobre el *Módulo de control principal* no es un problema, ya que el conversor demora un máximo de 20 nseg en realizar una conversión, mientras que la respuesta del PIC16F876 a la interrupción externa varía entre 3 y 3,75 ciclos de instrucción, lo que representa en nuestro caso (con un Xtal de 20Mhz) un tiempo que varía entre 600 nseg y

750 nseg. Es decir, que cuando el *Módulo de control principal* lea el valor del bus de datos del conversor analógico-digital, este estará ya establecido.

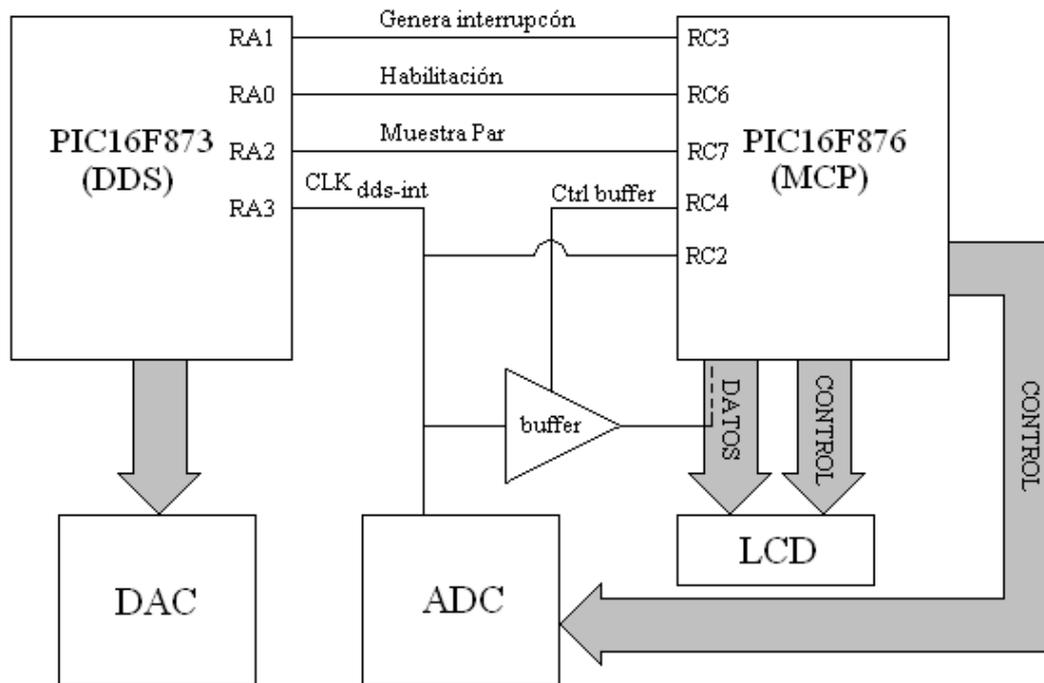


Figura 9: Diagrama simplificado del manejo de los buses

Resumiendo lo hasta ahora expuesto en este punto: el *Módulo de control principal* habilita al *DDS* para que empiece a generar la señal que, en asociación con el *OTA*, se inyectará en el cuerpo, así como también le indica que será él quien genere la señal de adquisición para el conversor analógico-digital y de interrupción externa para el mismo *Módulo de control principal*. En dos determinados instantes de la señal generada, el *Microcontrolador II* setea el valor correspondiente de la línea *MuestraPar* y genera la adquisición al mismo tiempo que le informa mediante la interrupción externa al *Módulo de control principal* que el valor de la muestra adquirida ya se encuentra en el bus de datos del conversor analógico-digital. Recordamos que esto no es un problema debido a la velocidad de respuesta mucho mayor del conversor respecto del *Microcontrolador II*. Seguidamente el *Módulo de control principal* saca del estado de alta impedancia al bus de datos del conversor analógico-digital, lee su contenido y lo acumula en la variable *li_suma_par* si el valor de la línea *MuestraPar* es alto, caso contrario lo acumula en la variable *li_suma_impar*.

Este proceso se repite hasta obtener 65535 valores de muestra par y 65535 valores de muestra impar, para luego promediar los mismos consiguiendo así una considerable exactitud.

Como se mencionó en la sección “*Método Empleado*”, los tiempos en los cuales se adquieren los dos valores de la tensión generada sobre la carga (resistencia patrón o paciente) y se genera la interrupción externa sobre el *Módulo de control principal*, son $t_1 \approx 0.8 \mu\text{seg}$ y $t_2 \approx 10.4 \mu\text{seg}$.

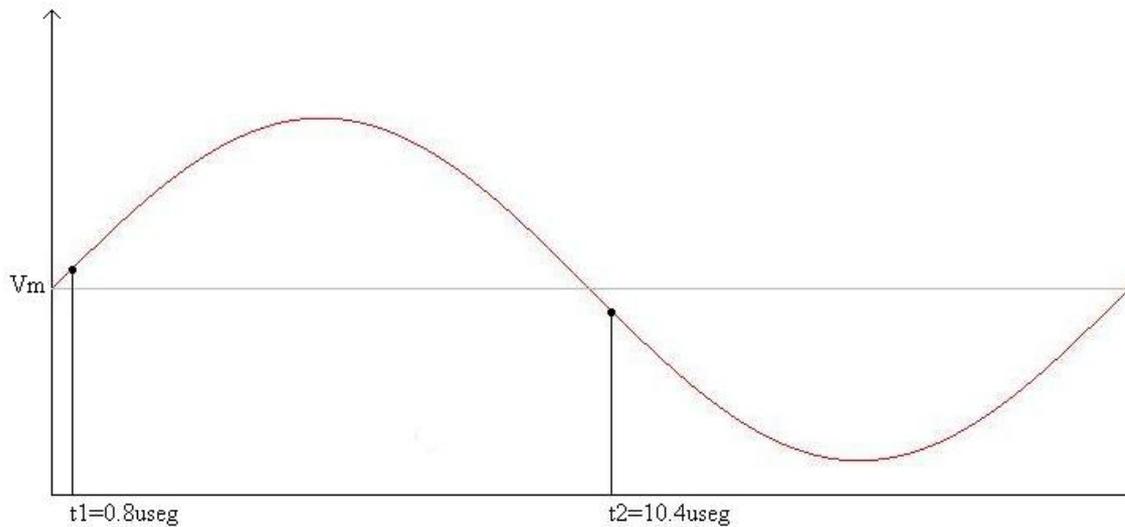


Gráfico 6

Cabe destacar que se debió implementar submuestreo debido a que los $9.6 \mu\text{seg}$ entre las dos muestras no era suficiente para que el *Módulo de control principal* ejecutara la rutina de interrupción, por lo que se optó por tomar las muestras par e impar separadas cuatro períodos de la señal generada ($80 \mu\text{seg}$).

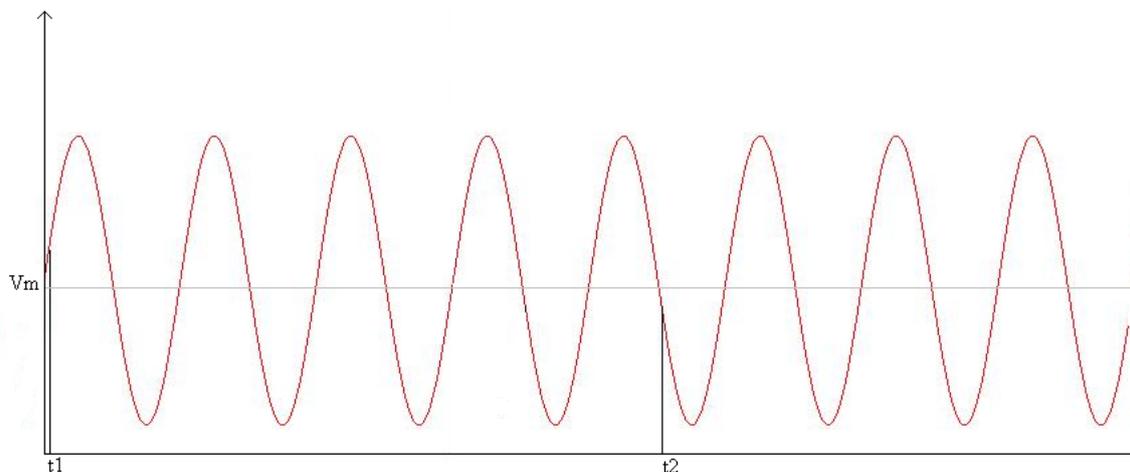


Gráfico 7

- d) Luego de haberse obtenido los valores promediados de N_1 y N_2 (valores de las muestras par e impar), resta obtener el valor de la continua presente en la señal que llega al convertor analógico-digital. Como se explicó en la sección “*Método empleado*” esta continua se suma a la señal levantada del cuerpo con el solo propósito de llevar dicha señal al rango óptimo a la entrada del convertor analógico-digital. A pesar que el valor de continua puede ajustarse a través de un preset, pequeños corrimientos de esta variable ocasionan

errores considerables principalmente en el cálculo de la fase de la impedancia corporal, debido a ello el equipo también realiza los siguientes pasos para computar esta magnitud

- e) El *Módulo de control principal* establece el valor de la línea *GenerarInt* en bajo, con lo cual el *Microcontrolador II* sigue recorriendo su tabla en memoria ROM para generar la senoidal, sin embargo ya no genera la señal de adquisición e interrupción por la línea $Clk_{DDS-INT}$, poniendo esta línea en estado de alta impedancia. También el *Módulo de control principal* pone en bajo la línea *CtrlBuffer*, ya que ahora los pulsos que comandan al conversor analógico-digital son generados por el mismo *Módulo de control principal* a través de la línea Clk_{BLA} , y si el *buffer* estuviese habilitado se produciría una colisión entre dicha línea y la línea de menor peso del bus de datos. Se deshabilitan las interrupciones.

El *Módulo de control principal* acumula en una variable en RAM 65535 muestras de la senoidal generada sobre la carga para calcular por medio de la promediación, su componente de continua (V_m). Luego de obtener esta magnitud, el equipo se encuentra en condiciones de calcular la fase del sistema a través de la ecuación indicada en la sección “*Método empleado*”:

$$\theta = \arctg \left(\frac{(N_2 - V_m) * \text{sen}(w * t_1) - (N_1 - V_m) * \text{sen}(w * t_2)}{(N_2 - V_m) * \text{cos}(w * t_1) - (N_1 - V_m) * \text{cos}(w * t_2)} \right)$$

Donde N_1 =Valor promediado de la muestra par
 N_2 =Valor promediado de la muestra impar
 V_m =Valor de continúa

La amplitud A es un dato que se necesita para obtener tanto la resistencia R como la reactancia capacitiva X_c de la impedancia corporal. Como ya se mencionó en la sección “*Método empleado*”:

$$A = \frac{(N_1 - V_m)}{\text{sen}(w * t_1 - \theta)}$$

$$R = \frac{A * \frac{(3.26 - 1.55)}{255} + 1.55}{0.8mA}$$

$$X_c = R * \text{tg}(\theta)$$

El cálculo de la función $\arctg()$ no presentó mayores inconvenientes, sin embargo las funciones $\text{sen}()$ y $\text{tg}()$ provistas por el software de programación y utilizadas en el cálculo de A y X_c demandaban una cantidad excesiva de memoria de programa por lo que se optó por calcular dichas funciones a través de sendas tablas grabadas en la memoria EEPROM del microcontrolador.

Se estableció que el rango de θ va desde los 2 a los 10 grados, con una resolución de 0.1 grado, originando 81 valores posibles para cada una de las funciones, con lo que se utilizaron 162 posiciones de las 256 disponibles en la EEPROM.

De esta forma se obtienen todos los parámetros necesarios (θ , R y Xc) para la estimación del TBW , FM y FFM , sin embargo ha de notarse que todos los pasos hasta ahora descritos se han realizado sobre la resistencia patrón. Esto se hace fundamentalmente para obtener el valor de θ que introduce el propio sistema, y de esa forma realizar la corrección correspondiente luego de medir la impedancia corporal del paciente.

4. Luego de haber obtenido la fase intrínseca del sistema, el *Módulo de control principal* conecta a la salida del OTA al paciente y repite todos los pasos descritos en el punto 3 para obtener nuevamente θ , R y Xc . Estos dos últimos valores representan la resistencia y la reactancia capacitiva de la impedancia corporal, sin embargo para obtener la fase del mismo se resta de la última fase calculada, la propia del sistema (obtenida en el punto 3). En esta instancia se calculan los resultados del análisis de acuerdo a las siguientes ecuaciones

$$TBW=0.377 * (\text{altura}^2/R) + 0.14 * \text{peso} - 0.08 * \text{edad} + 2.9 * \text{sexo} + 4.65 \text{ (Lukaski)}$$

$$FFM=-4.104 + 0.518 * (\text{altura}^2/R) + 0.231 * \text{peso} + 0.13 Xc + 4.229 * \text{sexo}$$

(Kyle), [2]

$$FM= \text{peso} - FFM$$

R =resistencia; Xc =reactancia; $\text{sexo}=1$ (si es masculino) $=0$ (si es femenino).

5. Se muestran los valores de TBW , FFM y FM , luego de lo cual el usuario puede optar por iniciar un nuevo estudio o ver nuevamente los resultados del estudio previamente realizado.

MANUAL DE MANTENIMIENTO

ESPECIFICACIONES GENERALES

ALIMENTACIÓN: Dos baterías de 9V.

CABLES: Blindados, símil cable de micrófono.

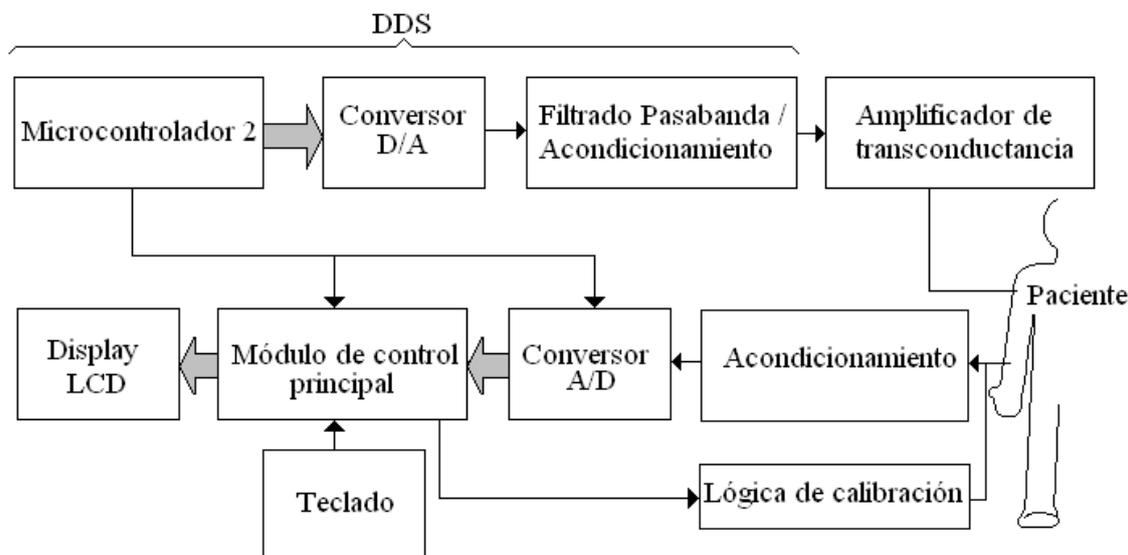
ELECTRODOS: Autoadhesivos, descartables, preferentemente de plata/cloruro de plata, como los usados para electrocardiograma.

FRECUENCIA DE TRABAJO: La señal que este dispositivo genera es de 50KHz y 0.8mA rms.

FRECUENCIA DE OSCILACIÓN DE LOS MICROCONTROLADORES: Verificar 20MHz en las patas 9 y 10 de cada uno.

BREVE DESCRIPCIÓN DEL FUNCIONAMIENTO

El siguiente diagrama muestra el funcionamiento básico del equipo.



Módulo de control principal

Implementado a partir de un microcontrolador PIC16F876 de Microchip. Tiene la función de controlar a los restantes bloques digitales que conforman el equipo y de realizar los cálculos para obtener los resultados del examen (TBW, FFM, FM). Debido a que dichos cálculos son en punto flotante, se optó por desarrollar el software bajo una distribución del lenguaje de programación "C" para microcontroladores PIC perteneciente a la empresa CCS®, denominada PIC-C.

Microcontrolador 2

Este dispositivo es un Microcontrolador PIC16F873 de Microchip, que opera a una frecuencia de clock de 20MHz, lo que le da una frecuencia de instrucción de 5 MHz. Cumple la función de llevar a uno de sus puertos, los valores en formato binario que corresponden a una senoide de 50KHz. Asimismo debe encargarse de la lógica involucrada para su posterior adquisición y procesamiento. En función de ello y en simultáneo con la generación de la forma de onda, levanta, durante intervalos perfectamente definidos, la pata RA3 mediante la cual ordena al conversor A/D que convierta y a su vez le indica al segundo microcontrolador que los valores de la conversión ya están disponibles. Esto último es posible debido a que, el tiempo que le lleva al microcontrolador 2 atender su rutina de interrupción es suficiente para que el conversor convierta y tenga listos los valores a entregar. **VER figura gral**

Conversor Digital a Analógico (DAC)

Se trata del conversor DAC0800. El mismo recibe los valores que le transfiere el PIC1 en 8 bits para convertirlos a formato analógico. El desempeño de este dispositivo no es crítico, debido a que la generación de armónicas no deseadas es corregida por las etapas de filtrado posteriores.

Filtrado

La señal generada por el DAC debe ser filtrada para llegar a una señal limpia de armónicas. Para ello se aplica un filtro pasabajos Butterworth de orden 2, con $f_c=100\text{KHz}$. La incorporación del segundo amplificador operacional (TL081) tiene como objetivo la corrección de cualquier offset que pudiera necesitar la etapa de amplificación de transconductancia.

Amplificador de transconductancia

La señal que debe inyectarse en el cuerpo es de $800\mu\text{A rms}$. El LM3080 es el encargado de convertir la senoide de tensión a una de corriente, de amplitud y frecuencia bien definida. Asimismo debe conservar este valor inalterable, sin importar la impedancia con la que se lo esté cargando.

Adquisición y acondicionamiento

Amplificador de instrumentación (INA 114)

Este integrado, de los denominados amplificadores de instrumentación, cuenta con una impedancia de entrada muy alta ($10^6\Omega$), un offset muy pequeño (del orden de las decenas de μV) y un rechazo a modo común cercano a los 120dB. Estas características lo hacen óptimo para la medición de señales corporales y forma parte del circuito con el cual se adquiere la señal de tensión. Ver circuito general completo

Acondicionamiento

Debido a que el rango del conversor AD es de 1,58 a 3,33 Volts, una vez adquirida la señal de tensión debe ajustarse de forma tal de poder ser convertida a formato digital . Para ello agrega un offset a la señal adquirida y se la modifica en amplitud.

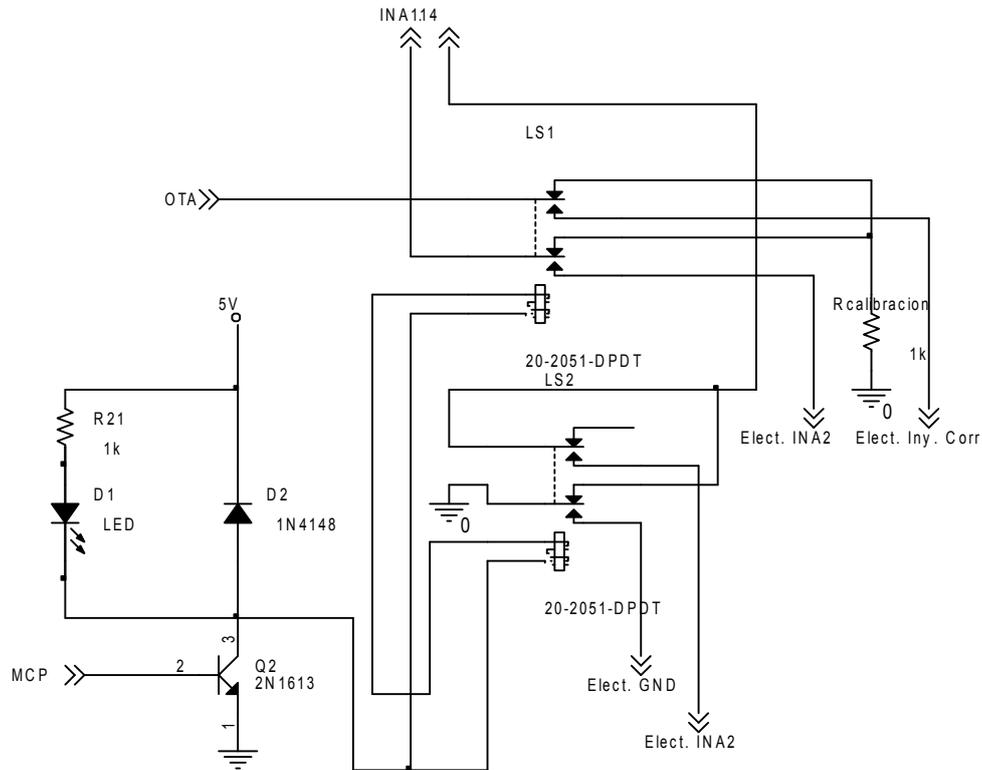


Figura 10

Relés de calibración

Previo al comienzo de cada examen, el equipo lleva a cabo una rutina de autocalibración, la cual consiste en la medición de una resistencia patrón para obtener del mismo la fase propia que introduce la el sub-sistema *DDS + OTA* y tenerla en cuenta en la medición del paciente, disminuyendo así errores por posibles derivas por temperatura o envejecimiento de los componentes. Para ello consta de un arreglo de dos relés que son comandados por el *Módulo de control principal*, y que primeramente conectan la resistencia patrón para obtener a través de ella la fase intrínseca del sistema, para luego sí conectar al paciente para realizar el análisis. Ver figura10

Conversor analógico-digital (ADC)

Cumple la función de digitalizar la señal entregada por la etapa de acondicionamiento para ingresarla al *Módulo de Control Principal*. Se utilizó un CI *TDA8703 (flash)*, debido fundamentalmente a su muy pequeña ventana de muestreo ($t_{ds} = 2 \text{ nseg}$).

Figura 11

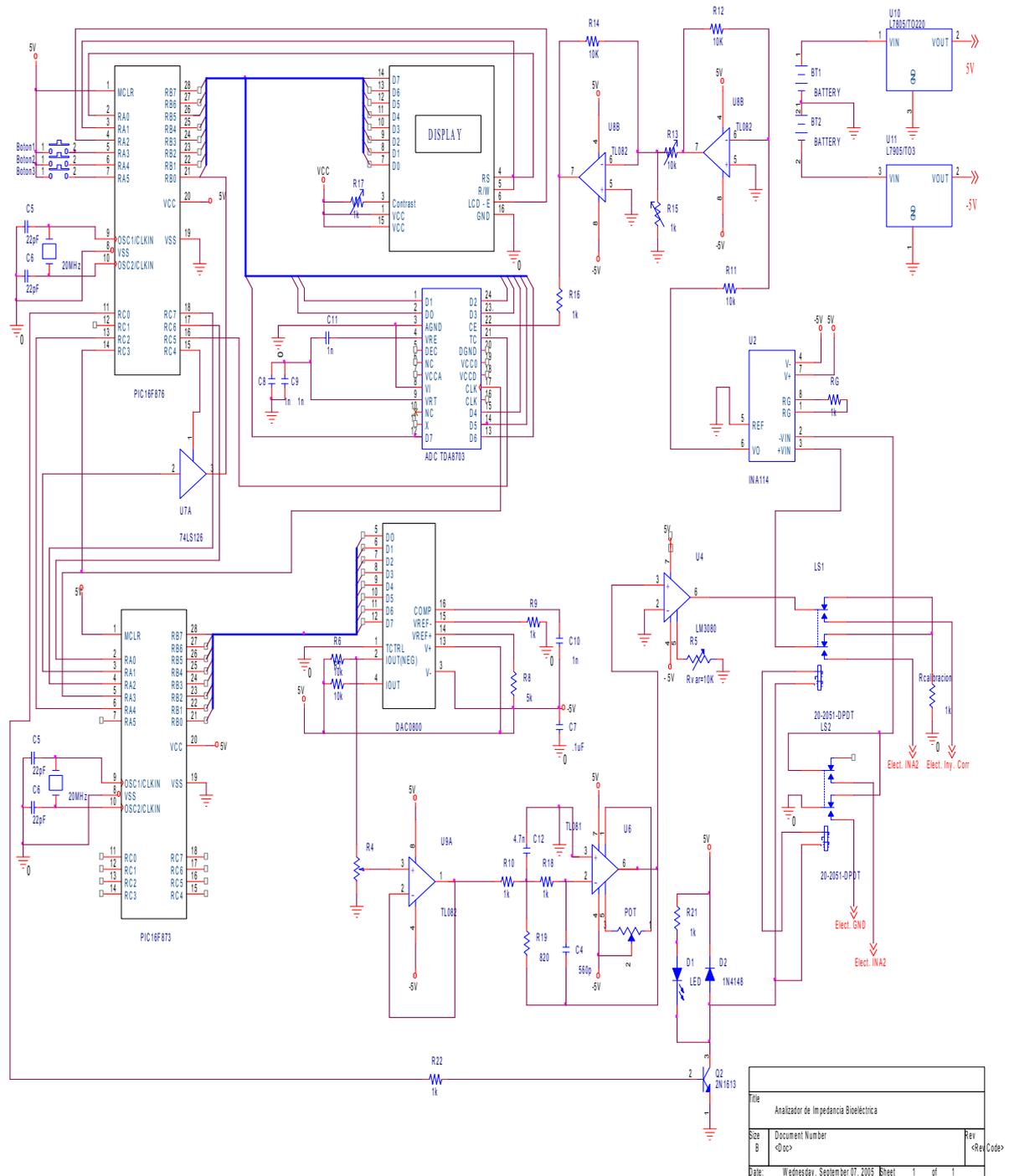
LCD

Posibilita visualizar el ingreso de los datos del paciente así como también informa los resultados del examen.

TECLADO

Consta de tres botones: ENTER, TAB y MODIFICAR. Mediante los mismos se ingresan los datos del paciente (edad, sexo, altura y peso) y se inicia el comienzo del análisis.

Todos las etapas explicadas anteriormente corresponden al siguiente esquemático.



PROCEDIMIENTO PARA LA LOCALIZACIÓN DE FALLAS

Para localizar posibles fallas en el funcionamiento del equipo, se recomienda seguir los pasos que se detallan a continuación.

- Carga de las baterías
- Estado de los electrodos y cables.

Si los ítems anteriores se encuentran en buen estado, se deberá verificar la circuitería de instrumento de la siguiente manera:

- Oscilación de los Microcontroladores: Senoidales de 20 MHz (clock) en los pines 9 y 10 de los PIC16f873 y PIC16F876.
- Seguimiento de la señal sintetizada.

Debe hacerse según lo detallado en el diagrama1, las referencias a las tensiones de alimentación deben hacerse según se muestra en la página 28.

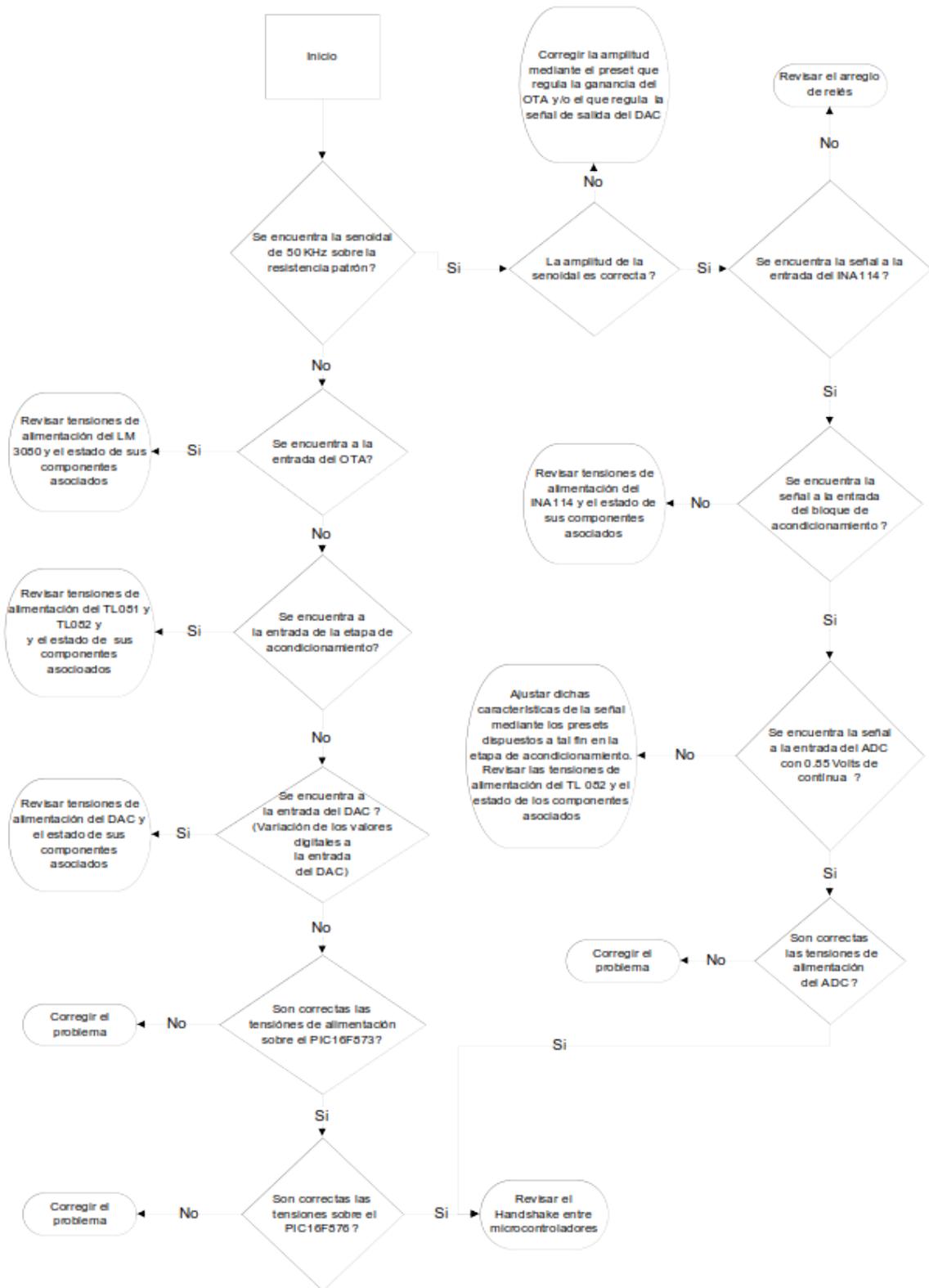


Diagrama 1

TENSIONES DE ALIMENTACIÓN:

LM7805

+5V: pata 3

LM7905

-5V: pata 2

PIC16F876 y PIC16F873

+5V: pata 20, pata 1 (MCLR)

GND: pata 8 y 19

74LS126

+5V: pata 14,

GND: pata 7.

LCD

+5V: pata 2

GND: pata 1

TL082

+5V: pata 4

-5V: pata 8

TL081

+5V: pata 7

-5V: pata 4

INA114

+5V: pata 7

-5V: pata 4

TDA 8703

+5V: pata 7, pata 19

GND: pata 20, pata 3

DAC800

+5V: pata 13

-5V: pata 3

El analizador de impedancia bioeléctrica es un instrumento destinado a la medición de la composición corporal. Su utilización es sumamente sencilla ya que se calibra en forma automática antes de cada medición y no precisa interfaz con otros dispositivos.

ALIMENTACION

- Este dispositivo funciona con 2 baterías de 9 Volts.

QUÉ MIDE EL ANALIZADOR

- El valor de la **Resistencia corporal (R)**, es decir la oposición que este le presenta al paso de corriente. Este parámetro depende de soluciones electrolíticas intracelulares y extracelulares.
- El valor de la Reactancia corporal (Xc). Toda célula metabólicamente activa tiene un potencial interno que varía en función de su tipo entre -50 y -100 mV y está separada de los fluidos extracelulares por una membrana aislante. Esta característica hace que la célula actúe frente al paso de corriente como una capacidad, es decir, almacenando carga eléctrica. Este hecho se cuantifica mediante su reactancia y su ángulo de fase (θ) que es medido por este instrumento.

Con estos dos valores, y el ingreso por teclado de: edad, peso, sexo y altura, el analizador estima la composición corporal, diferenciando en:

TBW (Total body Water): Agua total corporal.

FFM (Fat free mass): Masa libre de grasa o tejido magro.

FM(Fat mass): Materia grasa.

PASOS PARA LA MEDICIÓN

PREPARACION DEL PACIENTE

Para una medición precisa es importante tener en cuenta los siguientes lineamientos:

- No haber tenido actividad física dentro de las 8 horas.
- Abstenerse del consumo de alcohol durante las 12 horas previas al análisis.
- Mantenerse acostado y quieto durante la medición.
- Encontrarse seco de transpiración.
- No presentar fiebre.

PROCEDIMIENTO DE MEDICION

- El paciente debe recostarse sobre una superficie no conductora y descubrirse la muñeca y el pie del mismo lado. En general, está estandarizado el lado derecho aunque puede hacerse el estudio en ambos lados..

- El paciente debe mantener la horizontalidad separando los brazos unos treinta grados del tronco y con las piernas separadas entre si. Si tuviera anillos, collares u otro accesorio metálico es conveniente retirarlos.
- Se debe explicar al paciente los fundamentos del análisis haciéndole notar que la corriente inyectada es imperceptible.
- Ubique los electrodos según lo indica la figura 12. Limpiando con alcohol las zonas de contacto de los electrodos con la piel.



Figura 12

- Encienda el aparato
- En el display aparecerá el cartel : Analizador de impedancia bioeléctrica. Presione ENTER.
- Introduzca la edad.



Figura 13

Para modificar el valor donde se encuentra el cursor, presione la tecla CAMBIAR. Para pasar al carácter siguiente presione TAB. Una vez ingresado el ítem presione ENTER.

- En forma análoga ingrese: edad, sexo, peso y altura.
- Una vez ingresados los datos elija iniciar el test: SI, o si desea corregir algún valor, elija NO y modifique el parámetro que desee.

El instrumento se calibrará utilizando una resistencia patrón interna y luego comenzará el análisis. Una vez que lo haya terminado, le presentará en pantalla:

R: Resistencia corporal

θ : Angulo de fase.

TBW (Total body Water)

FFM (Fat free mass)

FM(Fat mass).

Luego de mostrar los parámetros el instrumento le ofrecerá la opción de repetir el test. Si decide hacerlo no es necesario reingresar los datos, aunque puede modificarlos si lo desea. Si desea terminar el análisis, puede apagar el equipo. Retire los electrodos y deséchelos.

MEDICIONES

Para verificar el funcionamiento del equipo, se cotejaron los valores obtenidos con juegos de resistencias y capacitores conocidos cuyos valores se encontraban dentro del rango de valores esperables para una impedancia corporal (debe recordarse que el rango de impedancias reconocidas por el equipo está limitado). Se hicieron sucesivas mediciones bajo las mismas condiciones (arreglo, conectores, alimentación, temperatura, lugar) obteniéndose una dispersión muy pequeña. La figura 14 muestra 20 mediciones sucesivas hechas sobre un RC paralelo con $R=499$ y $C=47\text{nf}$ ($\theta=7.71^\circ$).

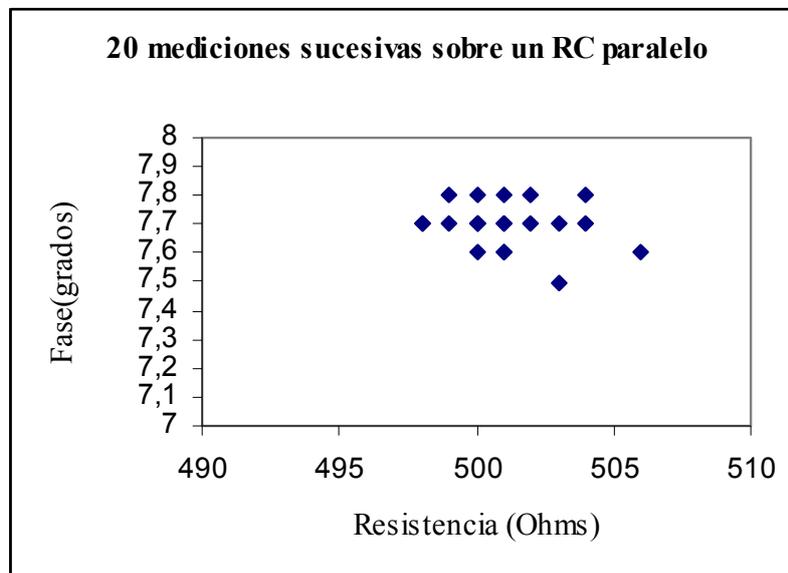


Figura 14

Estos valores presentan una media de 501,09 para la resistencia y 7,69 para la fase. Mientras que la desviación estándar es de 2.047 y 0.0804 respectivamente. Para el caso de mediciones realizadas sobre un paciente, se verificó un incremento en la dispersión de las medidas, lo que era esperable ya que se agregan mayores factores de variabilidad a la medición (ubicación de los electrodos, mayor ruido, etc)

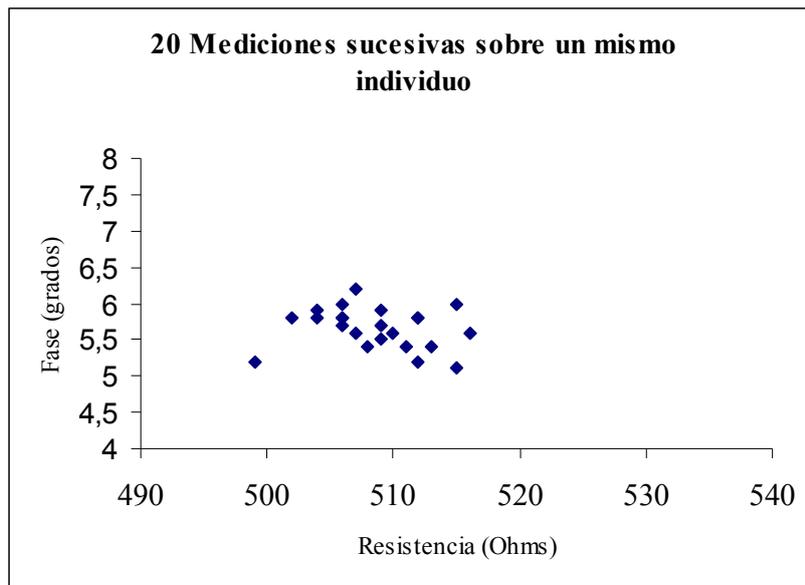


Figura 15

De los valores de la figura 15 obtiene una desviación estándar de 4,31 para la resistencia y 0,28 para la fase.

Puede apreciarse que los resultados obtenidos siguen siendo más que aceptables y la variación que representan es aún menor para el cálculo del TBW y FFM (menor al 1%). Ver ecuaciones utilizadas en la descripción de funcionamiento (pag. 21)

Es importante recalcar la importancia en la ubicación y fijación de los electrodos. Una mala adherencia, es la principal causa de errores en la medición, lo que se advierte en la bibliografía utilizada [1],[8],[20] y se pudo verificar en la práctica. La ubicación de los electrodos también resulta de gran importancia en la repetibilidad del análisis. Se pudo comprobar que un desplazamiento de los mismos entre 1 - 2 centímetros se traducían en una variación de la resistencia de un 2% aproximadamente.

CONCLUSIONES

El equipo construido es portátil, de bajo costo y no requiere comunicación con ningún otro dispositivo para realizar su análisis.

Los fundamentos abordados en este informe son en su mayoría aplicables a otros instrumentos en los que sea de interés conocer la impedancia del cuerpo o de alguna zona en especial del mismo (monitores multiparamétricos, electroglotógrafos, medidores de impedancia para desfibriladores, etc).

El manejo simultáneo y efectivo de los dos microcontroladores ha sido el mayor desafío del proyecto y su logro es el responsable del buen funcionamiento del equipo.

El método empleado ha permitido obtener resultados con una desviación pequeña y una resolución de 0.1 grado para la fase y 1Ω para la resistencia, los cuales son suficientes para la totalidad de los propósitos en los cuales conocemos, se aplica actualmente este análisis.

De esta forma se pone a disposición del especialista en nutrición u otro profesional del área de la medicina un instrumento de manejo sencillo e intuitivo, económico y que brinda un análisis rápido y efectivo de la composición corporal.

FUTURAS MEJORAS

El consumo del equipo es relativamente elevado, un juego de baterías dura unas decenas de análisis. Podrían implementarse diversas soluciones a este problema, como la de agregar un sistema que recargue las baterías del circuito mientras se encuentra en desuso, aplicar un conversor DC-DC o modificar el aislamiento del aparato según las normas vigentes haciéndolo apto para su alimentación a través de la línea de 220V.

En la bibliografía recopilada para el desarrollo de este proyecto, aparecen dispositivos de medición *foot to foot* (de pie a pie), los cuales efectúan el análisis de impedancia simplemente con el paciente parado encima del instrumento, sobre dos contactos metálicos (símil balanza de baño). Esta técnica resulta muy cómoda, ya que entre otras cosas, evita el uso de electrodos, aunque ciertamente, existe escepticismo respecto de su implementación con fines clínicos. La aplicación de esta técnica con la plataforma del analizador descrito en este informe es viable, y la adaptación constaría principalmente del reemplazo de los algoritmos para la obtención de impedancia (contemplación de la resistencia serie de la interfase electrodo piel).

Otra mejoría propuesta es la implementación de la técnica de medición multifrecuencia. Para la misma, dada la gran versatilidad de los microcontroladores programables, bastaría con modificar la generación digital de la señal (software del DDS) y reemplazar la etapa posterior de filtrado por alguna otra que permita el desplazamiento de la frecuencia central de paso.

BIBLIOGRAFÍA

1. L Piccoli, , D Nescolarde y J Rosell. *Análisis convencional y vectorial de bioimpedancia en la práctica clínica*. Nefrología Vol XXII Número 3. 2003
2. U Kyle, L Genton, L Karsegard, D Slosman, C Richard. *Single Prediction Equation for bioelectrical impedance análisis in adults aged 20-94 years*. Applied Nutritional investigation, Nutrition Volume 17 number 3, 2001.
3. S Sun, W Cameron, S Heymsfield, H Lukaski, D Schoeller, K Friedl, R Kuczmarsyi, K Flegal, C Johnson, V Hubbard. *Development of bioelectrical impedance analysis prediction equations for body composition with the use of a multicomponent model for use in epidemiologic surveys*. American Society for Clinical Nutrition, 2003.
4. R Y T Sung, P Lau, C W Yu, P K W Lam, E Nelson. *Measurements of body fat using leg to leg bioimpedance*. Arch Dis Child 2001; 85:263-267 (September).
5. L C Ward, B L Heitmann. *Assessment of boy composition by bioelectrical impedance analysis without the need for measurement of height*. Clinical Nutrition (2001) 20(1)9:21-26.
6. John S. James. *Bioelectrical Impedance Analysis (BIA) May Predict AIDS Survival*. 16/June/1995. AIDS Treatment News Archive.
7. Nuñez C, Gallagher D, Visser M, Pe-Sunyer FX, Wang Z, Heymsfield SB. *Bioimpedance analysis: Evaluation of leg-to-leg system based on pressure contact foot-pad electrodes*. Medicine and Science in Sports and Exercise. 1997.
8. R. Dörhöfer, M. Pirlich. *The BIA compendium 1st edition 07/2002*
9. Rudolph J. Liedtke *The Fundamentals of Bioelectrical Impedance Analysis*. February 1,1998.
10. Donald P Kotler, Santiago Burastero, Jack Wang, and Richard N Pierson Jr. *Prediction of body cell mass, fat free mass and total body water with bioelectrical impedance analysis: effects of race, sex, and Disease*.
11. J.Peterson. *Abstract:How Should BIA Be Performed, and How Can BIA Measurements Be Standardized*
12. CD de la cátedra: *Adquisición Digital de señales*. Versión 2004.
13. J. Farmer. *A real time operating system for PICmicro Microcontrollers*. Microchip Inc. 2002.
14. B. Negley. *Questions and answers concerning serial EEPROMs*. Microchip Technology incorporated, 1994.
15. *PICmicro Mid Range MCU Family Reference Manual*. . Microchip Technology incorporated, 2002.
16. *PIC16F87X Data Sheet 28/40-Pin 8 bit CMOS FLASH Microcontrollers*. . Microchip Technology incorporated, 2001.
17. *MPLAB IDE Simulator, Editor User`s guide*. Microchip Technology incorporated, 2002.
18. M. McQuilken, M. Glenewinkel. *Semiconductor application Note: Arithmetic waveform synthesis with the HC05/08 MCUs*. Motorota Inc, 1993.

- 19 *MPLAB IDE v6.xx Quick start Guide*. Microchip Technology incorporated, 2003.
- 20 *Measuring the ultimate outcome with Bioelectrical impedance analysis*. Nutrition Fact Serv. RJI Systems

ANEXO 1: Código

Código Módulo de Control Principal (C)

```
#include "d:\Proyecto\Codigo_PIC\BIA\BIA.h"
#include "d:\Proyecto\Codigo_PIC\BIA\STDLIB_1.H"
//Grabo en la EEPROM los vbalores del Seno y la Tangente

//Las primeras 81 posiciones de la EEPROM corresponden a la función seno, con el argumento en radianes y variando entre
0.4677482395 y 0.3281218993
//Es usada para obtener la amplitud de la senoidal mediante la funcion  $A=(n1-ValorMedio)/\text{sen}(K1-FI)$ , donde
 $K1=2*\pi*50000*1.6$ useg=0.50265482457
//y FI varia entre 2 y 10 grados con saltos de 0.1 grado

//Las segundas 80 posiciones de la EEPROM contienen la funcion tangente para valores del argumento variando entre
0.0349065850398 y 0.174532925199
//Se usa para obtener Xc teniendo R y FI a traves de  $Xc=R*\text{tg}(FI)$ , teniendo FI el rango y definicion ya mencionados

#rom
0x2100={0x52,0x53,0x53,0x53,0x54,0x54,0x55,0x55,0x56,0x56,0x56,0x57,0x57,0x58,0x58,0x58,0x59,0x59,0x5A,0x5A,0x5B,0x
5B,0x5B,0x5C,0x5C,0x5D,0x5D,0x5D,0x5E,0x5E,0x5F,0x5F,0x60,0x60,0x60,0x61,0x61,0x62,0x62,0x63,0x63,0x64,0x64,0
x64,0x65,0x65,0x66,0x66,0x66,0x67,0x67,0x68,0x68,0x69,0x69,0x69,0x6A,0x6A,0x6B,0x6B,0x6B,0x6C,0x6C,0x6D,0x6D,0x6D
,0x6E,0x6E,0x6F,0x6F,0x70,0x70,0x71,0x71,0x71,0x72,0x72,0x73,0x73,0x9,0x9,0xA,0xA,0xB,0xB,0xC,0xC,0xD,0xD,
0xE,0xE,0xF,0xF,0x10,0x10,0x10,0x11,0x11,0x12,0x12,0x13,0x13,0x14,0x14,0x15,0x15,0x15,0x16,0x16,0x17,0x17,0x18,0x18,0
x19,0x19,0x19,0x1A,0x1A,0x1B,0x1B,0x1C,0x1C,0x1D,0x1D,0x1E,0x1E,0x1E,0x1F,0x1F,0x20,0x20,0x21,0x21,0x22,0x22,0x22
,0x23,0x23,0x24,0x24,0x25,0x25,0x26,0x26,0x27,0x27,0x27,0x28,0x28,0x29,0x29,0x2A,0x2A,0x2B,0x2B,0x2C,0x2C,0x2D,0x2
D};

#use fast_io(a)
#use fast_io(b)
#use fast_io(c)

#define lcd_data portb //LCD data lines interface
#define lcd_data_tris trisb
#define lcd_ctrl porta //LCD control lines interface
#define LCD_E 0 //LCD Enable control line
#define LCD_RW 1 //LCD Read/Write control line
#define LCD_RS 2 //LCD Register-Select control line
#define muestra_par PIN_C7 // Para leer desde el DDS si la muestra es par o impar
#define ENTER PIN_A5
#define TAB PIN_A4
#define CHANGE PIN_A3
#define li_cant_muestras 65535
#define li_cant_muestras_continua 65535

#bit int_flag=0x0B.1 //0x0B=intcon
#bit conv_ce = 0x07.5
#bit habil_dds=0x07.6 //RC6 para indicarle al DDS que empiece a generar
#bit ADQ=0x07.3 //RC3 para indicarle al DDS si debe generar la señal de adquisición o poner esa pata como entrada para generarla
desde el BIA
#bit CONV_CLK=0x07.2 // RC2 genera el pulso para la adquisicion
#bit rele = 0x07.0 //Comanda el relé
#bit int_timer_flag=0x0B.2 //0x0B=intcon
#bit rele = 0x07.0
#bit habil_buffer=0x07.4 //RC4 habilita el buffer donde está conectada RB0

char ls_sexo='F';
char ls_iniciar='N';
char ls_otro='N';
int li_cent_altura=0;
int li_dec_altura=0;
int li_uni_altura=0;
int li_cent_peso=0;
int li_dec_peso=0;
int li_uni_peso=0;
int li_dec_edad=0;
int li_uni_edad=0;
int li_posicion=1;
int interrumpido=0;
int li_dato_leido=0;
```

```

long li_indice_par=0;
long li_indice_impar=0;
int32 li_suma_par =0;
int32 li_suma_impar =0;
int32 li_suma_continua =0;
long li_indice_continua=0;
float lf_continua=0;

//Variables usadas para guardar el entorno
int save_w;
#locate save_w=0x7f
int save_status;
#locate save_status=0x25
#byte status = 3

//USadas por las rutinas matematicas
#define PI 3.141592654
#define PI_DIV_BY_TWO 1.570796326794896
float const pat[4] = {0.17630401, 5.6710795, 22.376096, 19.818457};
float const qat[4] = {1.0000000, 11.368190, 28.982246, 19.818457};

// int H=0;
int1 S=0;
int W=0;
int E=0;
long R=0;
float FI=0;
float Xc=0;
float A;
float TBW=0;
float FFM=0;
float FM=0;
float n1=0;
float n2=0;
long i;

void USAR_ADC()
{
    set_tris_b(0xff); //PortB como entrada
    habil_buffer=0; //deshabilito el buffer
    CONV_CE=0; //Habilito el ADC
    //Las rutinas del LCD dejan al mismo en HI-z al terminar
}

void USAR_INT()
{
    set_tris_b(0xff); //PortB como entrada
    CONV_CE=1; //Por las dudas pone al ADC en Hi-Z
    habil_buffer=1; //Habilito el buffer
    //Las rutinas del LCD dejan al mismo en HI-z al terminar
}

void USAR_LCD()
{
    // output_bit(PIN_C5,1); //Por las dudas pone al ADC en Hi-Z
    set_tris_b(0x00); //Pone el puerto_b como salida
    habil_buffer=0; //deshabilito el buffer
    CONV_CE=1; //Por las dudas pone al ADC en Hi-Z
    //Las señales de habilitacion se manejan dentro de las rutinas del LCD
}

//;Tests if the LCD is busy. Returns when LCD busy-flag is inactive.
void LCDBUSY()
{
    int li_dato=1;
    set_tris_b(0xff); //Set PORTB for input
    while(li_dato==1) //Check Busy flag, High = Busy
    {
        output_bit( PIN_A2, 0); //Set LCD for command mode
        output_bit( PIN_A1, 1); //Setup to read busy flag
        output_bit( PIN_A0, 1); //LCD E-line High
        li_dato=input(PIN_B7); //Read busy flag + DDram address
        output_bit( PIN_A0, 0); //LCD E-line Low
    }
    output_bit(PIN_A1,0);
}

```

```

        set_tris_b(0x00); //Set PORTB for output
    }

void LCDPUTCMD(int li_comando)
{
    LCDBUSY();//Wait for LCD to be ready
    output_bit( PIN_A1, 0);
    output_bit( PIN_A2, 0);
    output_bit( PIN_A0, 1);
    output_b(li_comando);
    output_bit( PIN_A0, 0);
}

//Clears display and returns cursor to home position (upper-left corner).
void LCDCLEAR()
{
    LCDPUTCMD(0x01);
}

//Returns cursor to home position.
//Returns display to original position (when shifted).
void LCDHOME()
{
    LCDPUTCMD(0x02);
}

//Sends character to LCD
//Required character must be in W
void LCDPUTCHAR(int lc_mostrar)
{
    LCDBUSY(); //; Wait for LCD to be ready
    output_bit(PIN_A1,0); //; Set LCD in read mode
    output_bit(PIN_A2,1); //Set LCD in data mode
    output_bit(PIN_A0,1); //LCD E-line High
    output_b(lc_mostrar);
    output_bit(PIN_A0,0); //LCD E-line Low
}

// Sets the Display-Data-RAM address. DDRAM data is read/written after this setting.
// Required entry mode must be set in W
// b0-6 : required DDRAM address
// b7 : don't care
void LCDSDDA(int li_direccion)
{
    bit_set(li_direccion,7);
    LCDPUTCMD(li_direccion);
}

void LCDINIT ()
{
    //Busy-flag is not yet valid
    //ALL PORT output should output Low.
    USAR_LCD();
    output_a(0);//limpio el puerto de control
    delay_ms(15);
    LCDPUTCMD(0x38); //8-bit-interface, 2-lines
    delay_ms(15);
    LCDPUTCMD(0x38);
    delay_ms(15);
    LCDPUTCMD(0x38);
    LCDPUTCMD(0x38); //Function set
    LCDPUTCMD(0x08); //Display off
    LCDCLEAR();
    //Sets entry mode of the LCD
    //Required entry mode must be set in W
    //b0: 0 = no display shift, 1 = display shift
    //b1: 0 = auto-decrement, 1 = auto-increment
    //b2 : 1
    //b3-b7 : 0
    LCDPUTCMD(0x06);
    //Sets display control
    //Required entry mode must be set in W
    //b0: 0 = cursor blink off, 1 = cursor blink on (if b1 = 1)
    //b1: 0 = cursor off, 1 = cursor on
    //b2: 0 = display off, 1 = display on (display data remains in DD-RAM)
    //b3: 1
    //b2-b7 : 0
    LCDPUTCMD(0x0C);
    LCDHOME();
}

```

```

    }
void CONV_BCD_MOSTRAR( float li_dato_mostrar, int li_clear )
{
int li_dato1=0;
int li_centenas=0;
int li_decenas=0;
int li_unidades=0;
int li_decimas=0;
if (li_dato_mostrar >=100)
    {
    li_centenas=li_dato_mostrar / 100 ;
    li_dato1 = li_dato_mostrar - (li_centenas * 100) ;
    if (li_dato1 >=10)
        {
        li_decenas = li_dato1 / 10;
        li_unidades = li_dato1 - (li_decenas * 10);
        }
    else
        {
        li_unidades = li_dato1;
        }
    }
else
    {
    if (li_dato_mostrar >=10)
        {
        li_decenas = li_dato_mostrar / 10;
        li_unidades = li_dato_mostrar - (li_decenas * 10);
        }
    else
        {
        li_unidades=li_dato_mostrar;
        }
    }
    li_decimas=(li_dato_mostrar-li_centenas*100-li_decenas*10-li_unidades)*10+0.01;
li_centenas=li_centenas + 0x30;
li_decenas=li_decenas + 0x30;
li_unidades=li_unidades + 0x30;
li_decimas=li_decimas + 0x30;
if (li_clear)
    {
    LCDCLEAR();
    LCDHOME();
    }
LCDPUTCMD(0b00001100);
LCDPUTCHAR(li_centenas);
LCDPUTCHAR(li_decenas);
LCDPUTCHAR(li_unidades);
LCDPUTCHAR('.');
LCDPUTCHAR(li_decimas);
}

```

```

////////////////////////////////////
//      float atan(float x)
////////////////////////////////////
// Description : returns the arctangent value of the value x.
// Date : N/A
//
float atan(float x)
{
float y, res, r;
int1 s, flag;
s = 0;
flag = 0;
y = x;
if (x < 0)
    {
    s = 1;
    y = -y;
    }
if (y > 1.0)
    {
    y = 1.0/y;
    flag = 1;
    }
}

```

```

res = pat[0]*y*y + pat[1];
res = res*y*y + pat[2];
res = res*y*y + pat[3];
r = qat[0]*y*y + qat[1];
r = r*y*y + qat[2];
r = r*y*y + qat[3];
res = y*res/r;
if (flag) // for |x| > 1
    res = PI_DIV_BY_TWO - res;
if (s)
    res = -res;
return(res);
}
}
void SET_PANTALLA( int li_clear,int li_home,int li_blink)
{
if (li_clear==1)
    {
LCDCLEAR();
}
if (li_home==1)
    {
LCDHOME();
}
else
    {
LCDSDDA(0x40);
}
if (li_blink==1)
    {
//Sets display control
//Required entry mode must be set in W
//b0: 0 = cursor blink off, 1 = cursor blink on (if b1 = 1)
//b1: 0 = cursor off, 1 = cursor on
//b2: 0 = display off, 1 = display on (display data remains in DD-RAM)
//b3: 1
//b4-b7 : 0
LCDPUTCMD(0b00001111);}
else
    {
LCDPUTCMD(0b00001100);
}
}
}

void MOSTRAR_PRESENTACION()
{
SET_PANTALLA(1,1,0);
LCDPUTCHAR('A');
LCDPUTCHAR('\n');
LCDPUTCHAR('a');
LCDPUTCHAR('l');
LCDPUTCHAR('i');
LCDPUTCHAR('z');
LCDPUTCHAR('a');
LCDPUTCHAR('d');
LCDPUTCHAR('o');
LCDPUTCHAR('r');
LCDPUTCHAR(' ');
LCDPUTCHAR('d');
LCDPUTCHAR('e');
LCDPUTCHAR(' ');

SET_PANTALLA(0,0,0);
LCDPUTCHAR('B');
LCDPUTCHAR('i');
LCDPUTCHAR('o');
LCDPUTCHAR('i');
LCDPUTCHAR('m');
LCDPUTCHAR('p');
LCDPUTCHAR('e');
LCDPUTCHAR('d');
LCDPUTCHAR('a');
LCDPUTCHAR('n');
LCDPUTCHAR('c');
LCDPUTCHAR('i');
LCDPUTCHAR('a');
}
}

```

```

void MOSTRAR_SEXO()
{
    SET_PANTALLA(1,1,1);
    LCDPUTCHAR('S');
    LCDPUTCHAR('e');
    LCDPUTCHAR('x');
    LCDPUTCHAR('o');
    LCDPUTCHAR(':');
    LCDPUTCHAR(' ');
    LCDPUTCHAR(ls_sexo);
    LCDSDDA(0x06);
}

void MOSTRAR_INICIAR_TEST()
{
    SET_PANTALLA(1,1,1);
    LCDPUTCHAR('I');
    LCDPUTCHAR('n');
    LCDPUTCHAR('i');
    LCDPUTCHAR('c');
    LCDPUTCHAR('i');
    LCDPUTCHAR('a');
    LCDPUTCHAR('r');
    LCDPUTCHAR(':');
    LCDPUTCHAR(' ');
    LCDPUTCHAR(ls_iniciar);
    LCDSDDA(0x09);
}

void MOSTRAR_OTRO_TEST()
{
    SET_PANTALLA(1,1,1);
    LCDPUTCHAR('O');
    LCDPUTCHAR('t');
    LCDPUTCHAR('r');
    LCDPUTCHAR('o');
    LCDPUTCHAR(' ');
    LCDPUTCHAR('a');
    LCDPUTCHAR('n');
    LCDPUTCHAR('a');
    LCDPUTCHAR('l');
    LCDPUTCHAR('i');
    LCDPUTCHAR('s');
    LCDPUTCHAR('i');
    LCDPUTCHAR('s');
    LCDPUTCHAR(':');
    LCDPUTCHAR(' ');
    LCDPUTCHAR(ls_otro);
    LCDSDDA(0x0F);
}

void MOSTRAR_ALTURA()
{
    SET_PANTALLA(1,1,1);
    LCDPUTCHAR('A');
    LCDPUTCHAR('l');
    LCDPUTCHAR('t');
    LCDPUTCHAR('u');
    LCDPUTCHAR('r');
    LCDPUTCHAR('a');
    LCDPUTCHAR(' ');
    LCDPUTCHAR('l');
    LCDPUTCHAR('e');
    LCDPUTCHAR('m');
    LCDPUTCHAR(' ');
    LCDPUTCHAR(':');
    LCDPUTCHAR(li_cent_altura+0x30);
    LCDPUTCHAR(li_dec_altura+0x30);
    LCDPUTCHAR(li_uni_altura+0x30);
    switch (li_posicion)
    {
        case 1:
            {
                LCDSDDA(0xc);
                break;
            }
    }
}

```

```

        case 2:
            {
                LCDSDDA(0xd);
                break;
            }

        case 3:
            {
                LCDSDDA(0xe);
                break;
            }
    }
}

void MOSTRAR_PESO()
{
    SET_PANTALLA(1,1,1);
    LCDPUTCHAR('P');
    LCDPUTCHAR('e');
    LCDPUTCHAR('s');
    LCDPUTCHAR('o');
    LCDPUTCHAR(' ');
    LCDPUTCHAR('(');
    LCDPUTCHAR('K');
    LCDPUTCHAR('g');
    LCDPUTCHAR(')');
    LCDPUTCHAR(':');
    LCDPUTCHAR(li_cent_peso + 0x30);
    LCDPUTCHAR(li_dec_peso + 0x30);
    LCDPUTCHAR(li_uni_peso + 0x30);
    switch (li_posicion)
    {
        case 1:
            {
                LCDSDDA(0xA);
                break;
            }

        case 2:
            {
                LCDSDDA(0xB);
                break;
            }

        case 3:
            {
                LCDSDDA(0xC);
                break;
            }
    }
}

void MOSTRAR_EDAD()
{
    SET_PANTALLA(1,1,1);
    LCDPUTCHAR('E');
    LCDPUTCHAR('d');
    LCDPUTCHAR('a');
    LCDPUTCHAR('d');
    LCDPUTCHAR(':');
    LCDPUTCHAR(li_dec_edad + 0x30);
    LCDPUTCHAR(li_uni_edad + 0x30);
    switch (li_posicion)
    {
        case 1:
            {
                LCDSDDA(0x5);
                break;
            }

        case 2:
            {
                LCDSDDA(0x6);
            }
    }
}

```

```

        break;
    }
}

void MOSTRAR_CALIBRAR()
{
    SET_PANTALLA(1,1,0);
    LCDPUTCHAR(' ');
    LCDPUTCHAR('C');
    LCDPUTCHAR('a');
    LCDPUTCHAR('l');
    LCDPUTCHAR('i');
    LCDPUTCHAR('b');
    LCDPUTCHAR('r');
    LCDPUTCHAR('a');
    LCDPUTCHAR('n');
    LCDPUTCHAR('d');
    LCDPUTCHAR('o');
    LCDPUTCHAR('.');
    LCDPUTCHAR('.');
    LCDPUTCHAR('.');

    SET_PANTALLA(0,0,0);
    LCDPUTCHAR('P');
    LCDPUTCHAR('o');
    LCDPUTCHAR('r');
    LCDPUTCHAR(' ');
    LCDPUTCHAR('f');
    LCDPUTCHAR('a');
    LCDPUTCHAR('v');
    LCDPUTCHAR('o');
    LCDPUTCHAR('r');
    LCDPUTCHAR(' ');
    LCDPUTCHAR('e');
    LCDPUTCHAR('s');
    LCDPUTCHAR('p');
    LCDPUTCHAR('e');
    LCDPUTCHAR('r');
    LCDPUTCHAR('e');
}

void MOSTRAR_MIDIENDO()
{
    SET_PANTALLA(1,1,0);
    LCDPUTCHAR(' ');
    LCDPUTCHAR(' ');
    LCDPUTCHAR('M');
    LCDPUTCHAR('i');
    LCDPUTCHAR('d');
    LCDPUTCHAR('i');
    LCDPUTCHAR('e');
    LCDPUTCHAR('n');
    LCDPUTCHAR('d');
    LCDPUTCHAR('o');
    LCDPUTCHAR('.');
    LCDPUTCHAR('.');
    LCDPUTCHAR('.');

    SET_PANTALLA(0,0,0);
    LCDPUTCHAR('P');
    LCDPUTCHAR('o');
    LCDPUTCHAR('r');
    LCDPUTCHAR(' ');
    LCDPUTCHAR('f');
    LCDPUTCHAR('a');
    LCDPUTCHAR('v');
    LCDPUTCHAR('o');
    LCDPUTCHAR('r');
    LCDPUTCHAR(' ');
    LCDPUTCHAR('e');
    LCDPUTCHAR('s');
    LCDPUTCHAR('p');
    LCDPUTCHAR('e');
    LCDPUTCHAR('r');
    LCDPUTCHAR('e');
}

```

```

    }

void MOSTRAR_R()
{
    SET_PANTALLA(1,1,0);
    LCDPUTCHAR(' ');
    LCDPUTCHAR(' ');
    LCDPUTCHAR(' ');
    LCDPUTCHAR('R');
    LCDPUTCHAR(':');
    LCDPUTCHAR(' ');
    CONV_BCD_MOSTRAR(R,0);
    LCDPUTCHAR(' ');
    LCDPUTCHAR(0xF4);
}

void MOSTRAR_FI()
{
    SET_PANTALLA(0,0,0);
    LCDPUTCHAR(' ');
    LCDPUTCHAR(' ');
    LCDPUTCHAR(0xF2);
    LCDPUTCHAR(':');
    LCDPUTCHAR(' ');
    CONV_BCD_MOSTRAR(FI,0);
    LCDPUTCHAR(' ');
    LCDPUTCHAR('g');
    LCDPUTCHAR('r');
    LCDPUTCHAR('.');
}

void MOSTRAR_TBW()
{
    SET_PANTALLA(1,1,0);
    LCDPUTCHAR('T');
    LCDPUTCHAR('B');
    LCDPUTCHAR('W');
    LCDPUTCHAR(':');
    LCDPUTCHAR(' ');
    CONV_BCD_MOSTRAR(TBW,0);
    LCDPUTCHAR(' ');
    LCDPUTCHAR('L');
    LCDPUTCHAR('t');
    LCDPUTCHAR('s');
    LCDPUTCHAR('.');
}

void MOSTRAR_FFM()
{
    SET_PANTALLA(0,0,0);
    LCDPUTCHAR(' ');
    LCDPUTCHAR('F');
    LCDPUTCHAR('F');
    LCDPUTCHAR('M');
    LCDPUTCHAR(':');
    LCDPUTCHAR(' ');
    CONV_BCD_MOSTRAR(FFM,0);
    LCDPUTCHAR(' ');
    LCDPUTCHAR('K');
    LCDPUTCHAR('g');
    LCDPUTCHAR('.');
}

void MOSTRAR_FM()
{
    SET_PANTALLA(1,1,0);
    LCDPUTCHAR(' ');
    LCDPUTCHAR('F');
    LCDPUTCHAR('M');
    LCDPUTCHAR(':');
    LCDPUTCHAR(' ');
    CONV_BCD_MOSTRAR(FM,0);
    LCDPUTCHAR(' ');
    LCDPUTCHAR('K');
    LCDPUTCHAR('g');
    LCDPUTCHAR('.');
}

```

```

    }

void toggle_digito_altura()
{
    li_posicion=li_posicion+1;
    if(li_posicion==4)
    {
        li_posicion=1;
    }
}

void toggle_digito_edad()
{
    li_posicion=li_posicion+1;
    if(li_posicion==3)
    {
        li_posicion=1;
    }
}

void INCREMENTAR_ALTURA()
{
    switch (li_posicion)
    {
        case 1:
            {
                li_cent_altura= li_cent_altura+1;
                if(li_cent_altura==10)
                {
                    li_cent_altura=0;
                }
            }
            break;
        }
        case 2:
            {
                li_dec_altura= li_dec_altura+1;
                if(li_dec_altura==10)
                {
                    li_dec_altura=0;
                }
            }
            break;
        }
        case 3:
            {
                li_uni_altura= li_uni_altura+1;
                if(li_uni_altura==10)
                {
                    li_uni_altura=0;
                }
            }
            break;
        }
    }
}

void INCREMENTAR_PESO()
{
    switch (li_posicion)
    {
        case 1:
            {
                li_cent_peso = li_cent_peso+1;
                if(li_cent_peso==3)
                {
                    li_cent_peso=0;
                }
            }
            break;
        }
        case 2:
            {
                li_dec_peso = li_dec_peso+1;
                if(li_dec_peso==10)
                {
                    li_dec_peso=0;
                }
            }
            break;
        }
    }
}

```

```

        case 3:
            {
                li_uni_peso=    li_uni_peso+1;
                if(li_uni_peso==10)
                    {
                        li_uni_peso=0;
                    }
                break;
            }
        }
    }

void INCREMENTAR_EDAD()
{
    switch (li_posicion)
    {
        case 1:
            {
                li_dec_edad =    li_dec_edad+1;
                if(li_dec_edad==10)
                    {
                        li_dec_edad=0;
                    }
                break;
            }
        case 2:
            {
                li_uni_edad =    li_uni_edad+1;
                if(li_uni_edad==10)
                    {
                        li_uni_edad=0;
                    }
                break;
            }
        }
    }

void toggle_otro()
{
    if(ls_otro=='S')
        {
            ls_otro='N';
        }
    else
        {
            ls_otro='S';
        }
}

void toggle_iniciar()
{
    if(ls_iniciar=='S')
        {
            ls_iniciar='N';
        }
    else
        {
            ls_iniciar='S';
        }
}

void togglesexo()
{
    if(ls_sexo=='F')
        {
            ls_sexo='M';
        }
    else
        {
            ls_sexo='F';
        }
}

#int_EXT
void EXT_isr()

```

```

{
#asm
//store current state of processor
MOVWF save_w
SWAPF status,W
BCF status,5
BCF status,6
MOVWF save_status
#endasm
if (!interrumpido)
{
interrumpido=1;
USAR_ADC();
CONV_CE=0; //;SACO DEL TERCER ESTADO
li_dato_leido=input_b();
CONV_CE=1; //;PONGO EN TERCER ESTADO

if (input(PIN_C7))
{
li_indice_par++;
li_suma_par=li_suma_par + li_dato_leido;
}
else
{
li_indice_impar++;
li_suma_impar=li_suma_impar + li_dato_leido;
}
interrumpido=0;
}

int_flag=0;
USAR_INT();
enable_interrupts(GLOBAL);

// restore processor and return from interrupt
#asm
SWAPF save_status,W
MOVWF status
SWAPF save_w,F
SWAPF save_w,W
#endasm
}

void calculo()
{
float valor;
int indice;
//Obtengo la amplitud #####
//2*pi*50000*1.6*10^-6 = 0.5026548245
indice=((0.5026548245 - FI*PI/180 - 0.3281218993) / 0.001745329252);
valor=read_eeprom (indice);
valor=valor/255;
A=(n1-lf_continua)/valor;
//tengo el valor de A en n, para pasarlo a volts
// V=(Muestras(3.33-1.58)/(255*1.41))=0.004852693596
// y luego divido a la tension por los 0.8 mA , todo eso da 6.065866994
R=A*6.065866994;

#####

//Obtengo Xc #####
indice = (FI - 2) / 0.1 + 81;
valor=read_eeprom (indice);
valor=valor/255;
Xc=R*valor;
#####
// H=li_cent_altura*100 + li_dec_altura*10 + li_uni_altura;
//Usamos la variable FM en vez de la H porque no tenemos espacio en RAM
FM=(float)li_cent_altura*100 + (float)li_dec_altura*10 + (float)li_uni_altura;
W=li_cent_peso*100 + li_dec_peso*10 + li_uni_peso;
E=li_dec_edad*10 + li_uni_edad;

//Usamos la variable FM en vez de la H porque no tenemos espacio en RAM
TBW = 0.377 * ((float)FM*(float)FM)/R + 0.14 * W - 0.080 * E + 2.90 * S + 4.65 ;

```

$$FFM = -4.104 + (0.518 * (\text{float})FM * (\text{float})FM / R) + 0.231 * W + 0.13 * Xc + 4.229 * S;$$

```

FM=W - FFM;
}
void main()
{
int li_salir=0;
int n;
int li_demora=500;
setup_adc_ports(NO_ANALOGS);
setup_adc(ADC_OFF);
setup_spi(FALSE);
setup_timer_0(RTCC_INTERNAL|RTCC_DIV_1);
setup_timer_1(T1_DISABLED);
setup_timer_2(T2_DISABLED,0,1);
disable_interrupts(GLOBAL);

output_a(0x00);
output_b(0x00);
output_c(0x00);
//Ports
set_tris_a(0b00111000);
set_tris_c(0B10000100); //RC7 como entrada para leer si es muestra PAR, RC2 INICIALMENTE como entrada y luego
como salida para generar la adquisicion
habil_dds=0;
delay_ms(10);
ADQ=1;
habil_dds=1; //habilito al DDS para que empiece a generar
LCDINIT();
USAR_LCD();
MOSTRAR_PRESENTACION();

for(;;)
{
restart_wdt();
while (ls_iniciar=='N')
{
//Se queda esperando que se apriete el ENTER
while ( !input(ENTER) )
{}
MOSTRAR_SEXO();
delay_ms(li_demora);
li_salir=0;
while (!li_salir)
{
if ( input(CHANGE))
{
toggle_sexo();
MOSTRAR_SEXO();
delay_ms(li_demora);
}
else
if (input(ENTER))
{
li_salir=1;
}
}

restart_wdt();
li_dec_edad=0;
li_uni_edad=0;
li_posicion=1;
MOSTRAR_EDAD();
delay_ms(li_demora);
li_salir=0;
while (!li_salir)
{
if ( input(CHANGE))
{
INCREMENTAR_EDAD();
MOSTRAR_EDAD();
delay_ms(li_demora);
}
else
{
if ( input(TAB))
{

```

```

toggle_digito_edad();
MOSTRAR_EDAD();
delay_ms(li_demora);
}
else
{
if (input(ENTER)&& ( li_dec_edad >0 || li_uni_edad >0))
{
li_salir=1;
}
}
}
E=li_dec_edad*10 + li_uni_edad;
if(E==30)
{
li_cent_altura=4;
li_dec_altura=0;
li_uni_altura=0;
}
restart_wdt();
li_posicion=1;
MOSTRAR_ALTURA();
delay_ms(li_demora);
li_salir=0;
while (!li_salir)
{
if ( input(CHANGE))
{
INCREMENTAR_ALTURA();
MOSTRAR_ALTURA();
delay_ms(li_demora);
}
else
{
if ( input(TAB))
{
toggle_digito_altura();
MOSTRAR_ALTURA();
delay_ms(li_demora);
}
}
}
if (input(ENTER) && (li_cent_altura >0 || li_dec_altura >0 || li_uni_altura >0))
{
li_salir=1;
}
}
}
if (E==30)
{
li_cent_peso=0;
li_dec_peso=5;
li_uni_peso=0;
}

restart_wdt();
li_posicion=1;
MOSTRAR_PESO();
delay_ms(li_demora);
li_salir=0;
while (!li_salir)
{
if ( input(CHANGE))
{
INCREMENTAR_PESO();
MOSTRAR_PESO();
delay_ms(li_demora);
}
else
{
if ( input(TAB))
{
toggle_digito_altura();
}
}
}

```

```

        MOSTRAR_PESO();
        delay_ms(li_demora);
    }
    else
    {
        if (input(ENTER)&& (li_cent_peso >0 || li_dec_peso >0 || li_uni_peso >0))
        {
            li_salir=1;
        }
    }
}
restart_wdt();
ls_iniciar='S';
MOSTRAR_INICIAR_TEST();
delay_ms(li_demora);
li_salir=0;
while (!li_salir)
{
    if ( input(CHANGE))
    {
        toggle_iniciar();
        MOSTRAR_INICIAR_TEST();
        delay_ms(li_demora);
    }
    else
    {
        if (input(ENTER))
        {
            li_salir=1;
        }
    }
} //Cierro el while que pregunta por ls_iniciar
restart_wdt();
//Inicio del test
for (n=0;n<=1;+n)
{
    habil_dds=0;
    delay_ms(10);
    ADQ=1;
    habil_dds=1; //habilito al DDS para que empiece a generar
    if (n==0) //calibrando
    {
        MOSTRAR_CALIBRAR();
    }
    else
    {
        rele=1; //Conecto al paciente
        MOSTRAR_MIDIENDO();
    }

    restart_wdt();
    USAR_INT();
    enable_interrupts(INT_EXT);
    enable_interrupts(GLOBAL);
    li_indice_par=0;
    li_indice_impar=0;
    li_suma_par=0;
    li_suma_impar=0;
    while( !(li_indice_par>=li_cant_muestras) && (li_indice_impar >= li_cant_muestras) );
    {
    }
    disable_interrupts(GLOBAL);
    disable_interrupts(INT_EXT);
    n2=(float)li_suma_par/(float)(li_indice_par );
    n1=(float)li_suma_impar/(float)(li_indice_impar);
    habil_dds=0; //Deshabilito al DDS para hacer que el dds no adquiera mas y comenzar a adquirir desde el BIA
    delay_ms(10);
    ADQ=0;
    habil_dds=1;
    set_tris_c(0B10000000); //RC7 como entrada para leer si es muestra PAR, RC2 como salida para generar la adquisición
    //Luego de tener la fase, obtengo el valor medio
    USAR_ADC();
    CONV_CLK=0;
    li_suma_continua=0;
    li_indice_continua=0;
    for (i=0;i<li_cant_muestras_continua;+i) //5

```

```

    {
    li_indice_continua=li_indice_continua + 1; //1

    CONV_CLK=1; //genero el pulso de adquisicion //1

    CONV_CE=0; //;SACO DEL TERCER ESTADO //1
    li_suma_continua=li_suma_continua + input_b(); //11
    CONV_CE=1; //;PONGO EN TERCER ESTADO //1
    CONV_CLK=0; //reseteo la linea de adquisicion//1
    #asm
    nop
    //1
    nop
    //1
    nop
    //1
    #endasm
    }//3

    lf_continua=(float)li_suma_continua/(float)li_indice_continua;
    if (n==1)
    { rele=0;} //Desconecto al paciente

    //Obtengo la FASE #####
    //sin(2*pi*50000*1.6 useg)=0.4817536741
    //cos(2*pi*50000*1.6 useg)=0.87630668
    //sin(2*pi*50000*12 useg)=-0.587785252
    //cos(2*pi*50000*12 useg)=-0.809016994

    FI=atan( ( (n2-lf_continua) * 0.4817536741 - (n1-lf_continua) * (-0.587785252) ) / ( (n2-lf_continua) * 0.87630668 -
(n1-lf_continua) * (-0.809016994) ) );
    FI=FI*180/PI;
    if (FI<0)
    {FI=FI+180;}
    #####
    if (n==0)
    {TBW=FI;}//Tanto en esta linea como en la siguiente, uso la variable TBW en vez de una que debería
    llamarse FI_sistema pero no me queda más RAM para
    // declarar variables del tipo flotante.No hay problema porque TBW se instancia más adelante

} //Cierro el FOR que realiza la medicion dos veces: una sobre la resistencia de calibración y otra sobre el paciente

//Tengo en cuenta la fase del sistema
FI=FI-TBW;

calculo();

USAR_LCD();
MOSTRAR_R();
MOSTRAR_FI();
delay_ms(li_demora);
restart_wdt();
li_salir=0;
while (!li_salir)
{
if (input(ENTER))
{
li_salir=1;
}
}
MOSTRAR_TBW();
MOSTRAR_FFM();
delay_ms(li_demora);
restart_wdt();
li_salir=0;
while (!li_salir)
{
if (input(ENTER))
{
li_salir=1;
}
}
MOSTRAR_FM();
delay_ms(li_demora);
restart_wdt();

```

```

li_salir=0;
while (!li_salir)
{
    if (input(ENTER))
    {
        li_salir=1;
    }
}
restart_wdt();
ls_iniciar='N';
ls_otro='S';
MOSTRAR_OTRO_TEST();
delay_ms(li_demora);
li_salir=0;
while (!li_salir)
{
    if ( input(CHANGE))
    {
        toggle_otro();
        MOSTRAR_OTRO_TEST();
        delay_ms(li_demora);
    }
    else
    {
        if (input(ENTER))
        {
            li_salir=1;
        }
    }
}
restart_wdt();
while (ls_otro=='N')
{
    MOSTRAR_R();
    MOSTRAR_FI();
    delay_ms(li_demora);
    li_salir=0;
    while (!li_salir)
    {
        if (input(ENTER))
        {
            li_salir=1;
        }
    }

    MOSTRAR_TBW();
    MOSTRAR_FFM();
    delay_ms(li_demora);

    li_salir=0;
    while (!li_salir)
    {
        if (input(ENTER))
        {
            li_salir=1;
        }
    }

    MOSTRAR_FM();
    delay_ms(li_demora);
    li_salir=0;
    while (!li_salir)
    {
        if (input(ENTER))
        {
            li_salir=1;
        }
    }
}
ls_otro='S';
MOSTRAR_OTRO_TEST();
delay_ms(li_demora);
li_salir=0;
while (!li_salir)
if ( input(CHANGE))
{
    toggle_otro();
    MOSTRAR_OTRO_TEST();
    delay_ms(li_demora);
}

```

```
        }
    else
    {
        if (input(ENTER))
        {
            li_salir=1;
        }
    }
}
} //cierre del while que pregunta por ls_otro
} // cierre del for(;;)
}
```

Código DDS (Assembler)

LIST P=16F873

#INCLUDE <p16f873.inc>

__CONFIG H'3F3A'

DAC_DATOS EQU PORTB
DAC_DATOS_TRIS EQU TRISB

CONV_CONTROL EQU PORTA
CONV_CONTROL_TRIS EQU TRISA

SINCRO EQU PORTA
HABIL_DDS EQU 0
INTERRUP EQU 1
MUESTRA_PAR EQU 2
CONV_CLK EQU 3

org 0
goto Start
org 5

INICIALIZAR_PUERTOS

BANKSEL DAC_DATOS
CLRF DAC_DATOS
CLRF SINCRO

BANKSEL DAC_DATOS_TRIS
CLRF DAC_DATOS_TRIS ;PUERTO B COMO SALIDA

MOVLW B'00000111'
MOVWF ADCON1 ; PORTA como digital

MOVLW B'11110001'
MOVWF CONV_CONTROL_TRIS ; RA0 = ENTRADA PARA LA HABILITACION DEL DDS
; RA1 = SALIDA PARA GENERAR INTERRUPCION
; RA2 = SALIDA PARA INDICAR MUESTRA PAR
; RA3 = SALIDA PARA LOS PULSOS DE CLK DEL ADC

BANKSEL DAC_DATOS ; ME PARO EN EL BANCO 0

RETURN

Start

CALL INICIALIZAR_PUERTOS

CHECKEAR_HABIL

BTFSNC SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SENOA
GOTO CHK_ADQ
GOTO CHECKEAR_HABIL

CHK_ADQ

BTFSNC SINCRO,ADQ ;CHECKEO SI TENGO HABILITACION PARA GENERAR LA SEÑAL DE ADQUISICION PARA EL

TDA

GOTO PORT_CON_ADQ
GOTO PORT_SIN_ADQ

PORT_CON_ADQ

MOVLW B'11110001'
MOVWF CONV_CONTROL_TRIS ; RA0 = ENTRADA PARA LA HABILITACION DEL DDS
; RA1 = SALIDA PARA GENERAR INTERRUPCION
; RA2 = SALIDA PARA INDICAR MUESTRA PAR
; RA3 = SALIDA PARA LOS PULSOS DE CLK DEL ADC

GOTO GENERAR_CON_ADQ

PORT_SIN_ADQ

MOVLW B'11110001'
MOVWF CONV_CONTROL_TRIS ; RA0 = ENTRADA PARA LA HABILITACION DEL DDS
; RA1 = SALIDA PARA GENERAR INTERRUPCION
; RA2 = SALIDA PARA INDICAR MUESTRA PAR

; RA3 = ENTRADA PARA QUE NO JODA Y EL BIA

MANEJE LA INTERRUPCION
GOTO GENERAR_SIN_ADQ

;CANTIDAD DE CICLOS DE INSTRUCCION NECESARIOS PARA SACAR UN VALOR -->> 4
;20 uSEG / 0.2 uSEG * 4 = 25 MUESTRAS

GENERAR_CON_ADQ

MOVLW d'127'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'134'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'141'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'148'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'152'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'156'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'157'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'156'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'154'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'150'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'145'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'138'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'131'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'123'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'116'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'109'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'104'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'100'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'98'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'97'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'98'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0;
GOTO CHECKEAR_HABIL

MOVLW d'102'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'106'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'112'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'119'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

;Fin del primer ciclo de la semoidal

MOVLW d'127'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'134'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'141'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0

GOTO CHECKEAR_HABIL

MOVLW d'148'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'152'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'156'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'157'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'156'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'154'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'150'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'145'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'138'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'131'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'123'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'116'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'109'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'104'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'100'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'98'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'97'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'98'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0;
GOTO CHECKEAR_HABIL

MOVLW d'102'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'106'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'112'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'119'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

;Fin del segundo ciclo

MOVLW d'127'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'134'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'141'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'148'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'152'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'156'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'157'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'156'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0

GOTO CHECKEAR_HABIL

MOVLW d'154'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'150'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'145'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'138'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'131'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'123'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'116'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'109'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'104'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'100'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'98'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'97'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'98'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0;
GOTO CHECKEAR_HABIL

MOVLW d'102'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'106'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

```
MOVLW d'112'  
MOVWF DAC_DATOS  
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0  
GOTO CHECKEAR_HABIL
```

```
MOVLW d'119'  
MOVWF DAC_DATOS  
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0  
GOTO CHECKEAR_HABIL
```

;Fin del Tercer Ciclo

```
MOVLW d'127'  
MOVWF DAC_DATOS  
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0  
GOTO CHECKEAR_HABIL
```

```
MOVLW d'134'  
MOVWF DAC_DATOS  
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0  
GOTO CHECKEAR_HABIL
```

```
MOVLW d'141'  
MOVWF DAC_DATOS  
BSF CONV_CONTROL,CONV_CLK ;MANDO A ADQUIRIR UNA MUESTRA  
BCF SINCRO, MUESTRA_PAR ;INDICO QUE ES UNA MUESTRA IMPAR
```

```
MOVLW d'148'  
MOVWF DAC_DATOS  
BSF CONV_CONTROL,INTERRUPT;GENERO LA INTERRUPCION  
BCF CONV_CONTROL,CONV_CLK ;BAJO LA LINEA DE CLK
```

```
MOVLW d'152'  
MOVWF DAC_DATOS  
BCF CONV_CONTROL,INTERRUPT;BAJO LA LINEA DE INTERRUPCION  
NOP
```

```
MOVLW d'156'  
MOVWF DAC_DATOS  
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0  
GOTO CHECKEAR_HABIL
```

```
MOVLW d'157'  
MOVWF DAC_DATOS  
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0  
GOTO CHECKEAR_HABIL
```

```
MOVLW d'156'  
MOVWF DAC_DATOS  
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0  
GOTO CHECKEAR_HABIL
```

```
MOVLW d'154'  
MOVWF DAC_DATOS  
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0  
GOTO CHECKEAR_HABIL
```

```
MOVLW d'150'  
MOVWF DAC_DATOS  
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0  
GOTO CHECKEAR_HABIL
```

```
MOVLW d'145'  
MOVWF DAC_DATOS  
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0  
GOTO CHECKEAR_HABIL
```

```
MOVLW d'138'  
MOVWF DAC_DATOS  
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0  
GOTO CHECKEAR_HABIL
```

```
MOVLW d'131'  
MOVWF DAC_DATOS
```

BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'123'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'116'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'109'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'104'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'100'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'98'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'97'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'98'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0;
GOTO CHECKEAR_HABIL

MOVLW d'102'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'106'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'112'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'119'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

;Fin del cuarto ciclo

MOVLW d'127'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'134'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'141'

MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'148'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'152'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'156'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'157'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'156'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'154'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'150'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'145'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'138'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'131'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'123'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'116'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'109'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'104'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'100'
MOVWF DAC_DATOS

BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'98'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'97'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'98'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0;
GOTO CHECKEAR_HABIL

MOVLW d'102'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'106'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'112'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'119'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

;Fin del Quinto ciclo

MOVLW d'127'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'134'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'141'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'148'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'152'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'156'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'157'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'156'

MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SENO
GOTO CHECKEAR_HABIL

MOVLW d'154'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SENO
GOTO CHECKEAR_HABIL

MOVLW d'150'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SENO
GOTO CHECKEAR_HABIL

MOVLW d'145'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SENO
GOTO CHECKEAR_HABIL

MOVLW d'138'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SENO
GOTO CHECKEAR_HABIL

MOVLW d'131'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SENO
GOTO CHECKEAR_HABIL

MOVLW d'123'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SENO
GOTO CHECKEAR_HABIL

MOVLW d'116'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SENO
GOTO CHECKEAR_HABIL

MOVLW d'109'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SENO
GOTO CHECKEAR_HABIL

MOVLW d'104'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SENO
GOTO CHECKEAR_HABIL

MOVLW d'100'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SENO
GOTO CHECKEAR_HABIL

MOVLW d'98'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SENO
GOTO CHECKEAR_HABIL

MOVLW d'97'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SENO
GOTO CHECKEAR_HABIL

MOVLW d'98'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SENO;
GOTO CHECKEAR_HABIL

MOVLW d'102'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SENO
GOTO CHECKEAR_HABIL

MOVLW d'106'

MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN
GOTO CHECKEAR_HABIL

MOVLW d'112'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN
GOTO CHECKEAR_HABIL

MOVLW d'119'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN
GOTO CHECKEAR_HABIL

;Fin del sexto ciclo

MOVLW d'127'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN
GOTO CHECKEAR_HABIL

MOVLW d'134'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN
GOTO CHECKEAR_HABIL

MOVLW d'141'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN
GOTO CHECKEAR_HABIL

MOVLW d'148'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN
GOTO CHECKEAR_HABIL

MOVLW d'152'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN
GOTO CHECKEAR_HABIL

MOVLW d'156'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN
GOTO CHECKEAR_HABIL

MOVLW d'157'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN
GOTO CHECKEAR_HABIL

MOVLW d'156'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN
GOTO CHECKEAR_HABIL

MOVLW d'154'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN
GOTO CHECKEAR_HABIL

MOVLW d'150'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN
GOTO CHECKEAR_HABIL

MOVLW d'145'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN
GOTO CHECKEAR_HABIL

MOVLW d'138'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN
GOTO CHECKEAR_HABIL


```
MOVLW d'119'  
MOVWF DAC_DATOS  
GOTO GENERAR_CON_ADQ
```

```
;#####
```

```
GENERAR_SIN_ADQ
```

```
MOVLW d'127'  
MOVWF DAC_DATOS  
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0  
GOTO CHECKEAR_HABIL
```

```
MOVLW d'134'  
MOVWF DAC_DATOS  
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0  
GOTO CHECKEAR_HABIL
```

```
MOVLW d'141'  
MOVWF DAC_DATOS  
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0  
GOTO CHECKEAR_HABIL
```

```
MOVLW d'148'  
MOVWF DAC_DATOS  
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0  
GOTO CHECKEAR_HABIL
```

```
MOVLW d'152'  
MOVWF DAC_DATOS  
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0  
GOTO CHECKEAR_HABIL
```

```
MOVLW d'156'  
MOVWF DAC_DATOS  
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0  
GOTO CHECKEAR_HABIL
```

```
MOVLW d'157'  
MOVWF DAC_DATOS  
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0  
GOTO CHECKEAR_HABIL
```

```
MOVLW d'156'  
MOVWF DAC_DATOS  
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0  
GOTO CHECKEAR_HABIL
```

```
MOVLW d'154'  
MOVWF DAC_DATOS  
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0  
GOTO CHECKEAR_HABIL
```

```
MOVLW d'150'  
MOVWF DAC_DATOS  
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0  
GOTO CHECKEAR_HABIL
```

```
MOVLW d'145'  
MOVWF DAC_DATOS  
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0  
GOTO CHECKEAR_HABIL
```

```
MOVLW d'138'  
MOVWF DAC_DATOS  
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0  
GOTO CHECKEAR_HABIL
```

```
MOVLW d'131'  
MOVWF DAC_DATOS  
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0  
GOTO CHECKEAR_HABIL
```

MOVLW d'123'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'116'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'109'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'104'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'100'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'98'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'97'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'98'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0;
GOTO CHECKEAR_HABIL

MOVLW d'102'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'106'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'112'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'119'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

;Fin del primer ciclo de la semoidal

MOVLW d'127'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'134'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'141'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0

GOTO CHECKEAR_HABIL

MOVLW d'148'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'152'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'156'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'157'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'156'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'154'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'150'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'145'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'138'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'131'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'123'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'116'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'109'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'104'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'100'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'98'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SENO
GOTO CHECKEAR_HABIL

MOVLW d'97'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SENO
GOTO CHECKEAR_HABIL

MOVLW d'98'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SENO;
GOTO CHECKEAR_HABIL

MOVLW d'102'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SENO
GOTO CHECKEAR_HABIL

MOVLW d'106'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SENO
GOTO CHECKEAR_HABIL

MOVLW d'112'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SENO
GOTO CHECKEAR_HABIL

MOVLW d'119'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SENO
GOTO CHECKEAR_HABIL

;Fin del segundo ciclo

MOVLW d'127'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SENO
GOTO CHECKEAR_HABIL

MOVLW d'134'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SENO
GOTO CHECKEAR_HABIL

MOVLW d'141'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SENO
GOTO CHECKEAR_HABIL

MOVLW d'148'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SENO
GOTO CHECKEAR_HABIL

MOVLW d'152'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SENO
GOTO CHECKEAR_HABIL

MOVLW d'156'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SENO
GOTO CHECKEAR_HABIL

MOVLW d'157'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SENO
GOTO CHECKEAR_HABIL

MOVLW d'156'

MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'154'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'150'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'145'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'138'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'131'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'123'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'116'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'109'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'104'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'100'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'98'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'97'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'98'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0;
GOTO CHECKEAR_HABIL

MOVLW d'102'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'106'
MOVWF DAC_DATOS

BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SENO
GOTO CHECKEAR_HABIL

MOVLW d'112'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SENO
GOTO CHECKEAR_HABIL

MOVLW d'119'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SENO
GOTO CHECKEAR_HABIL

;Fin del Tercer Ciclo

MOVLW d'127'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SENO
GOTO CHECKEAR_HABIL

MOVLW d'134'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SENO
GOTO CHECKEAR_HABIL

MOVLW d'141'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SENO
GOTO CHECKEAR_HABIL

MOVLW d'148'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SENO
GOTO CHECKEAR_HABIL

MOVLW d'152'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SENO
GOTO CHECKEAR_HABIL

MOVLW d'156'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SENO
GOTO CHECKEAR_HABIL

MOVLW d'157'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SENO
GOTO CHECKEAR_HABIL

MOVLW d'156'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SENO
GOTO CHECKEAR_HABIL

MOVLW d'154'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SENO
GOTO CHECKEAR_HABIL

MOVLW d'150'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SENO
GOTO CHECKEAR_HABIL

MOVLW d'145'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SENO
GOTO CHECKEAR_HABIL

MOVLW d'138'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SENO
GOTO CHECKEAR_HABIL

MOVLW d'131'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'123'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'116'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'109'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'104'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'100'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'98'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'97'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'98'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0;
GOTO CHECKEAR_HABIL

MOVLW d'102'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'106'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'112'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'119'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

;Fin del cuarto ciclo

MOVLW d'127'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'134'

MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'141'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'148'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'152'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'156'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'157'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'156'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'154'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'150'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'145'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'138'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'131'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'123'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'116'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'109'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'104'
MOVWF DAC_DATOS

BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SENO
GOTO CHECKEAR_HABIL

MOVLW d'100'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SENO
GOTO CHECKEAR_HABIL

MOVLW d'98'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SENO
GOTO CHECKEAR_HABIL

MOVLW d'97'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SENO
GOTO CHECKEAR_HABIL

MOVLW d'98'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SENO;
GOTO CHECKEAR_HABIL

MOVLW d'102'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SENO
GOTO CHECKEAR_HABIL

MOVLW d'106'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SENO
GOTO CHECKEAR_HABIL

MOVLW d'112'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SENO
GOTO CHECKEAR_HABIL

MOVLW d'119'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SENO
GOTO CHECKEAR_HABIL

;Fin del Quinto ciclo

MOVLW d'127'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SENO
GOTO CHECKEAR_HABIL

MOVLW d'134'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SENO
GOTO CHECKEAR_HABIL

MOVLW d'141'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SENO
GOTO CHECKEAR_HABIL

MOVLW d'148'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SENO
GOTO CHECKEAR_HABIL

MOVLW d'152'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SENO
GOTO CHECKEAR_HABIL

MOVLW d'156'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SENO
GOTO CHECKEAR_HABIL

MOVLW d'157'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'156'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'154'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'150'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'145'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'138'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'131'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'123'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'116'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'109'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'104'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'100'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'98'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'97'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'98'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0;
GOTO CHECKEAR_HABIL

MOVLW d'102'

MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SENO
GOTO CHECKEAR_HABIL

MOVLW d'106'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SENO
GOTO CHECKEAR_HABIL

MOVLW d'112'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SENO
GOTO CHECKEAR_HABIL

MOVLW d'119'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SENO
GOTO CHECKEAR_HABIL

;Fin del sexto ciclo

MOVLW d'127'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SENO
GOTO CHECKEAR_HABIL

MOVLW d'134'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SENO
GOTO CHECKEAR_HABIL

MOVLW d'141'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SENO
GOTO CHECKEAR_HABIL

MOVLW d'148'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SENO
GOTO CHECKEAR_HABIL

MOVLW d'152'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SENO
GOTO CHECKEAR_HABIL

MOVLW d'156'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SENO
GOTO CHECKEAR_HABIL

MOVLW d'157'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SENO
GOTO CHECKEAR_HABIL

MOVLW d'156'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SENO
GOTO CHECKEAR_HABIL

MOVLW d'154'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SENO
GOTO CHECKEAR_HABIL

MOVLW d'150'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SENO
GOTO CHECKEAR_HABIL

MOVLW d'145'
MOVWF DAC_DATOS
BTFSS SINCRO,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SENO
GOTO CHECKEAR_HABIL

```

MOVLW d'138'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'131'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'123'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'116'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'109'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'104'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'100'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'98'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'97'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'98'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'102'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'106'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'112'
MOVWF DAC_DATOS
BTFSS SINCR0,HABIL_DDS ;CHECKEO SI TENGO HABILITACION PARA EMPEZAR A SINTETIZAR LA SEN0
GOTO CHECKEAR_HABIL

MOVLW d'119'
MOVWF DAC_DATOS

GOTO GENERAR_SIN_ADQ

END

```

ANEXO 2: Lista de componentes

SEMICONDUCTORES

Microcontrolador PIC16F873

Microcontrolador PIC16F876

Convertor DAC800

AO INA114

AO TL081

2AO TL082

OTA3080

Buffer DM74LS126

Convertor ADC TDA8703

Regulador LM7805

Regulador LM7905

TBJ BC548

3 diodos 1N4048

Resistencias (todas de 1/4Watt)

8*10K Ω

12*1K Ω

2*820 Ω

1*4.7K Ω

1*470 Ω

1*22 Ω

1*390k Ω

Resistencias variables

3 Potenciómetros blindados 50K Ω

1 Potenciómetro 5K Ω

1 Potenciómetro 10K Ω

1 Potenciómetro de ajuste vertical 100K Ω

2 presets 1K Ω

Resistencias de contacto metálico

3*1K Ω

Capacitores Electrolíticos

4.7nF 50V

100uF 16V

Capacitores cerámicos

4*22pF

1*560pf

1*47nF

2*100nF

Capacitores multicapa / poliester

2*1nF

1*82nF

1*10nF

VARIOS

2 Relays HJR 1-2C

Display 16*2 dígitos Hitachi compatible

3 Pulsadores sin retención

1 interruptor

1 led verde de 3mm

2 cristales $f_0=20\text{MHz}$.

2 borneras para dos contactos a tornillo

60 Postes macho

60 Postes hembra para cable.

Electrodos de ECG autoadhesivos